Salim Chikhi · Abdelmalek Amine ·
Allaoua Chaoui ·
Djamel Eddine Saidouni ·
Mohamed Khireddine Kholladi   *Editors*

Modelling and Implementation of Complex Systems

Proceedings of the 6th International
Symposium, MISC 2020, Batna, Algeria,
October 24—26, 2020

🖄 Springer

Lecture Notes in Networks and Systems

Volume 156

The series "Lecture Notes in Networks and Systems" publishes the latest developments in Networks and Systems—quickly, informally and with high quality. Original research reported in proceedings and post-proceedings represents the core of LNNS.

Volumes published in LNNS embrace all aspects and subfields of, as well as new challenges in, Networks and Systems.

The series contains proceedings and edited volumes in systems and networks, spanning the areas of Cyber-Physical Systems, Autonomous Systems, Sensor Networks, Control Systems, Energy Systems, Automotive Systems, Biological Systems, Vehicular Networking and Connected Vehicles, Aerospace Systems, Automation, Manufacturing, Smart Grids, Nonlinear Systems, Power Systems, Robotics, Social Systems, Economic Systems and other. Of particular value to both the contributors and the readership are the short publication timeframe and the world-wide distribution and exposure which enable both a wide and rapid dissemination of research output.

The series covers the theory, applications, and perspectives on the state of the art and future developments relevant to systems and networks, decision making, control, complex processes and related areas, as embedded in the fields of interdisciplinary and applied sciences, engineering, computer science, physics, economics, social, and life sciences, as well as the paradigms and methodologies behind them.

** Indexing: The books of this series are submitted to ISI Proceedings, SCOPUS, Google Scholar and Springerlink **

More information about this series at http://www.springer.com/series/15179

Salim Chikhi · Abdelmalek Amine ·
Allaoua Chaoui · Djamel Eddine Saidouni ·
Mohamed Khireddine Kholladi
Editors

Modelling and Implementation of Complex Systems

Proceedings of the 6th International Symposium, MISC 2020, Batna, Algeria, October 24–26, 2020

Editors
Salim Chikhi
Faculty of New Information
and Communication Technologies
University of Constantine 2
Constantine, Algeria

Allaoua Chaoui
Faculty of New Information
and Communication Technologies
University of Constantine 2
Constantine, Algeria

Mohamed Khireddine Kholladi
Faculty of Science and Technology,
Department of Mathematics
and Computer Science
University of El Oued
El-Oued, Algeria

Abdelmalek Amine
Faculty of Technology
University of Saida
Saida, Algeria

Djamel Eddine Saidouni
Faculty of New Information
and Communication Technologies
University of Constantine 2
Constantine, Algeria

Preface

This volume contains research papers accepted and presented at the 6th International Symposium on Modelling and Implementation of Complex Systems (MISC 2020), held during October 24–26, 2020, in Batna, Algeria. As the previous editions (MISC 2010 to MISC 2018), this symposium is intended as a tradition offering open forum and meeting space for researchers working in the field of complex systems science. This year, the MISC symposium received 113 submissions from 12 countries: Algeria, Belgium, Finland, France, UK, India, Ireland, Libya, Pakistan, Singapore, Turkey and UAE. In a rigorous reviewing process, the Program Committee selected 21 papers, which represent an acceptance rate of 19%. The PC included 112 researchers and 29 additional reviewers from 11 countries. The accepted papers were organized into sessions as follows: Internet of Things and Smart Systems, Machine Intelligence and Data Science, Cloud Computing and Networking, and Software Technology and Model Transformations.

We would like to thank the co-chairs of the Program Committee and all its members for their effort in the review process and the selection of the papers. We are grateful to the Organizing Committee members from the University of Batna 2 and the University of Constantine 2 for their contribution to the success of the symposium. Our thanks also go to the authors who submitted papers for their interest to our symposium. Enough thanks cannot be expressed to Dr. Nabil Belala for managing EasyChair system for MISC 2020 from submissions to proceedings elaboration and Dr. Ahmed-Chawki Chaouche for managing the symposium Web site.

October 2020

Salim Chikhi
Allaoua Chaoui
Abdelmalek Amine

Organization

The 6th International Symposium on Modelling and Implementation of Complex Systems (MISC 2020) was co-organized by University of Constantine 2–Abdelhamid Mehri and University of Batna 2–Mostefa Ben Boulaïd and took place in Batna, Algeria (October 24–26, 2020).

Honorary Chairs

Hacene Smadi	University of Batna 2, Algeria
Abdelouahab Chemam	University of Constantine 2, Algeria

General Chairs

Salim Chikhi	University of Constantine 2, Algeria
Abdelmalek Amine	University of Saïda, Algeria
Hamouma Moumen	University of Batna 2, Algeria

Steering Committee

Allaoua Chaoui	University of Constantine 2, Algeria
Salim Chikhi	University of Constantine 2, Algeria
Mohamed-Khireddine Kholladi	University of El Oued, Algeria
Djamel Eddine Saïdouni	University of Constantine 2, Algeria
Mohamed Bachir Yagoubi	University of Laghouat, Algeria
Hamouma Moumen	University of Batna 2, Algeria

Organizing Committee Chairs

Djamel Bellala   University of Batna 2, Algeria
Mourad Bouzenada   University of Constantine 2, Algeria

Organizing Committee

Ouahab Kadri   University of Batna 2, Algeria
Souheila Bouam   University of Batna 2, Algeria
Rima Djellab   University of Batna 2, Algeria
Brahimi Mahmoud   University of Batna 2, Algeria
Mounia Laouar   University of Batna 2, Algeria
Titouna Chafik   University of Batna 2, Algeria
Hocine Riadh   University of Batna 2, Algeria
Djeffal El-Amir   University of Batna 2, Algeria
Khamsa Djaroudib   University of Batna 2, Algeria
Alleg Mebarek   University of Batna 2, Algeria
Halaoua Madjid   University of Batna 2, Algeria
Chaabane Mezache   University of Batna 2, Algeria
Amel Behaz   University of Batna 2, Algeria
Ali Bentahar   University of Batna 2, Algeria
Farouk Zireg   University of Batna 2, Algeria
Hafiza Chaouche   University of Batna 2, Algeria

Publication Chairs

Nabil Belala   University of Constantine 2, Algeria
Ahmed-Chawki Chaouche   University of Constantine 2, Algeria

Publicity and Sponsor Chairs

Bedreddine Barguia   University of Batna 2, Algeria
Rafik Nezzar   University of Batna 2, Algeria
Ishak Benmohammed   University of Constantine 2, Algeria
Sami Barkat   University of Batna 2, Algeria

Program Committee Chairs

Allaoua Chaoui   University of Constantine 2, Algeria
Djamel Eddine Saïdouni   University of Constantine 2, Algeria
Chafik Arar   University of Batna 2, Algeria

Program Committee

Sihem Abbassen	University of Constantine 2, Algeria
Abdelkrim Abdelli	USTHB of Algiers, Algeria
Wahabou Abdou	University of Bourgogne, France
Abdelmalek Amine	University of Saida, Algeria
Abdelkrim Amirat	University of Souk Ahras, Algeria
Chafik Arar	University of Batna 2, Algeria
Baghdad Atmani	University of Oran 1, Algeria
Mohamed Chaouki Babahenini	University of Biskra, Algeria
Abdelmalik Bachir	University of Biskra, Algeria
Mohamed Batouche	University of Constantine 2, Algeria
Amel Behaz	University of Batna 2, Algeria
Ali Behloul	University of Batna 2, Algeria
Faiza Belala	University of Constantine 2, Algeria
Nabil Belala	University of Constantine 2, Algeria
Djamel Bellala	University of Batna 2, Algeria
Ghalem Belalam	University Oran 1, Algeria
Hacène Belhadef	University of Constantine 2, Algeria
Fouzia Benchikha	University of Constantine 2, Algeria
Saber Benharzallah	University of Batna 2, Algeria
Mohamed Benmohammed	University of Constantine 2, Algeria
Hammadi Bennoui	University of Biskra, Algeria
Azeddine Bilami	University of Batna 2, Algeria
Salim Bitam	University of Biskra, Algeria
Mustapha Bouakkaz	University of Laghouat, Algeria
Souheila Bouam	University of Batna 2, Algeria
Karim Bouamrane	University of Oran 1, Algeria
Djalal Eddine Boubiche	University of Batna 2, Algeria
Samia Boucherkha	University of Constantine 2, Algeria
Mohamed Amine Boudia	University of Saida, Algeria
Rachid Boudour	University of Annaba, Algeria
Mahmoud Boufaida	University of Constantine 2, Algeria
Zizette Boufaida	University of Constantine 2, Algeria
Kamel Boukhalfa	USTHB of Algiers, Algeria
Abdelkrim Bouramoul	University of Constantine 2, Algeria
Elbey Bourennane	University of Bourgogne, France
Mourad Bouzenada	University of Constantine 2, Algeria
Rachid Chalal	ESI of Algiers, Algeria
Yacine Challal	ESI of Algiers, Algeria
Ahmed-Chawki Chaouche	University of Constantine 2, Algeria
Foudil Cherif	University of Biskra, Algeria
Abdellah Chouarfia	University USTO of Oran, Algeria
Lakhdar Derdouri	University of Oum El Bouaghi, Algeria

Karim Djemame   University of Leeds, UK
Mahieddine Djoudi   University of Poitiers, France
Amer Draa   University of Constantine 2, Algeria
Khalil Drira   LAAS of Toulouse, France
Cédric Eichler   INSA Bourges, France
Zakaria Elberrichi   University of Sidi Bel Abbes, Algeria
Nadir Farah   University of Annaba, Algeria
Mohamed Amine Ferrag   University of Guelma, Algeria
Khadoudja Ghanem   University of Constantine 2, Algeria
Salim Ghanemi   University of Annaba, Algeria
Mohamed Gharzouli   University of Constantine 2, Algeria
Nacira Ghoualmi   University of Annaba, Algeria
Said Ghoul   Philadelphia University, Jordan
Zahia Guessoum   Sorbonne University, France
Gregorio Díaz-Descalzo   Castilla La Mancha University, Spain
Larbi Guezouli   University of Batna 2, Algeria
Lyamine Guezouli   University of Batna 2, Algeria
Djamila Hamdadou   University of Oran, Algeria
Saad Harous   United Arab Emirates University, UAE
Dalila Hocine   University of Tizi Ouzou, Algeria
Riadh Hocine   University of Batna 2, Algeria
Jean-Michel Ilié   Sorbonne University, France
Ouahab Kadri   University of Batna 2, Algeria
Okba Kazar   University of Biskra, Algeria
Tahar Kechadi   UCD School Dublin, Ireland
Elhilali Kerkouche   University of Jijel, Algeria
Kamel Khoualdi   University of King Abdulaziz of Jeddah,
                                    Saudi Arabia
Mohamed Nadjib Kouahla   University of Guelma, Algeria
Mohamed Tahar Kimour   University of Annaba, Algeria
Said Labed   University of Constantine 2, Algeria
Zakaria Laboudi   University of Oum El Bouaghi, Algeria
Yacine Lafifi   University of Guelma, Algeria
Ali Lemouari   University of Jijel, Algeria
Ramdane Maamri   University of Constantine 2, Algeria
Mimoun Malki   University of Sidi Bel Abbes, Algeria
Mourad Maouche   Philadelphia University, Jordan
Smaine Mazouzi   University of Skikda, Algeria
Djamila Mechta   University of Setif 1, Algeria
Kamel Eddine Melkemi   University of Batna 2, Algeria
Salah Merniz   University of Constantine 2, Algeria
Hayet-Farida Merouani   University of Annaba, Algeria
Mohammed A. Merzoug   University of Batna 2, Algeria
Souham Meshoul   University of Constantine 2, Algeria
Kamel Messaoudi   University of Souk Ahras, Algeria

Djamel Meslati   University of Annaba, Algeria
Chaker Mezioud   University of Constantine 2, Algeria
Abdelouaheb Moussaoui   University of Setif, Algeria
Mohamed Elhadi Rahmani   University of Saida, Algeria
Mathieu Roche   CIRAD of Montpellier, France
Zaidi Sahnoun   University of Constantine 2, Algeria
Maamar Sedrati   University of Batna 2, Algeria
Rachid Seghir   University of Batna 2, Algeria
Larbi Sekhri   University Oran 1, Algeria
Abdelhak-Djamel Seriai   University of Montpellier 2, France
Hamid Seridi   University of Guelma, Algeria
Hassina Seridi   University of Annaba, Algeria
Yahia Slimani   Faculty of Science of Tunis, Tunisia
Hichem Talbi   University of Constantine 2, Algeria
Noria Taghezout   University Oran 1, Algeria
Sadek Labib Terissa   University of Biskra, Algeria
Chouki Tibermacine   University of Montpelier 2, France
Chafik Titouna   University of Batna 2, Algeria
Faiza Titouna   University of Batna 2, Algeria
Belabbas Yagoubi   University of Oran 1, Algeria
Nacereddine Zarour   University of Constantine 2, Algeria
Nadia Zeghib   University of Constantine 2, Algeria
Mounira Zerari   University of Constantine 2, Algeria
Amer Zerek   Zawia University, Libya
Noureddine Zerhouni   ENSMM of Besançon, France
Abdelhafid Zitouni   University of Constantine 2, Algeria

Co-editors

Abdelmalek Amine   University of Saïda, Algeria
Allaoua Chaoui   University of Constantine 2, Algeria
Salim Chikhi   University of Constantine 2, Algeria
Mohamed Khireddine   University of El Oued, Algeria
  Kholladi
Djamel Eddine Saïdouni   University of Constantine 2, Algeria

Additional Reviewers

Barka, Kamel
Belayachi, Naima
Bensalem, Imene
Benzadri, Zakaria
Boucebsi, Rachida
Boukelia, Abdelbasset

Boulemden, Ahmed
Brahimi, Mohamed
Brahimi, Said
Djellal, Asma
Farou, Brahim
Guessoum, Meriem
Heba, Ismail
Kerdoudi, Mohamed Lamine
Kouahla, Zineddine
Mami, Mohammed Amine
Mansoul, Abdelhak
Miles, Badr-Eddine
Rhayem, Ahlem
Rahab, Hichem
Rouabhi, Miloud
Sellami, Samir
Taibi, Aissa
Yachba, Khadidja
Zarour, Karim
Zernadji, Tarek
Zitouni, Farouq

Contents

Software Technology and Model Transformations

Cloud Computing, Networking and IoT

Dynamic Replication Based on a Data Classification Model in Cloud Computing

Imad Eddine Miloudi ⓘ, Belabbas Yagoubi$^{(\boxtimes)}$ ⓘ, and Fatima Zohra Bellounar ⓘ

Laboratoire d'Informatique d'Oran - LIO, University Oran 1, Oran, Algeria
miloudi.imad@gmail.com, byagoubi@gmail.com, fbellounar@yahoo.fr

Abstract. Cloud Computing provides on demand resources for customers and enterprises to outsource their online activities efficiently and less expensively. However, the cloud environment is heterogeneous and very dynamic, storage node failures and increasing demands on data can lead to data unavailability situations leading to a decrease in quality of service. Cloud service providers face the challenge of ensuring maximum data availability and reliability. Replication of data to different nodes in the cloud has become the most common solution for achieving good performance in terms of load balancing, response time and availability. In this article, we propose a new dynamic replication strategy based on a data classification model that would adapt the replication process according to user behavior towards data. This strategy dynamically and adaptively creates the replicas necessary in order to obtain the desired performance such as, reduced response time and improved system availability while ensuring the quality of service. The solution also attempts to meet customer requirements by respecting the SLA contract. The CloudSim simulator was used to evaluate the proposed strategy and compare it to other strategies. The results obtained showed an improvement in the criteria studied in a satisfactory manner.

Keywords: Cloud computing · Dynamic replication · Classification · SLA · Cloudsim

1 Introduction

Cloud computing is a universal and practical model that provides services on demand [1, 2]. Where the customer only pays for what he consumes [3]. This paradigm provides hardware infrastructure and software as services to enterprises and users [4]. As a result, cloud computing frees the end user from the problems of finding computing and data storage capacity [5]. With this model, the client can choose the services he needs. All cloud service providers must provide the right quality of service (QoS) metrics to speed up data access, increase data availability, provide better fault tolerance and data recovery, distribute workload and minimize bandwidth consumption [6, 7]. So the Cloud faces several challenges which notably affect the quality of service. Several data management systems have been proposed, to satisfy these metrics, such as the Google File System

[8] and the Hadoop distributed file system [9], all these systems adopt a key technique which is data replication.

Replication is the process that creates different replicas of the same service on different nodes. With this technique, a data intensive application or system can achieve high availability, better fault tolerance and data recovery as well as high performance [6, 10]. Data replication is indispensable for large scale systems, especially clouds. However, like each technique, replication has many problems that must be taken into account before being able to benefit from all the advantages offered [11]. These problems generate several questions such as: What to replicate?, How many to replicate?, Where to place replicas?, When to replicate?.

Our contribution consists in proposing a model for the management of data replication in Cloud Computing. This solution aims to optimize the performance required by users in terms of availability and especially in terms of reduction of response time. The proposed solution is based on a data classification model that would adapt the replication process according to user behavior towards data. The solution also attempts to meet the replication objectives mentioned above. It has been implemented and simulated using the CloudSim simulator [12] as well as the results have been compared with other existing strategies.

The rest of the article is organized as follows: Sect. 2 cites related work. Section 3 is reserved for a detailed description of the proposed strategy. Section 4 illustrates the experimental results and the last section concludes this work and exposes future work.

2   Related Work

There are many works which study the management of replicas in the Cloud and which are based on different models, we can cite.

The authors of [13] study availability and workload problems in cloud computing, by proposing a mathematical model that calculates the minimum number of replicas in order to satisfy the availability requirements of a file and estimates the probability of blocking the nodes in a heterogeneous environment to place the replicas. Results demonstrate that the strategy meets availability requirements, access latency has been improved, load balancing and system stability have been observed.

In this article [14], the authors focused on the problem of cost in could computing, through the proposal of a new strategy called Cost effective Incremental Replication (CIR) in datacenters. This strategy applies an incremental replication approach to minimize the number of replicas while meeting reliability requirements in order to achieve cost effectiveness. In the evaluation simulation, they have demonstrated that the strategy can significantly reduce the cost of data storage, especially when the data is stored for a short period of time or is not very reliable.

In order to optimize energy consumption and communication delay, a new replication solution has been proposed in [15]. This solution used a three tiered topology: core, aggregation and access which are interconnected. This data replication technique improves communication delay and network bandwidth between geographically distributed datacenters as well as the interior of each datacenter. The performance evaluation was carried out using GreenCloud.

The author in [16] proposed a multi-objective dynamic replication strategy to find the placement and replacement of replicas. In this strategy, users can define weightings according to their own needs such as setting a higher value on expected performance. The results demonstrate that strategy can improve performance in terms of average response time, efficient use of the network, load balancing, frequency of replication and storage usage.

The authors [17] proposed an algorithm that selects the best site according to the context of the network and the data nodes which can reduce the average request time of the file. This algorithm is based on two phases, the first reduces the search in the catalog and the second proposes a selection model which envisages an overview of the defined criteria to make the best selection of the replica site.

In [18] the authors presented a dynamic data replication strategy used exponential smoothing prediction method, This strategy analyzes the access history of each file according to the number of accesses in order to predict the optimal number of replicas. This allows them to reduce the response time and the additional cost of the cloud storage system.

The authors of [19] presented a dynamic replication strategy that responds to customer needs while taking into account the benefit to the provider. This strategy is based on estimates of expenses and revenues during the execution of a query. As a result, replicas are distributed based on this cost. In addition, the placement is done with a load balancing between the regions. The results indicate that strategy can significantly increase the availability and performance of the cloud system, while taking into account the profit of the provider.

3   The Proposed Replication Strategy

Given the large volume of data stored in the Cloud as well as the unpredictable behavior of users who consult this data at widely varying frequencies, an adaptive data replication model is required. We found that there are periods when data are consulted much more than other periods, such as the summer product sales period, football championship season, occasions, national and religious holidays, …etc. Consequently, we thought of integrating into the replication model a data classification module which will replicate the data predictively.

Our contribution therefore consists in proposing a replication model based on user behavior using data classification. This model will improve the response time to requests, availability, storage capacity of nodes as well as the management of the replicas in order to satisfy the needs of the users according to the SLA contract.

To better understand the proposed approach, we will present the diagram below which gives a global view of the functionalities of the application (Fig. 1):
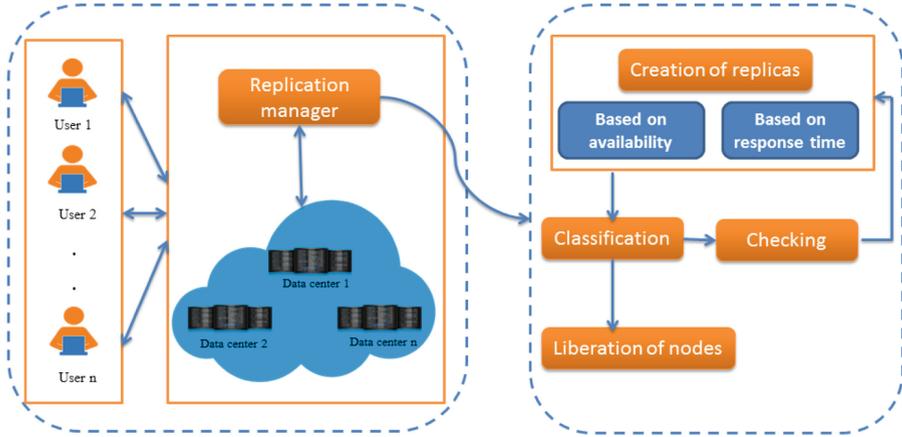
Fig. 1. Functional diagram of the proposed strategy

3.1 Creation

3.1.1 Creating Replicas Based on Availability

Initially, the proposed strategy creates replicas according to the availability required by the user. Data replication is necessary if the availability required in the SLA contract is higher than the current availability of a storage node. To calculate the minimum number of replicas that must be added to the system in order to guarantee the required availability of the file we use the following formula 1 [20, 21]:

$$A_{file\ i}\ =\ 1 - (1 - A_{nodej})^n \tag{1}$$

$$\text{Then}\quad n\ =\ \frac{\ln\left(1 - A_{file\ i}\right)}{\ln\left(1 - A_{nodej}\right)} \tag{2}$$

Where n is the number of replicas meeting the availability for a file, $A_{file\ i}$ is the availability required in SLA for a file and A_{nodej} is the availability of a storage node.

3.1.2 Creating Replicas Based on Response Time

After creating the necessary replicas to ensure availability, we will regularly test if these replicas still satisfy the SLA response time. Otherwise, this module will be triggered.

Initially, the number of replicas to satisfy the SLA response time is calculated using the following formula 3 [16]:

$$NBr_i\ =\ \frac{\sum_{J=1}^{n} \frac{size_i}{Bj}\ *\ AF(i,j)}{RT_iSLA} \tag{3}$$

Where NBr_i represents the number of replicas estimated for file i by responding to the response time specified in SLA, $size_i$ indicates the size of file i, Bj indicates the

bandwidth of node j, $AF(i, j)$ presents the access frequency of file i on node j, RT_iSLA is the response time specified in the SLA contract for file i.

After calculating this number, we are going to calculate the difference between the number of replicas, NBr_i and the number of replicas meeting availability n which is calculated by the formula 4.

$$Th = NBr - n \qquad (4)$$

If *Th* is more than zero, replication is necessary, we will then save the data with the dates of access and the number *Th*. These dates will be used to classify the data. The operation of creating response time based replicas will be repeated on a regular basis in order to avoid violation of the SLA contract. This will enrich the histories and will be used for the classification of data according to the dates of consultation.

3.2   Classification

This module consists to make a classification of data according to the behavior of the users. It represents a large part of our contribution. We are leaning on the step of classification because it will help us to create and remove replicas in a dynamic and adaptive manner. This will have like effect a profit in storage space and service on the side of customers and providers.

In this module, we have classified the data according to the time constraint. Data is classified according to the period when the number of accesses is very high. The principal objective of this module is to determine the period of time when the data does not have too much access, in order to determine its lifetime and be able to trigger the liberation module if necessary. Another objective is the assurance of response time for viral data.

3.2.1   Classified Data Types

- **Data relating to the occasion:** the data is consulted in a period less than or equal to 15 days.
- **Instant data:** the data was consulted in only one moment
- **Viral data:** the data is always consulted with a very high rate
- **Unclassified data:** the data has been consulted in different periods compared to previous years.
- **Half yearly data:** the data has been accessed in a six month period.
- **Periodic data:** the data has been consulted in a period of nine months

3.2.2   The Process to Follow

The classification of data types will be determined by following several steps:

1. **Union and intersection of dates by years:** after creating the replicas responding to the response time, we have as output the consultation dates by different years which are registered with the number of replicas. The union and the intersection of these dates give us the intervals used in the calculations below.

2. **Calculation of the number of union (Nu) and intersection days (Ni) as well as the start and end date of each:** this step will help us to determine the data membership class, specifying on the one hand, the number of days of common consultation dates for the intersection and on the other hand the number of days of all dates for the union.
3. **Calculation of NES (Number of days between end and start date of the intersection):** the NES represents the difference between the start and the end date of the intersection interval. Through this parameter, we will determine if the dates are near or not. To determine if the data will be classified or not, the P parameter of derogation has been proposed.
4. **Calculation of the derogation parameter (P):** it is a parameter that represents the number of additional days tolerated for a data to be classified. (See the following example)

The parameter "P" is expressed by the following formula:

$$P = \frac{(I[1]-U[1]) + (U[n] - I[n])}{Nu - Ni} \times 10 \tag{5}$$

With $I[1]$ is the first date of the intersection. $U[1]$ indicates the first date of the union. $U[n]$ indicates the last date of the union. $I[n]$ indicates the last date of the intersection. Ni indicates the number of days of the intersection of a data. Nu indicates the number of days of the union of a data.

To better understand the usefulness of the NES and P parameters, we will present the following example:

Tables 1 and 2 represent the dates (days) of consultation of a file fi for two years.

Table 1. The dates of consultation of file fi during year 1.

Year 1								
D1	D2	D3	D5	D7	D8	D9	D10	D60

Table 2. The dates of consultation of file fi during year 2.

Year 2								
D3	D4	D5	D7	D9	D10	D11	D60	

The following two tables represent the union and the intersection of the consultation dates for file fi during two years (Tables 3 and 4).

In order to be able to classify the data «file fi», we will first calculate the parameter NES:

NES = D60 − D3 = 57 days.

This parameter indicates to us that the dates of consultation of this file are not near, so this file may not be classified. Calculation of the derogation parameter P, can determine whether this data will be classified or not. For this we will do the following calculation:

Table 3. The intersection between the dates of consultation of file fi

Year 1∩Year 2					
D3	D5	D7	D9	D10	D60

Table 4. The union between the dates of consultation of file fi

Year 1 U Year 2										
D1	D2	D3	D4	D5	D7	D8	D9	D10	D11	D60

The number of union (Nu) is eleven and the number of intersection (Ni) is six.

$$P = \frac{(\mathbf{D}3 - \mathbf{D}1) + (\mathbf{D}60 - \mathbf{D}60)}{11 - 6} * 10 = 4 \, \text{days}$$

The value 4 represents the number of days (dates) tolerated which are nearer to intersection interval for the data to be classified. Or D60 − D10 = 50 days. D60 is not near the interval of the intersection, thus we can note that this data does not belong to a category of data consulted in a certain time, which prevents it from being classified.

3.2.3   Flow Chart of the Classification Method

In what follows, we will use the following notations:

- U: the union between the years of a data i
- I: the intersection between the years of a data i
- Ni: number of intersection of a data i
- Nu: number of union of a data i
- NES: number of days between end and start date of the intersection
- P: derogation parameter for data i (Fig. 2).

3.3   Viral Data Checking Module

Viral data is data whose access frequency is always high, this will cause a significant increase in response time. To avoid this, we used the viral data verification module in order to create new replicas if necessary.

This module is based on an algorithm that will be executed on a regular basis. its principle is summed up by the checking of response times in order not to violate the SLA contract. In this case, the module will increase the availability which will have a direct effect on the response time.
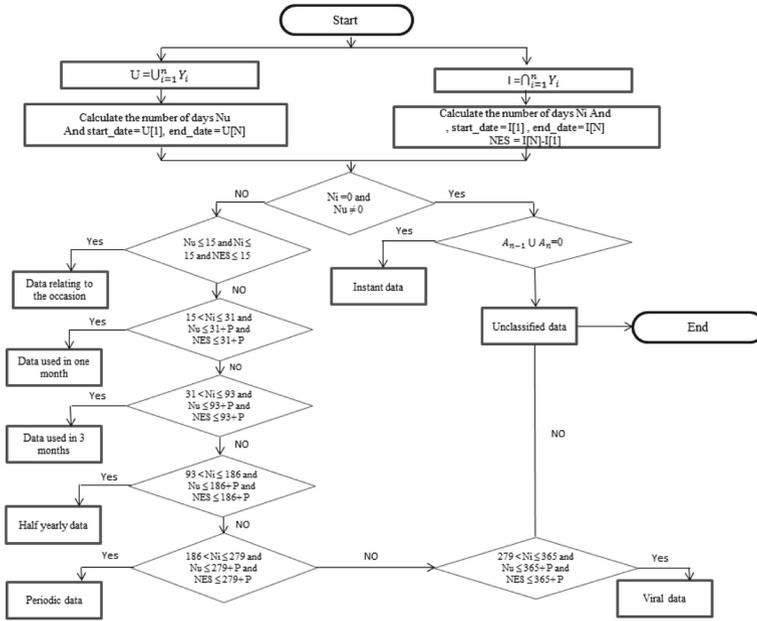
Fig. 2. File classification flow chart

Algorithm 1: Response time check for viral data

Input: List of file classes, Current response time list for each file RT, SLA parameters
Output: Replica creation.
BEGIN
while (Data is viral) **do**
  for each node i contains the data **do**
    if (Rt[i] > Tsla) **then**
    // Creation of replicas to meet response time
    end if
  end for
end while
END

3.4 Liberation

This module consists in having the created replicas deleted responding to the response time of a data which does not belong to the unclassified and viral classes. This process will result in an optimization of the storage space as well as a gain on the customer and provider side. The principle consists in calculating a liberation factor for each node having the replica in order to choose which one should delete the replicas. This factor

is calculated by the following formula:

$$LF_i = AS_j \tag{6}$$

With AS_j is the available storage capacity for a node j and LF_i indicates the liberation factor.

The deletion operation is done as follows:

1. Calculation of the liberation factor for each node storing all the replicas of the Fi file. Then sort them increasingly.
2. Suppression of replicas responding to the response time in the nodes having the smallest liberation factors.

Algorithm 2: Liberation

Input: Matrix result of the classification algorithm, Lists of nodes concerning the data.
Output: Suppression of replicas.
BEGIN
While (The data is not viral and unclassified) **do**
  if (the end date of the union = current date +1) **then**
    for each node j **do**
    // Calculate available storage space
    $AS_j = \sum_{i=1}^{n} F_j$
    // Calculate liberation factor
    $LF_i = AS_j$
    end for
   // Sort ascending (LF) ;
   // Delete the files in the nodes with the smallest LF according to the number of replicas already created by S.
   end if
  end while
  END

---

4 Implementation and Experimentation

4.1 Experimental Study

In order to validate and evaluate the behavior of the proposed replication solution, we carried out a series of experiments using the CloudSim simulator [12] which is a good environment for testing dynamic replication strategies.

In these experiments, we are interested in the following metrics: response time, number of replicas created, number of replicas deleted, and free storage space.

In our experiments, the comparison of the results is mainly carried out between four replication strategies implemented which are: the proposed strategy, CDRM [13], ADRS [16], without using a data replication.

4.2   Simulation Parameters

To carry out the various experiments, we used a certain number of simulation parameters whose values are defined in Table 5. These parameters are the same for all of the experiments presented.

Table 5. Simulation parameters

Parameter	Value
Number of data center	6
Number of storage nodes	120
Storage capacity	[30 .. 50] GB
Bandwidth of storage nodes	[90 .. 130] MB/s
MIPS of processing	1000 MIPS
Number of files	100
File size	[1 .. 3] GB
Total number of Cloudlets	[300, 600, 900, 1200, 1500, 1800]
Length of Cloudlet	[700 1000] KB

4.3   Experimental Results and Analysis

4.3.1   Response Time to Requests

In this experiment, we measure the average response time obtained with our replication approach as well as the strategies implemented. Figure 3 shows the impact of replication on request response time. These results show that with the increase in the number of requests, the response time of the four strategies also increases.

The response time of the strategy without replication is high because each file is stored in a single node, so all requests that request this file are sent to this node which causes a high load. As a result, requests are waiting in queues and this causes increased response time. On the other hand, CDRM, ADRS and the proposed strategies increase the availability of replicas which improves the response time.

From Fig. 3, we notice a significant decrease in response time by applying the proposed strategy. A gain of 66.2% is observed compared to the approach without replication, 52.4% compared to the CDRM strategy and 33.9% compared to the ADRS strategy.

Our strategy gives a better response time, because it regularly replicates files that do not meet the response time of the SLA contract, this by determining the minimum number of replicas that must be added to the system to ensure the required response time. On the other hand, in the CDRM strategy, the number of replicas created that respond to the response time is determined statically according to a threshold and does not adapt to the behavior of cloud users. Concerning ADRS, the problem of creating replicas is not treated, the creation in this strategy is done in a random way.
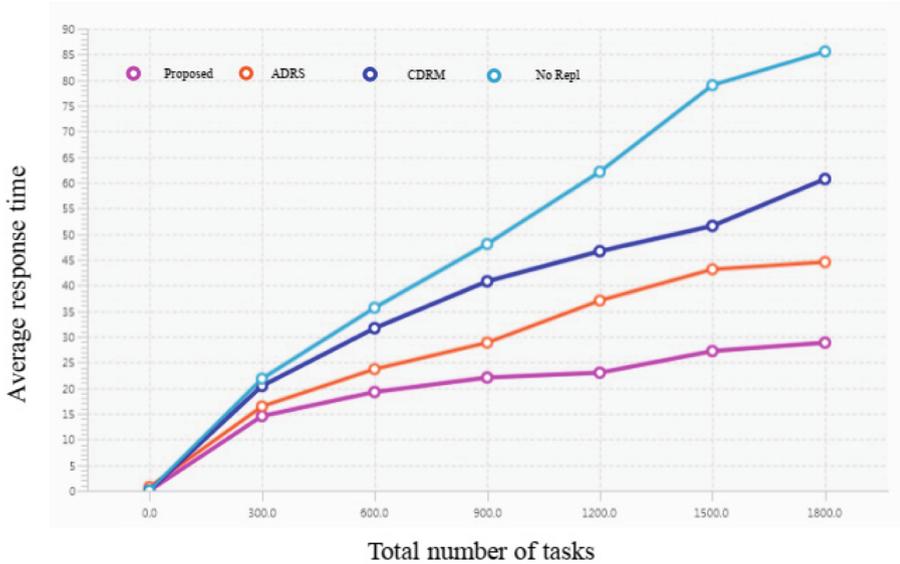
Fig. 3. Average response time based on change in number of requests.

4.3.2 Number of Replicas Created

In this experiment, we are studying the contribution of our strategy regarding the number of replicas created. We compared only three strategies because the strategy without replication does not replicate data dynamically. Figure 4 shows that the ADRS strategy
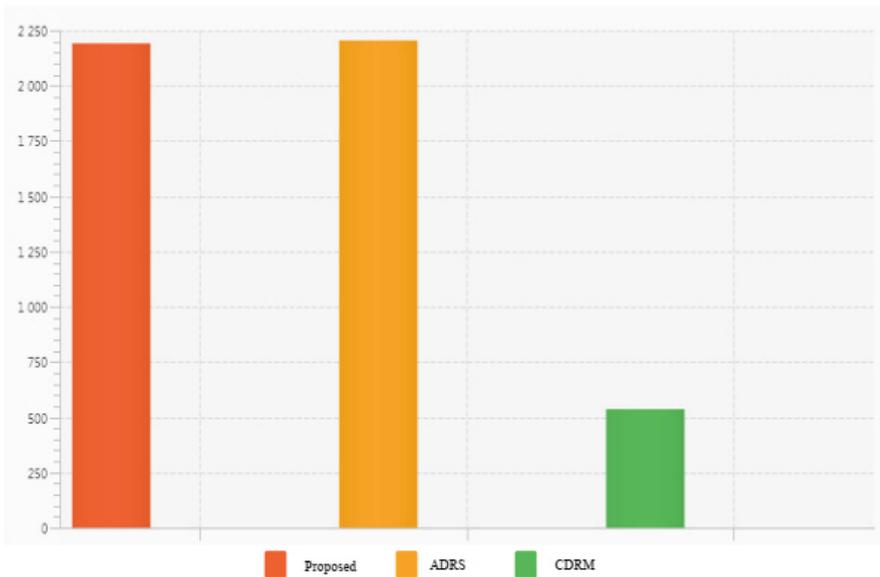
Fig. 4. Representation the number of replicas created.

creates the highest number of replicas. One would think that the proposed approach creates many more replicas because this creation is done according to two criteria namely availability and response time. However, the results show that the number of replicas created does not exceed that of the ADRS method and better meets the other QoS criteria. The number of replicas created by the CDRM method was predictable because although it is dynamic, this method does not adapt to user behavior.

4.3.3   Number of Replicas Deleted
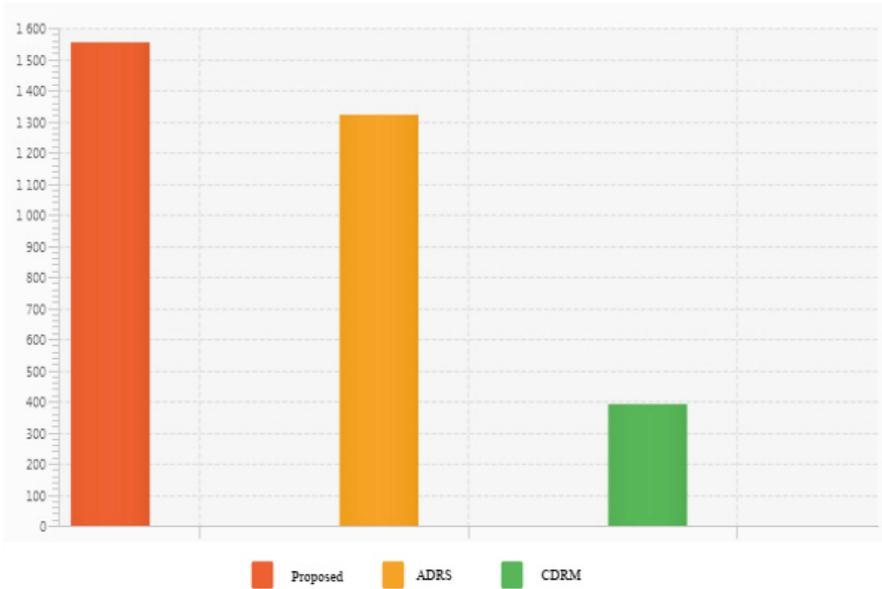
Fig. 5. Representation the number of replicas deleted

Figure 5 represents the number of replicas deleted for the three algorithms during the whole simulation. In the proposed strategy, we note that the number of replicas deleted is the highest, because it uses a liberation module which deletes the replicas when their consultation intervals have expired. Concerning ADRS, it only uses a replacement module that is triggered when the storage space is insufficient. For CDRM, it is completely logical that the number of deleted replicas should be the smallest because it depends on the number of replicas created.

After analyzing the results, we found that the frequency of creation/deletion in the CDRM method is the highest. Other methods handle data better (See Table 6).

Table 6. Number of replicas created and deleted.

Strategy	Number of replicas created	Number of replicas deleted
Proposed strategy	2193	1553
ADRS	2203	1321
CDRM	539	392

4.3.4 Storage Usage
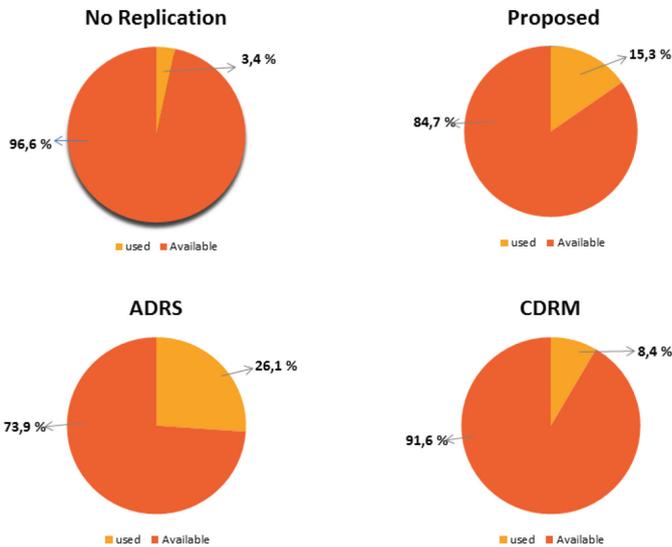
Fig. 6. Storage resource usage

Figure 6 shows the rate of space used in relation to total space. Note that the percentage of storage space used is directly related to the number and size of replicas created and deleted during the simulation, because it is not a question of replicating any request to meet the quality of service. In the proposed strategy, the variation of the data classes influences the number of replicas deleted, therefore directly on the storage space used.

The storage space used in the proposed approach is less than the ADRS approach with a difference of 10.8% and higher than the CDRM approach with only a difference of 6.9%. We can conclude that the proposed approach appropriately uses the management of replica storage in the nodes, while improving response time compared to other strategies.

5 Conclusion

Cloud Computing offers customers the most adaptable services to their needs, it is based on an economic model which allows to users to be invoiced only on the share of

resource use. However, the Cloud faces several challenges that notably affect the quality of service or the reliability of data. Replica management is one of the most important issues in the Cloud, which greatly affects the performance of the cloud system. In this article, we have proposed a new replication strategy that adapts to unpredictable user behavior, by integrating a data classification module that will replicate data in an adaptive manner. It consists of replicating the data dynamically in the nodes according to an availability required by the user, in order to improve availability and reduce response time. The proposed replication solution has four main phases which are: creation of replicas, data classification, release of storage space and verification of compliance with the SLA contract. To experiment and validate our work, we implemented and simulated our strategy under Cloudsim, with the goal of comparing the results with two existing strategies (CDRM and ADRS). The results obtained from our experiences have shown that the proposed strategy has satisfactorily improved the criteria studied.

This work has opened up some interesting perspectives which we summarize in the following points: i) integrate a replica consistency management service, ii) validate our proposal in a real cloud, iii) use the classification module for the prediction.

References

1. Mell, P., Grance, T.: The NIST definition of cloud computing recommendations of the national institute of standards and technology. J. NIST Spec. Publ. **145**, 7 (2011)
2. Senyo, P.K., Addae, E., Boateng, R.: Cloud computing research: a review of research themes, frameworks, methods and future research directions. Int. J. Inf. Manag. **38**(1), 128–139 (2018)
3. Li, K.: Power and performance management for parallel computations in clouds and data centers. J. Comput. Syst. Sci. **82**(2), 174–190 (2016)
4. Bokhari, M.U., Makki, Q., Tamandani, Y.K.: A survey on cloud computing. In: Aggarwal, V.B., Bhatnagar, V., Mishra, D.K. (eds.) Big Data Analytics. AISC, vol. 654, pp. 149–164. Springer, Singapore (2018). https://doi.org/10.1007/978-981-10-6620-7_16
5. Kofahi, N.A., Al-Rabad, A.: Identifying the top threats in cloud computing and its suggested solutions: a survey. Adv. Netw. **6**(1), 1–13 (2018)
6. Malik, S.U.R., et al.: Performance analysis of data intensive cloud systems based on data management and replication: a survey. J. Distrib. Parallel Databases **34**(2), 179–215 (2016). https://doi.org/10.1007/s10619-015-7173-2
7. Tos, U., Mokadem, R., Hameurlain, A., Ayav, T., Bora, S.: Ensuring performance and provider profit through data replication in cloud systems. Cluster Comput. **21**(3), 1479–1492 (2017). https://doi.org/10.1007/s10586-017-1507-y
8. Ghemawat, S., Gobioff, H., Leung, S.-T.: The Google file system. In: Proceedings of SOSP, pp. 29–43 (2003)
9. Borthakur, D.: The Hadoop Distributed File System: Architecture and Design. Hadoop Project Website (2007)
10. Mansouri, Y., Buyya, R.: Dynamic replication and migration of data objects with hot-spot and cold-spot statuses across storage data centers. J. Parallel Distrib. Comput. **126**, 121–133 (2019)
11. Mokadem, R., Hameurlain, A.: A data replication strategy with tenant performance and provider economic profit guarantees in cloud data centers. J. Syst. Softw. **159**, 110447 (2020)
12. Calheiros, R.N., Ranjan, R., Beloglazov, A., De Rose, C.A., Buyya, R.: CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. Softw.: Pract. Exp. **41**(1), 23–50 (2011)

13. Wei, Q., Veeravalli, B., Gong, B., Zeng, L., Feng, D.: CDRM: a cost-effective dynamic replication management scheme for cloud storage cluster. In: 2010 IEEE International Conference on Cluster Computing, Crete, Greece, pp. 188–196. IEEE (2010)

14. Li, W., Yang, Y., Yuan, D.: A novel cost-effective dynamic data replication strategy for reliability in cloud data centres. In: 2011 IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing, Sydney, NSW, Australia, pp. 496–502. IEEE (2011)

15. Boru, D., Kliazovich, D., Granelli, F., Bouvry, P., Zomaya, A.Y.: Energy-efficient data replication in cloud computing datacenters. Cluster Comput. **18**(1), 385–402 (2015). https://doi.org/10.1007/s10586-014-0404-x

16. Mansouri, N.: Adaptive data replication strategy in cloud computing for performance improvement. Front. Comput. Sci. **10**(5), 925–935 (2016). https://doi.org/10.1007/s11704-016-5182-6

17. Xue, M., Shen, J., Guo, X.: Two phase enhancing replica selection in cloud storage system. In 35th Chinese Control Conference, Chengdu, China, pp. 5255–5260. IEEE (2016)

18. He, L., Qian, Z., Shang, F.: A novel predicted replication strategy in cloud storage. J. Supercomput. **76**(7), 4838–4856 (2018). https://doi.org/10.1007/s11227-018-2647-4

19. Limam, S., Mokadem, R., Belalem, G.: Data replication strategy with satisfaction of availability, performance and tenant budget requirements. Cluster Comput. **22**(4), 1199–1210 (2019). https://doi.org/10.1007/s10586-018-02899-6

20. Sun, D., Chang, G., Gao, S., Jin, L.Z., Wang, X.W.: Modeling a dynamic data replication strategy to increase system availability in cloud computing environments. J. Comput. Sci. Technol. **27**(2), 256–272 (2012). https://doi.org/10.1007/s11390-012-1221-4

21. Meroufel, B., Belalem, G.: Dynamic replication based on availability and popularity in the presence of failures. J. Inf. Process. Syst. **8**(2), 263–278 (2012)

EECORONA: Energy Efficiency Coordinate and Routing System for Nanonetworks

Islam Amine Bouchedjera$^{(\boxtimes)}$ ⓘ, Zibouda Aliouat ⓘ, and Lemia Louail ⓘ

LRSD Laboratory, Faculty of Sciences, University Ferhat Abbas Setif-1, Setif, Algeria
{bouchedjera.islam,zaliouat,lemia.louail}@univ-setif.dz

Abstract. In the near future, the Internet of NanoThings will enable the emerging of several unprecedented applications in several fields, which are until now unimaginable, and can not be realized with traditional communication networks. Software-Defined Metamaterials (SDMs) is a promising application recently proposed in the industrial field of smart materials, where a network of nanodevices is embedded in the structure of metamaterials, to allow the latter to change their electromagnetic behavior (e.g., cloaking, filtering and steering of sound and light) at runtime. Despite the routing schemes proposed for SDMs could cope with the unique challenges in the nanonetworks, such as the very high path loss and the extremely poor data computing and storing capabilities, there is no point-to-point routing scheme that directly considers the limited stored energy capabilities. The present work proposes three versions of an adjusted flood-based point-to-point routing scheme for static and dense 2D nanonetworks. These schemes aim to enhance the pioneering routing scheme proposed by Liaskos et al. in terms of energy efficiency while keeping high communication reliability. The results of extensive simulations over various performance scenarios using nano-sim tool on NS-3 show the advantages of the proposed schemes in terms of energy consumption, successful packet delivery ratio and forwarding packet rate.

Keywords: Terahertz band · Nanonetwork · Routing · Flooding · Energy harvesting · Software-defined metamaterials · Nano-sim

1 Introduction

Nanonetworks composed of devices in a scale ranging from one to a few hundred nanometers will have the potential to play a vital role in the future by enabling the detection and monitoring of new types of events at the nanoscale. The exploiting of the nanonetwork particularities and the interconnection of them with traditional communication networks through the Internet define a novel networking paradigm called the Internet of NanoThings [2]. The Internet of NanoThings will allow opening the door to many unprecedented applications,

which are hitherto unimaginable in several fields, such as the industrial field, biomedical, environmental and military [2,18]. The need to fabricate nanocomponents for nanodevices has motivated the study of new nanomaterials. One of the most promising nanomaterials is the graphene, known as the wonder material of the 21st century [6,9,10]. On the one hand, the graphene-based nanoantennas could operate in the Terahertz band (0.1–10 THz). This band suffers from a very high propagation loss, which drastically limits the communication range of nanodevices. However, it provides large bandwidth and achievable throughput up to tens Tb/s [7,10], which allows designing simple but efficient modulation and medium sharing schemes, such as Time Spreading On-Off Keying (TS-OOK) [8]. TS-OOK, a pulse-based modulation and sharing medium scheme, which uses an electromagnetic pulse of duration Tp to transmit a bit of 1, and silence for a bit of 0. The time between transmitting two consecutive bits is Ts, which is much larger than Tp. Simultaneously, nanodevices could receive concurrently several packets by taking advantage of TS-OOK, which allows interleaving bits of different receiving packets. On the other hand, the nanonetwork may contain thousands of nanodevices or even much more, imposing a very inexpensive architecture, as well as poor hardware capabilities, especially in terms of energy storage. The main constraints in nanonetworks are the limited stored energy capabilities of nanodevices, in contrast to the continuous energy consumption to supply their components and to communicate [9]. Lately, novel energy harvesting technologies have been proposed to replenish the energy supply of nanodevices. These technologies allow nanonetworks to overcome their energy restrictions and even achieve an infinite lifetime, provided that the consumption processes and energy harvesting are jointly designed [5,6,9]. Consequently, in this study, we seek to achieve a trade-off between the lightweight requirement hardware capabilities, consumed energy and reliable communications by proposing three versions of an adjusted flood-based peer-to-peer (i.e., point-to-point) routing scheme for static and dense 2D nanonetworks. The design of peer-to-peer routing schemes in nanonetworks is still in infancy due to the aforementioned unique particularities of such network, and to the best of our knowledge, there are only two existing peer-to-peer schemes in the literature for 2D nanonetworks, that are: *i*) Deployable routing system for nanonetworks (DEROUS) [13], a peer-to-peer routing scheme deployed within static, dense, 2D nanonetworks containing identical nanodevices that are uniformly or randomly placed in a circular area, and around the nanodevice in the center of this area, namely BEACON-node. This scheme controls the retransmission process by limiting the number of forwarders on the whole network; only the nodes with good reception statistics retransmit packets. DEROUS routes the packets between the source and destination nodes along radial and circular paths. *ii*) Coordinate and routing system for nanonetworks (CORONA) [16], an addressing and peer-to-peer routing scheme, which is designed for static, dense, 2D nanonetworks. This scheme controls the retransmission process by limiting the flooding area; only the nodes located in a limited area between the source and destination nodes are permitted to forward packets. In Sect. 3, we will detail the self-assigning addresses and the routing

purposes of CORONA, since the three proposed routing versions aim to enhance the performance of CORONA in terms of energy consumption.

The remainder of this paper is organized as follows: the next section highlights the application context and the models used in the proposed schemes. Section 3 describes our contribution. Section 4 mentions the performance scenarios and discusses the results. Finally, Sect. 5 concludes the paper.

2   Application Context and System Model

2.1   Application Context

The proposed schemes are designed for nanonetworks applied in the recently proposed Software-Defined Metamaterials (SDMs) [12]. Particularly, focusing on the communication routing in nanonetworks embedded in the structure of 2D metamaterials, known as the HyperSurfaces [17]. Recently in physics, the research direction in metasurfaces has enabled the construction of new promising components with unnatural electromagnetic (EM) behaviors (e.g., cloaking, filtering and steering of sound and light) [1,12]. Despite their remarkable properties, metamaterials have been designed for a specific application under preset conditions and can not be reused. To overcome these limitations, Liaskos et al. proposed the concept of SDMs [12] that allows metamaterials to change their EM behavior at runtime, utilizing a network of nanocontrollers (sensors/actuators). Each nanocontroller interacts locally and receives globally the programmatic directives to obtain the required EM behavior for a given purpose. However, integrated controllers and sensors may need to communicate in point-to-point or multicast mode, to coordinate and maintain the correct behavior of the SDM [1].

2.2   Terahertz Band

According to the state of the art [6,10], graphene and its derivatives, can be used to fabricate novel nanoscale transceivers and antennas, which enable nanodevices to communicate at Terahertz band (0.1–10 THz). However, the latter suffers from the very high path loss, which poses a major constraint on the communication range. In addition to the spreading loss when a wave propagates through the medium such as in the low frequencies (MHz and GHz), the attenuation due to absorbed EM energy by molecules along the transmission path is a unique feature of the THz band. Where a part of the propagating wave is converted into kinetic energy at the molecule level. Jornet and al. proposed a modified Friis free-space formula for calculating the path loss in water vapor at THz, which has two parts: the spread path loss and the molecular absorption loss [7]. Molecules along the transmission path in the medium do not only attenuate the transmitted signal, but also introduce noise. According to the results obtained in [3,4], the molecular absorption defines several transparency windows, whose position, width and number depend on the transmission distance. Within each transparency window, the impact of molecular absorption loss and noise is almost non-existent. Thus, the molecular absorption becomes the dominating factor determining the communication range.

The results of Jornet et al. [4], suggest that the utilization of the first transparency window (0.1–0.54 THz) with a transmission power of 26.5 nW allows achieving a capacity of up to 10 Mbps and reliable communication up to 10 m.

2.3   Energy Harvesting Model

Due to the tiny size of nanodevices, both classical energy supply techniques (e.g., the manual recharging or replacement of a battery) and conventional energy harvesting mechanisms (e.g., solar energy, wind power, or underwater turbulences) are impracticable. Accordingly, new ways to appropriately power the nanodevice components are needed. Piezoelectric nanogenerators using Zinc Oxide (ZnO) nanowires are the most promising technologies for energy harvesting at the nanoscale [5,6,9]. The nanowires can convert vibrations (e.g., ambient vibrations or artificially generated ultrasonic waves) into electric energy. The latter is stored in a nanocapacitor to later be used in the supply of the nanodevice components. The maximum energy stored (E_{max}) in the widely used nanocapacitor is approximately 800 pJ calculated by the following formula [9],

$$E_{max} = \frac{1}{2}C_{cap}V_g^2 \tag{1}$$

where V_g of 0.42 V is the generator voltage and C_{cap} of 9 nF is the total capacitance of the nanocapacitor integrated into the nanodevice. Since energy harvesting amount is unstable, the capacitor charges relatively fast at low levels of stored energy and decelerates upon the stored energy gets closer to the maximum [9]. To model the energy harvesting of a given nanodevice, it is required to know its current harvesting cycle (n_{cycle}), given by [9]:

$$n_{cycle} = \left\lceil -\frac{C_{cap}V_g}{\Delta Q} \ln\left(1 - \sqrt{\frac{2E_{level}}{C_{cap}V_g^2}}\right) \right\rceil \tag{2}$$

where C_{cap}, V_g and ΔQ are the total capacitance of the nanocapacitor, the generator voltage and the harvested charge per cycle, respectively; while E_{level} is the current energy level of the given nanodevice. Since the n_{cycle} is known, the energy harvesting rate (λ_e) for the next harvesting cycle ($n_{cycle+1}$) in Joule/second at which the nanocapacitor is charged, can be performed as [19]:

$$\lambda_e = fV_g\Delta Q \left(e^{-\frac{\Delta Q}{C_{cap}V_g}n_{cycle+1}} - e^{-\frac{2\Delta Q}{C_{cap}V_g}n_{cycle+1}}\right) \tag{3}$$

where f is the vibration frequency of the external energy source, V_g is the generator voltage, C_{cap} is the total capacitance of the nanocapacitor and ΔQ is the harvested charge per cycle.

2.4   Energy Consumption Model

In this study, we are interested in the energy consumed by the nanodevice during the communication phase, since the latter represents the main source of energy

consumption. The energy consumption of transmitting (E_{pkt-Tx}) or receiving (E_{pkt-Rx}) a packet of N_{bit} in the THz band based on TS-OOK is given by:

$$E_{pkt-Tx} = W N_{bit} E_{pulse-Tx}$$
$$E_{pkt-Rx} = N_{bit} E_{pulse-Rx} \tag{4}$$

where $E_{pulse-Tx}$ and $E_{pulse-Rx}$ are the energy consumed in the transmission and in the reception of a pulse, respectively, and W usually set to 0.5 that represents the probability of transmitting a pulse "1" instead of a silent "0". Based on [11], in this study $E_{pulse-Tx}$ and $E_{pulse-Rx}$ is respectively set to $100\,\mathrm{pJ}$ and $0.1\,\mathrm{pJ}$ when the nanodevice communication range is $1\,\mathrm{cm}$.

3   Assigning of Addresses and Routing System

This section presents three proposed versions of an adjusted flood-based peer-to-peer routing scheme for static and dense 2D nanonetworks. The proposed schemes aim to enhance CORONA routing scheme [16] in terms of energy efficiency by reducing the number of redundant retransmissions, while keeping high communication reliability. To the best of our knowledge, this work represents the first study in the nanonetwork field that proposes and evaluates a point-to-point routing scheme taking into account the challenges posed by the high-restricted nanodevice power supply. Firstly, in Sect. 3.1, we present the coordinate geolocation address system used on CORONA scheme, because the proposed routing schemes operate on top of this system. Then, in Sect. 3.2, we will highlight each version of the proposed routing scheme.

3.1   Coordinate Geolocation Address System

The authors in [16], proposed a self-assigning address system with minimal overhead. This system operates on a rectangular area over which a large set of nanodevices is uniformly deployed, the layout may be grid or random. Four ordinary nanodevices called anchors are placed at the four vertices of this area, where the indexing of the anchors must follow a clockwise or counter-clockwise order, as shown in Fig. 1. Based on this order, each anchor sequentially, after a safe time-out sends a setup packet, with the SETUP flag set to 1 (i.e., a setup packet), the ANCHOR field set to its index and N_HOPS field set to 1 (see Fig. 2a). Each receiving node realizes via the SETUP flag that the packet serves self-assigning address purposes. This node proceeds to set its N_HOPS$_i$ corresponding to the given $anchor\_i$ to the minimum N_HOPS value over the incoming packets, then it increments N_HOPS by one and retransmits the packet. Figure 1a illustrates the process, where the $anchor\_0$ starts to send a setup packet, while the arcs show the hop-based distance between the $anchor\_0$ and each node in the nanonetwork. At concluding this process, each node in the nanonetwork sets locally its (not unique) address. The latter is composed of a set of four distance-attribute values, each corresponding to the distance between the given node and one of the

(a) Hop-based distance between the *anchor_0* and each node in the

(b) Flood area limited between the node A and the node B.
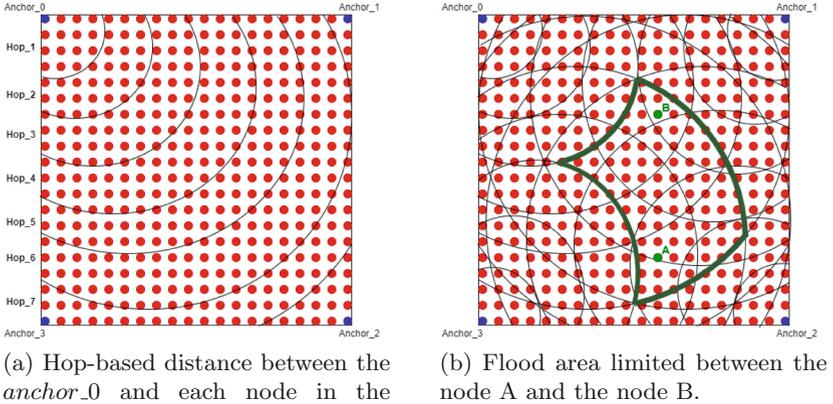
Fig. 1. Illustration of the assigning of addresses and routing system.

four anchors. Some nodes may have the same address since they are located in the same area. The number of nodes in an area depends on: *i)* the nanonetwork topology type and *ii)* the transmission power used.

3.2 Routing Phase

Based on the aforementioned addressing, the proposed schemes can route packets from the source node S with coordinates (s1, s2, s3, s4) to the destination node D with coordinates (d1, d2, d3, d4) in a flood-based manner, but only a subset of nodes whose coordinates are between those of the source and destination nodes are permitted to forward the packet. Each receiving node in the neighbor can deduce whether it should forward the received packet or not and also can determine accurately the location of the destination node, using only two of the four anchor coordinates based on ANCHOR_I and ANCHOR_J fields incorporated in the packet [16] (see Fig. 2b). Regarding CORONA routing process, the source node is in charge of assigning ANCHOR_I and ANCHOR_J fields with the indexes of the appropriate two anchors, before sending a packet. Upon receiving a packet in a node T with coordinates (t1, t2, t3, t4), the latter deduces whether it is located between the communicating nodes or not based on the criterion (5). If is the case, node T forwards the packet. Otherwise, it discards the packet.

$$(t_i \in [s_i, d_i]) \, and \, (t_j \in [s_j, d_j]) \tag{5}$$

As shown in Fig. 1b, when nodes A and B want to communicate, all nodes located in the green arc-shaped area retransmit the packets. Despite, CORONA scheme efficiently reduces the number of redundant retransmission compared to the pure flooding technique by limiting the flood area, but the latter may still maintain a large number of nodes participating in the retransmission process, leading to high-energy consumption. This sets the starting point of the present

(a) Structure of a setup packet

(b) Structure of a data packet
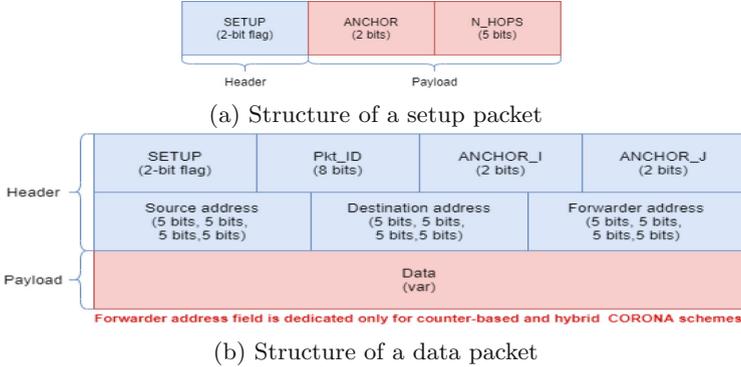
Fig. 2. Structure of the used packets

work, where we focus on reducing the number of redundant retransmissions in the limited area by applying some technique to have energy efficiency, while keeping a reasonable degree of path multiplicity to assure high communication reliability.

Energy-Based Probabilistic CORONA Routing Scheme: In this scheme, nodes that receive a packet retransmit it according to an energy-based probability. The operating process of this scheme involves the following steps, as summarized through the block scheme in Fig. 3:

1. Upon receiving a packet, each node checks whether this packet has never been treated, to avoid treating multi copies. Otherwise, the packet is simply discarded. A treated packet means that this packet has been already forwarded or discarded from this node.
2. If this packet has never been treated, the node checks whether it is located between the communicating nodes based on the criterion (5).
3. Then the selected nodes in the previous step retransmit the packet with the following forwarding probability:

$$P1 = \frac{E_{level} - E_{off} - E_{pkt-Tx}}{E_{max}} \qquad (6)$$

where E_{level} is the current energy level of the given node, E_{off} is the required energy level to power the node hardware, E_{pkt-Tx} is the required energy level to transmit a packet and E_{max} is the maximum energy level.

According to the forwarding probability $P1$, nodes with a higher energy level have a higher chance to forward the packet, and vice versa.

Counter-Based CORONA Routing Scheme: Despite the previous scheme could save energy by reducing redundant transmissions and receptions in the

limited area between the communicating nodes, its performance may suffer from the die out problem [15], which appears when some zones in this area do not receive packets, because there is no guarantee that a packet will reach all the nodes, specifically, the destination node in our case. To avoid this problem, we propose a counter-based version. In this scheme, a node decides whether to forward a packet or not by counting the number of its copies received during a random delay, i.e., Random Assessment Delay (RAD) [15]. This scheme involves the following steps, as summarized through the block scheme in Fig. 4:

1. Upon receiving a packet, each node checks whether this packet has never been treated. Otherwise, the packet is simply discarded.
2. Then the node checks if this is the first reception of the packet. If not, the node checks if the forwarding node located in its area (has the same address that incorporated in the packet, see Fig. 2b), if is the case, a packet counter redundancy is incremented by one. The latter aims to guarantee that a packet is forwarded at least a given number of times (R) in its area. We choose R = 1, to gain the minimum number of forwarding in an area; only one node is enough to forward the packet.
3. Upon the first reception of the packet, the node checks if it is located between the source S and the destination D nodes according to the criterion (5).
4. If the criterion (5) is verified, the node delays the packet until the expiration of RAD to forward it, or until receiving the intended redundancy copies R, so the packet is discarded. The node randomly chooses RAD inside the time window t, that takes charge of the peculiarities of nanonetworks, given by:

$$t = \alpha * n * T \tag{7}$$

where n is the number of nodes that have the same address with the given node, that can be simply found after the end of the self-assigning address phase by exchanging HELLO packets and α is a parameter that allows adapting to the peculiarities of the used modulation technique, TS-OOK, that permits the interleaved of packets at the receiving node. Where $\alpha = 1$ means there is no interleaved packets, such as in the traditional networks. Henceforth, $\alpha = 1/C$, in order to tradeoff between the delay, node memory usage and number of received copies, where C is the number of interleaved packets that a node can concurrently track and process. While T is the time required to receive a packet, to decode it and to decide to forward it, given by:

$$T = \gamma * T_p + (\gamma - 1) * T_s + T_l + T_d \tag{8}$$

where γ is the packet size in bits, T_p is the time to transmit an electromagnetic pulse, T_s is the time between transmitting two consecutive bits, T_d is the time required to decode and decide to forward a packet and $T_l = c/d$ is the propagation time of the last bit, where c and d are the speed of light, and the communication range, respectively.
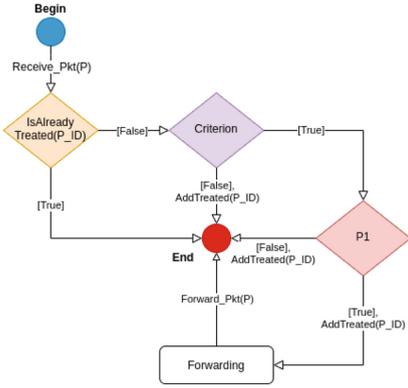
Fig. 3. Block scheme of the energy-based probabilistic routing version.
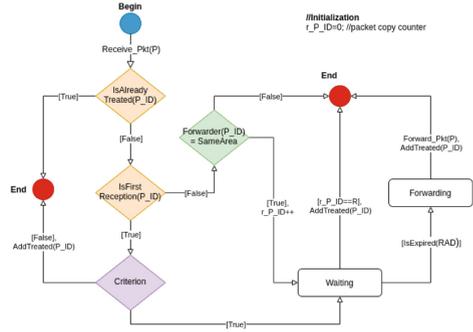
Fig. 4. Block scheme of the counter-based routing version.

Hybrid CORONA Routing Scheme: This version jointly applies counter-based and probabilistic flooding techniques on the nodes located in the limited area between the communicating nodes, to reduce the number of nodes that forward the packets, thus having energy efficiently performance. As summarized through the block scheme in Fig. 5, this scheme involves the following steps:

1. Upon receiving a packet, each node checks whether this packet has never been treated. If not, the packet is simply discarded.
2. Then the node checks if this is the first reception of the packet. If not, the node checks if the forwarding node located in its area, if is the case, a packet counter redundancy is incremented by one.
3. If this is the first reception of the packet, the node checks if it is located between the source S and the destination D nodes according to the criterion (5). If the latter is verified, the node retransmits the packet with the forwarding probability $P1$. However, nodes who forwarded a packet in the last time are exempt this time, to give them more time for harvesting energy.
4. Nodes that do not forward the packet due to the forwarding probability $P1$, they delay the packet until the expiration of RAD to try to transmit it again, with the forwarding probability $P2$, or until receiving the intended redundancy copies R of the packet, where $R = 1$, so the packet is discarded. The forwarding probability $P2$ is based on the number of nodes in an area, where nodes in an area that has a high number of nodes have a low chance of forwarding the packet, and vice versa. The forwarding probability $P2$, given by:

$$P2 = \frac{K}{n} \tag{9}$$

where K is an adjusted parameter and n is the number of nodes which have the same address with the given node. According to extensive simulations, $k = 1$ offers the best results in terms of energy consumption and successful packet delivery rate.
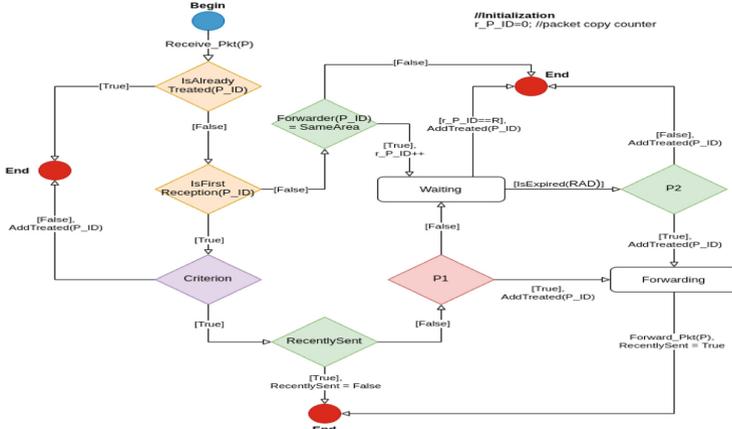
Fig. 5. Block scheme of the hybrid routing version.

4 Performance Evaluation

In this section, by means of extensive simulations on the nano-sim [14] open-source tool on NS-3, we evaluate the performance of the three proposed versions of an adjusted flood-based peer-to-peer routing scheme. In Sect. 4.1, we describe the evaluating metrics that have been considered in the performance evaluation, while in Sect. 4.2, we present the simulation setup and assumptions. Finally, in Sect. 4.3, we exhibit and discuss the obtained results.

4.1 Evaluating Metrics and Performance Scenarios

At the end of 100 cycles for each simulation, the following evaluating metrics are logged against the increasing of the packet inter-arrival time and the increasing of the nanonetwork density:

1. Packet Delivery Ratio (PDR):

$$PDR = \frac{NbrSuccPkt}{NbrPkt} \qquad (10)$$

   where $NbrSuccPkt$ and $NbrPkt$ are the number of packets successfully arrived to the destination node, and the total number of packets generated in the nanonetwork, respectively.
2. Average Ratio of Forwarders (ARF) represents the average ratio of forwarder nodes that participate for a successful packet delivery:

$$ARF = \frac{\frac{GlobalPkt}{PDR}}{NetSize} \qquad (11)$$

   where $GlobalPkt$ and PDR are the number of packet forwarding operations in the whole network per simulation and the packet delivery ratio for a given simulation, respectively. While $NetSize$ is the nanonetwork size.

Table 1. Simulation Parameters.

Parameter	Value
Frequency	0.1 THz
Pulse energy	100 pJ
Pulse duration	100 fs
β: TS-OOK time spread ratio	100
SNR	10 dB
Communicating range	1 cm
Nanonetwork size	1600
t_{cycle}: harvesting cycle time	20 ms
Packet payload size	100 bits
Node packet information queue length	20
C: number of interleaved packets	5
Scenario 1	
Packet inter-arrival time (ms)	50, 100, 150, 200, 250, 300, 350
X: spacing (cm)	0.125
Scenario 2	
Packet inter-arrival time (ms)	300
X: spacing (cm)	0.166, 0.142, 0.125, 0.111, 0.1

3. Average Residual Energy (ARE):

$$ARE = \frac{\sum_{i=1}^{NbrPkt} are_i}{NbrPkt} \tag{12}$$

where are_i is the average residual energy in the nanonetwork after the i-th communication and $NbrPkt$ is the total number of packets generated in the nanonetwork.

4.2   Simulation Setup and Assumptions

The present study considers uniform regular grid 2D topologies, with $X\,cm$ spacing, where X varies according to the corresponding performance scenario. The regular grid layout is considered due to its direct applicability to the smart metamaterial applications (e.g., SDMs), where each node represents a nanocontroller [1]. The selected layouts fill a fixed rectangular area with 1600 identical nanonodes. The harvesting energy mechanism of nanodevices is based on the vibration generated by the vents of the air conditioning system of a room, where the vibration frequency f, $1/t_{cycle} = 50$ Hz [9]. In line with [4], the nanonetwork settings and communication-related parameters are summarized in Table 1. It should be mentioned that each value in the figures of Sect. 4.3 represents the average value
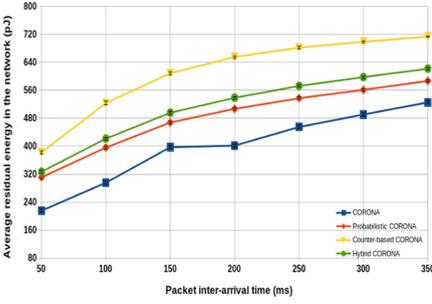
Fig. 6. Average Residual Energy imposed by the compared schemes, versus the packet inter-arrival time.
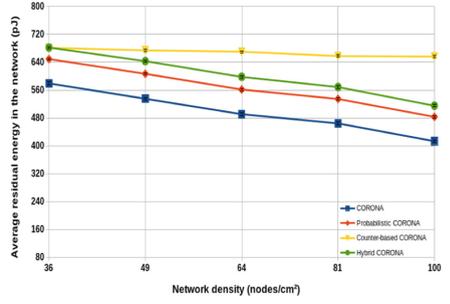
Fig. 7. Average Residual Energy imposed by the compared schemes, versus the nanonetwork density.
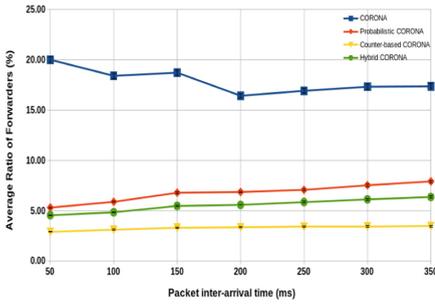
Fig. 8. Average ratio of forwarders imposed by the compared schemes, versus the packet inter-arrival time.
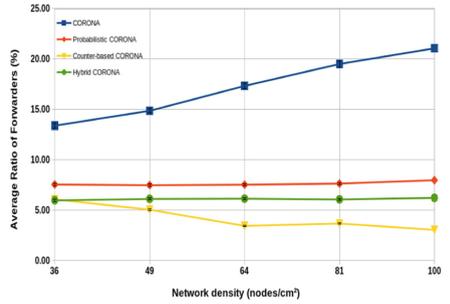
Fig. 9. Average ratio of forwarders imposed by the compared schemes, versus the nanonetwork density.

of 50 values obtained by 50 simulations. In each simulation, a series of 100 operation cycles take place, in each one, the source and the destination nodes are selected randomly. The error bars of the figures represent the 95% confidence intervals, to give greater reliability to the results obtained.

4.3   Results and Analysis

1. Varying the packet inter-arrival time
   The average residual energy in the nanonetwork, the average ratio of nodes that participate for successful packet devilry and the packet delivery ratio are presented over an increase of the packet inter-arrival time in Fig. 6, Fig. 8 and Fig. 10, respectively. Overall, the increase of the time between the transmission of two consecutive packets offers more time to the nanodevices to harvest more energy, which increases the global residual energy in the nanonetwork, thus increasing of the successful packet delivery ratio, as shown in Fig. 6 and Fig. 10. The counter-based CORONA scheme shows the best performance in terms of residual energy. Due to its routing mechanism that aims to reduce
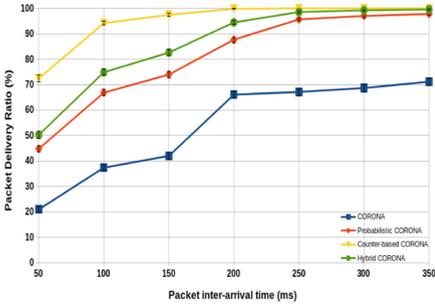
Fig. 10. Packet delivery ratio imposed by the compared schemes, versus the packet inter-arrival time.
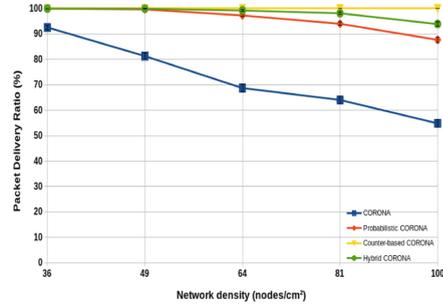
Fig. 11. Packet delivery ratio imposed by the compared schemes, versus the nanonetwork density.

the number of participating nodes for the retransmission process as confirmed in Fig. 8, where only one node in each area is enough to forward the packet. In contrast, CORONA scheme shows the worst performance in terms of residual energy, because for each communication all nodes between the communicating nodes forward the packet. While the participating rate of nodes in the retransmission process, for the probabilistic CORONA and the hybrid CORONA schemes increases at a slight rate with the increase of the packet inter-arrival time, i.e., the increase of residual energy, due to the used forwarding probability $P1$, where the higher energy level, the higher chance to forward the packet, see Fig. 8.

2. Varying the nanonetwork density

The average residual energy in the nanonetwork, the average ratio of nodes that participate for successful packet devilry and the packet delivery ratio are presented over an increase of the nanonetwork density in Fig. 7, Fig. 9 and Fig. 11, respectively. Figure 7 shows that, the larger the nanonetwork density, the more large number of nodes located between the communicating nodes, resulting in high energy consumption. For most schemes, the high energy consumption is due to the high rate of retransmission of packets and the high overhearing, as shown in Fig. 9. CORONA scheme has the worst performance in terms of the rate of nodes that forward the packet, where it has the higher rate with a significant increase over the increasing of the nanonetwork density. While the rate of nodes that forward the packet for the probabilistic CORONA and the hybrid CORONA schemes not affected with the increase of the nanonetwork density because the routing mechanism of these schemes not directly depend on the nanonetwork density, see Fig. 9. In terms of communication reliability, as shown in Fig. 11, the counter-based CORONA scheme offers a successful packet delivery ratio up to 100% regardless of the nanonetwork density. While the probabilistic CORONA and the hybrid CORONA schemes show slight decreases over the increasing of the nanonetwork density, due to the die out problem and the increase of collision rate.

5   Conclusion

The present work has supported the fact that the jointly designed of the consumption processes and energy harvesting will allow nanonetworks to achieve good performances through an infinite lifetime. The proposed routing schemes offer a good trade-off between the communication reliability and the energy consumption compared to CORONA scheme. However, the counter-based CORONA scheme shows remarkable advantages over the other schemes. Moreover, this scheme always finds the shorter path between the communicating nodes. Accordingly, we are motivated in future work to study the behavior of this scheme under rigorous performance scenarios over a nanonetwork simulator that enabling to handle a large number of nanonodes, up to millions. Furthermore, we want to make an in-depth study to find the appropriate time window size that offers a good tread-off between the successful packet delivery ratio, the end-to-end delay and the memory space usage of nanodevices.

References

1. Abadal, S., Liaskos, C., Tsioliaridou, A., Ioannidis, S., Pitsillides, A., Solé-Pareta, J., Alarcón, E., Cabellos-Aparicio, A.: Computing and communications for the software-defined metamaterial paradigm: a context analysis. IEEE Access **5**, 6225–6235 (2017)
2. Akyildiz, I.F., Jornet, J.M.: The internet of nano-things. IEEE Wirel. Commun. **17**(6), 58–63 (2010)
3. Akyildiz, I.F., Jornet, J.M., Han, C.: Terahertz band: next frontier for wireless communications. Phys. Commun. **12**, 16–32 (2014)
4. Boronin, P., Petrov, V., Moltchanov, D., Koucheryavy, Y., Jornet, J.M.: Capacity and throughput analysis of nanoscale machine communication through transparency windows in the terahertz band. Nano Commun. Netw. **5**(3), 72–82 (2014)
5. Canovas-Carrasco, S., Garcia-Sanchez, A.J., Garcia-Haro, J.: A nanoscale communication network scheme and energy model for a human hand scenario. Nano Commun. Netw. **15**, 17–27 (2018)
6. Canovas-Carrasco, S., Garcia-Sanchez, A.J., Garcia-Sanchez, F., Garcia-Haro, J.: Conceptual design of a nano-networking device. Sensors **16**(12), 2104 (2016)
7. Jornet, J.M., Akyildiz, I.F.: Channel modeling and capacity analysis for electromagnetic wireless nanonetworks in the terahertz band. IEEE Trans. Wireless Commun. **10**(10), 3211–3221 (2011)
8. Jornet, J.M., Akyildiz, I.F.: Information capacity of pulse-based wireless nanosensor networks. In: 2011 8th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks, pp. 80–88. IEEE (2011)
9. Jornet, J.M., Akyildiz, I.F.: Joint energy harvesting and communication analysis for perpetual wireless nanosensor networks in the terahertz band. IEEE Trans. Nanotechnol. **11**(3), 570–580 (2012)
10. Jornet, J.M., Akyildiz, I.F., et al.: Fundamentals of electromagnetic nanonetworks in the terahertz band. Found. Trends® Networking **7**(2-3), 77–233 (2013)
11. Jornet, J.M., Pujol, J.C., Pareta, J.S.: Phlame: a physical layer aware mac protocol for electromagnetic nanonetworks in the terahertz band. Nano Commun. Netw. **3**(1), 74–81 (2012)

12. Liaskos, C., et al.: Design and development of software defined metamaterials for nanonetworks. IEEE Circuits Syst. Mag. **15**(4), 12–25 (2015)
13. Liaskos, C., Tsioliaridou, A., Ioannidis, S., Kantartzis, N., Pitsillides, A.: A deployable routing system for nanonetworks. In: 2016 IEEE International Conference on Communications (ICC), pp. 1–6. IEEE (2016)
14. Piro, G., Grieco, L.A., Boggia, G., Camarda, P.: Nano-sim: simulating electromagnetic-based nanonetworks in the network simulator 3. In: SimuTools, pp. 203–210 (2013)
15. Reina, D., Toral, S., Johnson, P., Barrero, F.: A survey on probabilistic broadcast schemes for wireless ad hoc networks. Ad Hoc Netw. **25**, 263–292 (2015)
16. Tsioliaridou, A., Liaskos, C., Ioannidis, S., Pitsillides, A.: Corona: a coordinate and routing system for nanonetworks. In: Proceedings of the Second Annual International Conference on Nanoscale Computing and Communication, pp. 1–6 (2015)
17. Tsioliaridou, A., Liaskos, C., Pitsillides, A., Ioannidis, S.: A novel protocol for network-controlled metasurfaces. In: The 4th ACM International Conference on Nanoscale Computing and Communication, pp. 1–6 (2017)
18. Wang, P., Jornet, J.M., Malik, M.A., Akkari, N., Akyildiz, I.F.: Energy and spectrum-aware mac protocol for perpetual wireless nanosensor networks in the terahertz band. Ad Hoc Netw. **11**(8), 2541–2555 (2013)
19. Xu, J., Kan, J., Zhang, Y.: Centralized energy harvesting-based tdma protocol for terahertz nanosensor networks. Sensors **19**(20), 4508 (2019)

Communication-Flow Privacy-Preservation in 6LoWPANs-Based IoT Networks

Asma Iman Kouachi$^{(\boxtimes)}$ and Abdelmalik Bachir$^{(\boxtimes)}$

LESIA Laboratory, Computer Science Department, Mohamed Khider University, Biskra, Algeria
{a.kouachi,a.bachir}@univ-biskra.dz

Abstract. An Internet of Things (IoT) network is a worldwide network connecting billions of devices with each other over the Internet. IPv6 Low power Wireless Personal Area Network (6LoWPAN) is an enabling technology for a complete end-to-end IoTs architecture. A 6LoWPAN contains small devices that communicate with other small devices or other powerful devices over the Internet. Communication privacy is a feature that allows the communicating users to make sure that there is no other entity that is spying on them. As such, it is a very important feature for the success of IoT applications. While traditionally privacy has been dealt with by hiding the content of the data content exchanged between the communicating pairs through the use of encryption, it has been shown that this cannot be sufficient in many application scenarios. It has been shown that metadata information such as the ones included in packet headers for the sake of the operation of communication protocols, such as IP addresses, reveal information such as the identities of the communicating pairs, which in some applications is considered critical information.

While there are many surveys in the literature dealing with communication privacy in the IoT, to our knowledge, little has been done on communication-flow identifiers privacy preservation in 6LoWPAN-based IoT networks. In this survey, we thoroughly expose the prime focus of the existing solutions on communication identifiers privacy in 6LoWPANs, clarifying the important information about: at which layer solutions operate, based on which protocol, against which attack, for which application, based on simulations or real prototypes, which sensitive information or communication identifiers are protected, which Privacy-Preserving Technique (PPT) is used, and how long is the duration of the protection against privacy attacks. We provide a comprehensive coverage on all of these aspects focusing on the main design guidelines that drove existing solutions while showing their merits and shortcomings.

Keywords: Internet of Things (IoT) · 6LoWPAN · Privacy · Anonymity · Pseudonymity · Unlinkability

1   Introduction

The basic idea of IoT technology is to enable the interaction of many devices (things) with each other for enhancing users' life and experience. The thing in the Internet of Things means an embedded computing device that can send and receive information in the network. A thing relies on an embedded system that is based on micro-controllers with small memory utilization and has very limited memory, computation, and energy capacities [40]. Figure 1 clarifies several IoT applications, including smart environment, electronic healthcare, industrial applications, agriculture, surveillance, military applications, etc. According to the website Statista [34] (from 2015 to 2025), the number of IoT-devices in the world will increase from 30.7 billion in 2020 to 75.44 billion in 2025. Thus, with this growth, IoT challenges such as security and privacy will pose more serious issues than ever before. Privacy is one of the most prominent issues for the majority of IoT applications. How to preserve privacy is still a open problem, especially in sensitive applications such as electronic Healthcare (e-Health).
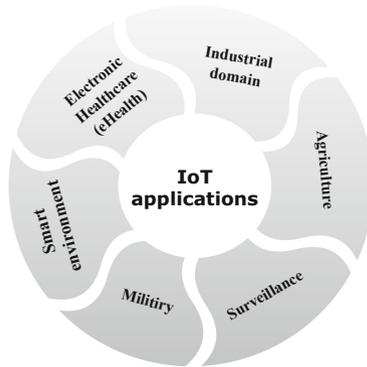
Fig. 1. IoT applications.

According to [27], there are five kinds of sensitive information in the IoT: identity, location, device type, time, and data. For each kind, there are specific privacy techniques. Thus, privacy is not only limited, for an attack can discover the device's information like the device's name or its identifier [31]. This attack can also infer the device's place, with whom it communicates? The device's purpose of communication, and its service anytime and anywhere. Communication breached privacy problem is that a traffic analysis attack can infer that two entities are communicating with each other (with whom it is communicating?). This problem is centered around the persistence of the five flow information/communication identifiers which are source and destination IP addresses, source and destination port numbers, and transport mode in packet headers, and metadata or network properties as size, time, sequence numbers, etc. Communication identifiers privacy can be achieved when the anonymity (source anonymity

and destination anonymity) and unlinkability (source-destination unlinkability) properties are achieved. In other words, if the flow information are anonymous and unlinkable, the communication will be anonymous and unlinkable automatically, and the traffic analysis attack can not link the communication entities (the source and the destination), and vise versa.

Preserving communication privacy in the IoT is harder than in IP communication in the Internet [10], because IoT-devices have many constraints on their limited-resources in energy and computing power (must use lightweight computation) for cryptography or others computations.

In this survey, we collect the recent 6LoWPAN-based solutions for preserving communication identifiers privacy, and analyze them by extracting the most important information which are: (1) on which layer this solution works?, (2) what is the protocol that this solution is based on?, (3) what is the kind of attacks that this solution is powerful against?, (4) to which application this solution applies?, (5) with which simulator that this solution was validated?, (6) what are the sensitive information/communication identifiers that this solution considers?, (7) what is the Privacy-Preserving Technique (PPT) in use?, (8) for how long does this solution keep privacy preserved?. The main goal of our survey is to help researchers to propose an efficient 6LoWPAN-based solution in the future for achieving communication identifiers privacy.

The rest of this survey is organized as follows. An overview of 6LoWPANs is presented in Sect. 2. In Sect. 3, we present privacy definitions, a reference model of privacy, communication privacy-preserving techniques, and anonymous communication privacy in the IoT. We give recent 6LoWPAN-based communication identifiers privacy main solutions in Sect. 4 followed by a summary in Sect. 5. Finally, conclusions are given in Sect. 6.

2 IPv6 Low Power Wireless Personal Area Network (6LoWPAN)

Although the use of IPv6 allows the addressing of a very large number of objects, the length of its header can be excessive for its use with the IEEE 802.15.4 radio standard. Therefore, a compression mechanism for headers of different layers (transport and network) is needed to minimize header's size and maximize the rate of effective data transfer [29]. To this end, the Internet Engineering Task Force (IETF) has developed an adaptation layer for constrained objects that have limited battery, processing, memory, and storage resources which is called 6LoWPAN layer. The main role of this layer is to make IPv6 addresses adaptable to IEEE 802.15.4 protocols. This requirement is challenging because there is a difference between both technologies, particularly the difference between the MTU size of IPv6 packets which is about 1280-byte length, and the MTU size of IEEE 802.15.4 packets which is about 127 bytes [10,29]. 6LoWPAN layer is located between the network and data link layers. As shown in Fig. 2, a 6LoW-PAN contains a set of devices and routers connected to the Internet by a 6LoW-PAN Border Router (6LBR) acting as a gateway. 6LBR is also called LBR.
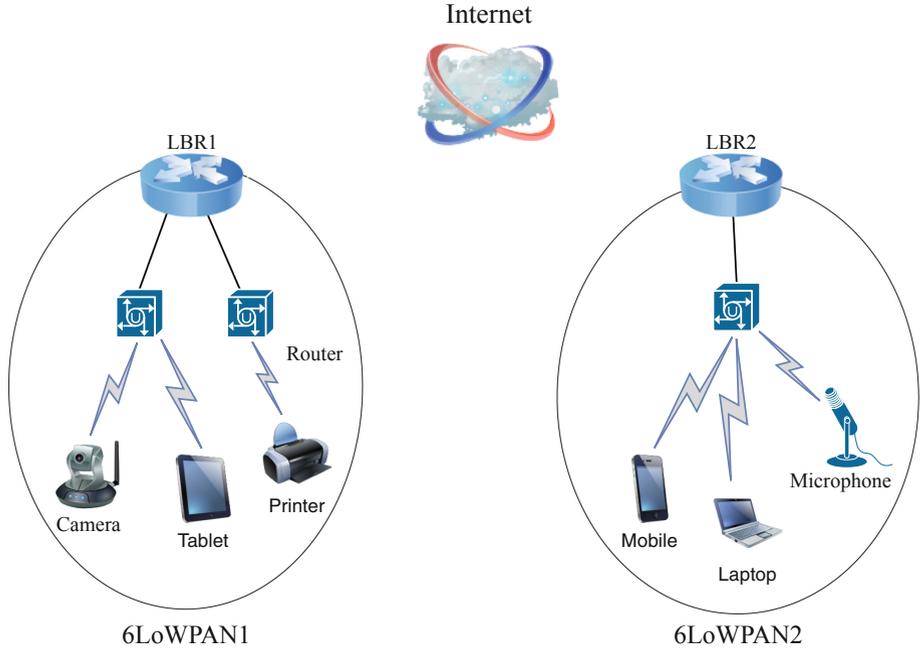
Fig. 2. Organization of an IoT network based on 6LoWPAN. Each 6LoWPAN may have many devices connected to intermediate routers which are in turn connected to the global Internet through a 6LBR gateway.

3   Communication Privacy

Privacy is an important issue for most IoT applications, particularly those which are deemed sensitive such as healthcare, and military applications. For instance, in healthcare applications, an attacker can infer a lot of sensitive, personal, medical, and vital information about a patient [3,26,28,39]. In [28], all IoT applications need to preserve privacy, but sensitive applications like healthcare applications have to preserve more patients' privacy, and their communications too. The privacy can be a part of security in some papers like in [4,5,16].

In this section, we provide some definitions on privacy in Sect. 3.1, and the communication privacy preservation techniques are in Sect. 3.2. Finally, anonymous communication privacy in the IoT is the subject of the last Sect. 3.3.

3.1   Privacy Definitions

There are many definitions of privacy, and they are different according to which where they are used. Generally, privacy-definitions are the following.

– According to [20], the user has the right to control their personal information such as name, address, age, job, etc., and prevent its disclosure to unauthorized people.

– In [9], privacy is the user's right to be alone without anyone watching or disturbing them.
– The user has the ability to select when, how, and what their personal information that are sharing with others. In addition, the user has to control their personal information too, i.e. select who can access their personal information and who can not [1,9,19,31].
– In [35], privacy is the unwillingness to share any personal and sensitive information with others. It can be equivalent to the following expression "It's none of your business".
– According to [11], achieving privacy also requires hiding sensitive header information such as source and destination IPv6 addresses, source and destination port numbers, as well as the transport mode used.

In communication identifiers privacy, traffic analysis attack aim to sniff traffic during the communication. These attacks collect the network information/communication identifiers (source and destination IP addresses, source and destination port numbers, and type of transport protocol) by extracting and analyzing the input traffic and the output traffic. The attacks aim at disclosing the communicating pairs and can be used in combination with other attacks to achieve higher privacy breaching.

3.2   Communication Privacy Preservation Techniques (CPPT)

There are two important techniques for preserving communication identifiers privacy which are:

– Data perturbation techniques: according to [19], its goal is focusing on hiding or modifying the sensitive information, these techniques include:
  • Noise addition techniques: or randomization techniques. They aim to add noise to the sensitive information in order to disorganize the communication. With the randomization technique, the linkability between communication pairs will be hard to achieve [32],
  • Anonymization techniques: is the most important technique for preserving privacy. It focuses on hiding the original identities [1] with protecting the accuracy of information [9]. There are many ways to achieve anonymity using one of the following anonymity tools in [32] as follows:
    ∗ Generalization: is based on an indirect way to preserve privacy. For example, to preserve the personal information of the user such as birthday date, to know the user's age range, we can approximate his age range through the information with which he created an account instead of to his birth date directly. Note that this technique is better when we want to preserve data privacy not communication privacy. In other words, generalization technique is used with identifiers (device's name, device's type, etc), not identities (device's IP address, device's MAC address)

  * Multiple identities: working on identities rather than on identifiers
    enhances further communication privacy. Using many identities makes
    the linkability between identity and other items difficult. Multiple
    identities technique include:
    • Pseudonymization: typically, this technique focuses on identifiers
      rather than identities although it can also be used with identities
      by using fictitious identities during the communication instead of
      the real identities.
    • Digital Identity Management: the goal of this technique is simi-
      lar to the previous one. This technique focuses on two things as
      follows: (i) generating many identities as pseudonyms, and (ii)
      managing those pseudonyms (when the user has to use them), i.e.
      use the first pseudonym with Facebook, the second with Google,
      etc.
  * Communication obfuscation: obfuscation is hiding information by
    adding fake information to the original one. It can be used, for exam-
    ple, to degrade the quality of information. Thus, this solution impacts
    on the accuracy of information negatively [9]. Typically obfuscation
    focuses on preserving the privacy of meta-data privacy (such as source
    and destination IP addresses, source and destination port numbers,
    as well as transport protocol). It can be achieved through one of the
    following techniques:
    · Virtual Private Network (VPN): is an approach based on a proxy
      to achieve anonymity. The proxy here is the VPN Gateway. At
      the endpoints of the VPN, the traffic analysis attack can infer all
      metadata. Thus, communication privacy is not completely pre-
      served as correlation between traffic entering and exiting the VPN
      can be made.
    · Proxy server: there is an intermediate entity, i.e. proxy between
      the sender and the receiver. To preserve sender anonymity, the
      proxy communicates with the receiver by its IP address instead
      of the sender's IP address. So, the proxy hides the IP address of
      the sender like in the anonymizer [33].
    · Mix-Networks (Mix-Net): is extended of the previous technique.
      This technique includes a set of proxy entities (mixes) instead
      of a single proxy entity. Use a random path from the sender to
      the receiver to provide preserving privacy. Mix-Net focuses on
      preserving two properties of privacy in the sender and receiver
      sides, which are the anonymity and the unlinkability. Its main
      idea is taking a set of messages in order to scramble/delay/re-
      encode them to make the tracking of the communication flows
      difficult to the traffic analysis attack [31],
    · The Onion Routing (TOR): is based on Mix-Net. TOR is com-
      monly referred to as Tor and is a popular protocol for anonymiza-
      tion of communication in the Internet. Anonymization is done by

hiding the identity of communication pairs (the sender and the receiver) [25].
– Protecting header information: according to [11], there are two techniques to protect header information which are:
  • Temporary Stateless Address Auto-configuration: focuses on modifying the interface identifier (IID) of the IPv6 address (the IPv6 address contains prefix field and IID field) from time to time. Temporary address needs to change prefix and IID fields to achieve better privacy.
  • Cryptography Generated Addresses (CGA): this technique is proposed to prevent spoofing or stealing attacks to get the IPv6 address.

3.3 Anonymous Communication in the IoT

According to [36], anonymous communication in the IoT is a major challenge. The traditional anonymous communication systems need big computation power and large bandwidth, and the IoT-devices have limitations in power, computation, processing, memory, and bandwidth. There are two kinds of anonymous communication solution in the IoT [36]:

– Anonymous communication solutions based on computation offloading: the main idea of these solutions is using the traditional anonymous communication solutions at the gateway (like 6LBR),
– Anonymous communication solutions based on lightweight cryptography: this kind is also divided into two types:
  • Anonymous communication based on identity encryption,
  • Anonymous communication based on pseudonym encryption.

4   Main Contributions

Preserving communication identifiers privacy is a very important property for the success of IoT applications. However, traditional solutions cannot apply directly to 6LoWPANs due the constrained capabilities of devices (memory, computation, power) [14, 27, 40]. In this section, we touch upon the main solutions for preserving communication identifiers privacy in 6LoWPANs considering the most important features, i.e (1) in which layer this solution operates? to know where the communication privacy will be achieved, (2) what is the protocol that this solution is based on? the solution can be based on a traditional protocol or it is a new one itself, (3) what is the kind of attacks that this solution is powerful against? there is not any solution which is powerful against all communication privacy attacks, (4) to which application this solution apply?, some solutions are specific to applications whereas others are not, (5) what is the simulator used for solution validation? popular general simulators such ns-2, and ns-3, as well as IoT-specific simulators such as Cooja, TinyBLE, have also been used, (6) what are the sensitive information that this solution focuses on? the sensitive information are source and/or destination MAC, IP addresses and/or port numbers, (7) what is the privacy-preserving technique in use?, and (8) how long does a given solution lasts? the smaller this parameter, the better the solution. In the following, we provide brief descriptions of main contributions.

Tor for Smart Home. The original Tor protocol for privacy protection in the internet has a limitation about the transport protocol. It works only with TCP connections. To integrate the Tor into the IoT technology, the authors of [13] used the Tor protocol for the smart home applications in order to anonymize the communication. The authors set a gateway that runs Tails (The Amnesic Incognito Live System) [8]. Tails ensures that all connections pass through the Tor, and no MAC address spoofing, i.e. no traffic tracking based on MAC addresses. It is to be noted that this solution (Tor in Smart Home) did not address the context of 6LoWPANs and has only been tested with IPv4 addresses. Therefore, its applicability in 6LoWPANs may need major modifications.

Generating IPv6 Pseudonyms. In [37], the authors propose a lightweight IPv6 address auto-generation algorithm in 6LoWPANs with the aim of preserving communication privacy without introducing additional headers. In particular, they consider that each 6LoWPAN has one LBR and a few clusters. For each cluster, there is a cluster head (CH) acting as Access Router (AR) and some cluster members (CM) acting as terminals. The structure of IPv6 address in this paper has three fields: (a) global routing prefix, (b) cluster ID, and (c) member ID, i.e. hierarchical addressing. A CM gets its IPv6 address by three-step procedure: (i) its prefix is obtained from the LBR, (ii) its cluster ID is obtained from its cluster head (AR), and (iii) its member ID is generated by itself by picking a member ID randomly from its pool for each time window. With this solution, there is a complexity at the CH as it is responsible for allocating and sending the information about address changes to all its CMs. In addition, CMs are the only ones that change their member ID. A problem with this solution is that it keeps using the same MAC address. Therefore, even when a CM changes its IPv6 address every time window, it keeps using the same MAC address, and thus could be used by an attacker to link changing IPv6 addresses together from their corresponding MAC address.

Using MAC Pseudonyms (CryptoCop). In [30], the authors identified MAC addresses as a vulnerability that limits the use of pseudonyms at the routing layer. They considered the situation of Bluetooth Low Energy (BLE) protocols in addition to 6LoWPANs. In fact, in these protocols devices keep using the same MAC address even when pseudonyms are used at the IP layer. As a solution, they proposed to change the source MAC address for each frame to preserve source anonymity thereby achieving sender-frames unlinkability. However, their solution still suffers from receiver-frames linkability as it only achieved source anonymity. In addition, it is not clear how does the sender inform the destination about its new MAC address.

Changing IPv6 and MAC Address Each Time Window. In [2], the authors proposed a new neighbor discovery extension called 6LoWPAN-ND with the goal of minimizing the overhead of operations of the traditional neighbor

discovery scheme and to change IPv6 and MAC addresses for each time window. According to the traditional architecture of 6LoWPANs consisting of a LBR, ARs, and devices, their scheme relies on the following operations. To get a new MAC address, a device starts by sending a request to its corresponding AR with its information (device ID, nonce, etc.). The AR forwards the request to the LBR which selects a new MAC address from its pool and sends the reply back to the AR with device information, which in turn forwards the reply back to the device. Once the device receives its new MAC address it makes use of it to create a new IPv6 address. Although this solution provide a certain level of anonymity through the use of pseudonyms both the link and routing layers, it has a non negligible overhead caused by the number of messages exchanged between the device, the AR and the LBR. In addition, it creates pressure at the LBR due the centralization of the operations of MAC assignment to the devices. Moreover, the solution does not seem to has effect on those destination nodes located outside the 6LoWPAN (e.g.. a destination located in the global internet).

Mutual Change of Source and Destination MAC Addresses. In [6], the authors proposed a scheme called Ephemeral to hide both source and destination MAC addresses of devices within a 6LoWPAN at each time window. Their proposal makes some assumptions and operates as follows. Each device has two keys K_1 for encrypting the payload of the frame, and K_2 for generating new MAC addresses. The generation of new source and destination MAC addresses called nsMAC and ndMAC respectively relies on the use of cryptographic functions. Specifically, the use: (1) a random value r which is on 14B or 15B long, (2) a counter c which is 1B long (there are two kinds of the counter, one for the source s_c and other for the destination d_c), (3) a key K_2, and (4) the initial source and destination MAC addresses called sMAC and dMAC, respectively. Each node knows the MAC address and the initial r of its neighbors. In addition, the updated r will be sent over Internet Control Message Protocol (ICMP) messages for a time window. The source entity has to generate the two new MAC addresses nsMAC and ndMAC to replace the initial source and destination sMAC and dMAC MAC addresses. After that, the source entity adds two fields (2B long) to the frame header which are s_c and d_c, and sends the frame to the destination entity which needs to decrypt the frame header to infer the original source and destination MAC addresses sMAC and dMAC. Although Ephemeral provides source and destination MAC anonymity, it has some limitations as it adds extra overhead (2B field in the header). In addition, it is not clear how it operates for flows going outside the 6LoWPAN where destination nodes can be located anywhere in the global internet.

Changing All Communication Identifiers. In [41], the authors propose uMT6D which is an adaptation of the traditional MT6D [7] for the IoT technology. uMT6D changes the source and destination IPv6 addresses for each time window, as well as source and destination MAC addresses and port numbers to prevent the network traffic attack from linking the communication between two

entities. Rotating addresses is based on a lightweight hash function. uMT6D encapsulates the original encrypted IPv6 packet into the new uMT6D packet with new addresses. Rotation, encryption, and encapsulation addresses can be done in the gateway or at the device. uMT6D added a new header to the original IPv6 header to preserve communication privacy which leads to more energy consumption and increased latency.

Using Lightweight IDs. In [38], the authors proposed a solutions called Communication Security and Privacy (CSP) support for 6LoWPANs, which relies on the use of lightweight IDs instead of IPv6 addresses. CSP defines three types of IDs: Permanent ID (PID), Session ID (SID), and Temporary ID (TID). Each node in the 6LoWPAN communicates with its 6LBR to get its PID and SID, using the source address TID as a MAC address, and the destination address is 6LBR PID. The 6LBR maintains a mapping table with correspondances between PIDs and SIDs. There are two cases according to the location of the destination which are: (i) the source and the destination are in the same 6LoWPAN network, (ii) the source and the destination are in different 6LoWPAN networks. While this solution aims at hiding all communication identifiers, the use of IDs instead of IPv6 addresses may cause compatibility issues. In addition, there is a need for a coordination protocols between routers for the updating of the changing IDs.

NAT-Inspired Solutions. In [17], the authors proposed a solution called new Advanced Persistent Threat (APT) to increase communication identifiers privacy between two entities inside Autonomous Systems (ASs). Their main idea is similar to the technique used by Network Address Translation (NAT) as they define two types of addresses: public (used for external communication toward another AS) and private (used for internal communication within the same AS). The Border Gateway (BR) is an important part of this solution as it has the role of ensuring address translation from and to external ASs. As such it is considered as the most vulnerable point in the entire architecture. In addition to this weakness, APT did not envisage MAC address change so communication privacy within the local network might not be preserved. Moreover, the way the mapping is performed by the BR makes it vulnerable to traffic correlation attacks at the BRs.

Using Tor for UDP Communications (Tor-UPD). In [24], the authors proposed to use Tor to provide communication identifiers anonymity in 6LoW-PANs. However, due to the limited computation, storage, and energy resources, most of communications in 6LoWPANs are based on UDP. Therefore, they proposed to modify existing Tor operations, which works on TCP traffic only, to be compatible with UPD communications. They defined IoT Onion Routers (IORs) to anonymize the source and destination of communications pair as well as port numbers. As other solutions, it is not clear to which device class this solution can apply as it does not seem to operate on very constrained devices. In addition, there is no changing of MAC addresses which makes the solution vulnerable to local attackers.

Using One Time Address (uOTA). In [15], the authors proposed Micro One Time Addresse (uOTA) which is a solution that aims to prevent both (i) insider attacks (attacker situated in the same cluster or in the same 6LoWPAN with the communication entities), and (ii) outsider attacks (attacker situated outside the 6LoWPAN) from de-anonymizing the communication between the two communicating entities and linking the relationship between them. The main idea of uOTA is mainly inspired from OTA [18] and adapted to operate in constrained 6LoWPANs. uOTA uses one IPv6 address to send or to receive just one packet as well as port numbers. As other similar proposals, uOTA has some shortcomings as it (i) introduces extra delay due to IPv6 translations, (ii) the coordination between routers has to be done, and (iii) uOTA does not envisage changing MAC addresses which makes it vulnerable to local attackers.

Using Congruence Classes to Allocate Addresses. In [21], the authors proposed a solution called Privacy-enabled Disjoint and Dynamic Address Auto-Configuration (PeDAAC) protocol for 6LoWPANs with the aim of allocating MAC and IPv6 addresses dynamically without the need for a Duplicate Address Detection (DAD) protocol while achieving anonymity and unlinkability of communication as well as location privacy against spatial and temporal correlations. The main idea of PeDAAC is based on the use of a congruence classes which have interesting mathematical properties in creating non duplicate addresses, the authors did not explicitly show how the coordination between new addresses is achieved such as the creation of AR-UID and node-UID fields in the structure of IPv6 addresses. In addition, as shown in [22], their solution is not resistant against attackers situated within the same cluster. Moreover, PeDAAC makes assumptions that layer 2 messages exchanged are secure without explicitly showing how this is achieved.

Using Tor for MQTT Protocol (MQTT-Tor). In [24], the authors propose to solve the problem of communication identifiers for the Message Queue Telemetry Transport (MQTT) protocol stack based on Tor, called MQTT-Tor. In the traditional MQTT protocol, there is a broker acting as a server for publishing or subscribing topics through clients. In MQTT-Tor, a set of brokers is set as Tor routers and clients (devices) play the role of Tor clients. The main limitations of this solutions is that is does not apply to very constrained devices and does not cope with local attackers as MAC addresses are not changed.

Delegating Tor Operations (Tor-Delegation). In [12], the authors showed that the original Tor cannot be applied directly to IoT networks dues to the constrained nature of IoT devices and the incompatibility of protocols. They proposed to delegate and offload those sophisticated encryption operations to a router or a webserver of the device owner which would act as Onion Router (OR) entry node. The delegation server established the Tor circuit during the connection step and crypts the data, the source and destination IP addresses

Table 1. Summary of main communication flow privacy protection solutions

Approach	Year	Layer	Based on	Against attack	Application	Simulator or tool	Sensitive information	PPT	Duration
Tor-smart home [13]	2014	Network and MAC	Tor	Outsider attack	Smart home	Experimental e.g. on Skype	sIPv4, dIPv4, sMAC, dMAC	Tor	Peer session
APS [37]	2015	Network	IPv6	Eavesdropper attack	Smart home	ns-2	sIPv6	Randomization	Time window
CryptoCop [30]	2016	MAC	BLE	Eavesdropper attack	Heart rate prototype	TinyBLE	sMAC	Cryptography	Peer packet transmission
6LoWPAN-ND extension [23]	2016	Network and MAC	6LoWP-AN-ND	Eavesdropper attack	Unspecified	ns-3	sMAC, dMAC sIPv6, dIPv6	NAT proxy	Time window
Ephemeral [6]	2016	MAC	IEEE 802.15.4	Traffic analyzing	Unspecified	Cooja and WSNET	sMAC, dMAC	Cryptography	Time window
uMT6D [41]	2017	Network and Transport	MT6D	Network traffic	Unspecified	Cooja	sIPv6, dIPv6 sPort, dPort	Proxy	Time window
CSP [38]	2017	Network	IPv6	Address exhaustion, and disclosure information	Patient monitoring	ns-2	sIPv6, dIPv6	Cryptography	Peer session
APT [17]	2017	Network	IPv4	External attack	Smart cars	Unspecified	sIPv4, dIPv4	NAT proxy	Communication session
Tor-UDP [24]	2017	Transport	Tor	De-anonymizer attack	Unspecified	Java	sIPv4, dIPv4 sPort, dPort	Tor	Peer session
uOTA [15]	2018	Network and Transport	OTA	Traffic attack	Healthcare	Cooja	sIPv6, dIPv6 sPort, dPort	Multiple IDs pseudonymization	Packet transmission or reception
PeDAAC [21]	2018	Network and MAC	IPv6	Eavesdropper attack	Indoor scenario	Cooja	sIPv6, dIPv6 sMAC, dMAC	Multiple IDs pseudonymization	Time window, node movement, link failure.
MQTT-Tor [24]	2019	Application	MQTT	De-anonymizer attack	Unspecified	Python	Requests	Tor	Peer session
Tor-Delegation [12]	2019	Network	Tor	Network attacks	Unspecified	Cooja	sIPv4, dIPv4	Tor	Peer session

many times depending on the number of ORs along the Tor circuit. Although this might be a good solution to the communication identifiers privacy in IoT networks, the use of the delegation server breaks the end-to-end reachability that is advocated by the design of 6LoWPANs. In addition, the solution is also vulnerable to local attackers as MAC addresses are not changed.

5 Summary of Main Contributions

We sum up the 6LoWPAN-based communication identifiers privacy preserving solutions in Table 1. In our study on communication flow privacy preservation in 6LoWPANs, we find that a many papers are based on obfuscating communication by proxies or by Tor routers, and are based cryptography techniques. We can conclude that these techniques are the most effective for preserving communication privacy in the 6LoWPANs. In addition, 50% of cited papers in this study have not been designed for a specific application and validated their proposals based on simulations.

6 Conclusion

In this survey, we shed light on the relevant 6LoWPAN-based solutions for preserving communication identifiers privacy. We have provided a comprehensive coverage of communication identifiers privacy problems and presented the design guidelines that have been the most important proposed in the literature focusing on analysing each of them from different aspects including the involved communication layer, protocols, type of attacks considered, application scenarios, validation through prototyping of by simulation, etc. We have concluded that successful solutions need to take into account communication flow preservation at all of Link, Network, and Transport layers and should use effective PPTs for preserving communication privacy such as pseudonymization and obfuscation based on Tor or proxy techniques, and reduce the lifespan of the pseudonyms used.

References

1. Atlam, H.F., Wills, G.B.: IoT security, privacy, safety and ethics. In: Farsi, M., Daneshkhah, A., Hosseinian-Far, A., Jahankhani, H. (eds.) Digital Twin Technologies and Smart Cities, pp. 123–149. Springer, Cham (2020)
2. Brilli, L., Pecorella, T., Pierucci, L., Fantacci, R.: A novel 6LoWPAN-ND extension to enhance privacy in IEEE 802.15.4 networks. In: 2016 IEEE Global Communications Conference (GLOBECOM), pp. 1–6. IEEE (2016)
3. Bugeja, J., Jacobsson, A., Davidsson, P.: On privacy and security challenges in smart connected homes. In: 2016 European Intelligence and Security Informatics Conference (EISIC), pp. 172–175. IEEE (2016)
4. Conti, M., Dehghantanha, A., Franke, K., Watson, S.: Internet of Things security and forensics: challenges and opportunities (2018)

5. Da Xu, L., He, W., Li, S.: Internet of Things in industries: a survey. IEEE Trans. Ind. Inform. **10**(4), 2233–2243 (2014)

6. Dos Santos, J., Hennebert, C., Fonbonne, J., Lauradoux, C.: Ephemeral: lightweight pseudonyms for 6LoWPAN MAC addresses. In: IEEE PIMRC (2016)

7. Dunlop, M., Groat, S., Urbanski, W., Marchany, R., Tront, J.: MT6D: a moving target IPv6 defense. In: MILCOM 2011 - 2011 Military Communications Conference, pp. 1321–1326. IEEE (2011)

8. Tails Foundation: Tails-Privacy for anyone anywhere (2019). https://tails.boum.org/index.en.html. Accessed Nov 2019

9. Haus, M., Waqas, M., Ding, A.Y., Li, Y., Tarkoma, S., Ott, J.: Security and privacy in Device-to-Device (D2D) communication: a review. IEEE Commun. Surv. Tutor. **19**(2), 1054–1079 (2017)

10. Hennebert, C., Dos Santos, J.: Security protocols and privacy issues into 6LoWPAN stack: a synthesis. IEEE Internet Things J. **1**(5), 384–398 (2014)

11. Hennebert, C., Santos, J.D.: Security protocols and privacy issues into 6LoWPAN stack: a synthesis. IEEE Internet Things J. **1**(5), 384–398 (2014). https://doi.org/10.1109/JIOT.2014.2359538

12. Hiller, J., Pennekamp, J., Dahlmanns, M., Henze, M., Panchenko, A., Wehrle, K.: Tailoring onion routing to the Internet of Things: security and privacy in untrusted environments. In: 2019 IEEE 27th International Conference on Network Protocols (ICNP), pp. 1–12. IEEE (2019)

13. Hoang, N.P., Pishva, D.: A TOR-based anonymous communication approach to secure smart home appliances. In: 2015 17th International Conference on Advanced Communication Technology (ICACT), pp. 517–525. IEEE (2015)

14. Hossain, M., Islam, S.R., Ali, F., Kwak, K.S., Hasan, R.: An Internet of Things-based health prescription assistant and its security system design. Future Gener. Comput. Syst. **82**, 422–439 (2018)

15. Kouachi, A.I., Sahraoui, S., Bachir, A.: Per packet flow anonymization in 6LoW-PAN IoT networks. In: 2018 6th International Conference on Wireless Networks and Mobile Communications (WINCOM), pp. 1–7. IEEE (2018)

16. Kusek, M.: Internet of Things: today and tomorrow. In: 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), pp. 0335–0338. IEEE (2018)

17. Kwak, B.O., Chung, T.S.: Trust domain based trustworthy networking. In: 2017 International Conference on Information and Communication Technology Convergence (ICTC), pp. 1247–1259. IEEE (2017)

18. Lee, T., Pappas, C., Szalachowski, P., Perrig, A.: Communication based on per-packet one-time addresses. In: 2016 IEEE 24th International Conference on Network Protocols (ICNP), pp. 1–10. IEEE (2016)

19. Loukil, F., Ghedira-Guegan, C., Benharkat, A.N., Boukadi, K., Maamar, Z.: Privacy-aware in the IoT applications: a systematic literature review. In: OTM Confederated International Conferences "On the Move to Meaningful Internet Systems", pp. 552–569. Springer (2017)

20. Malandrino, D., Petta, A., Scarano, V., Serra, L., Spinelli, R., Krishnamurthy, B.: Privacy awareness about information leakage: Who knows what about me? In: Proceedings of the 12th ACM workshop on Workshop on privacy in the electronic society. pp. 279–284 (2013)

21. Mavani, M., Asawa, K.: Privacy enabled disjoint and dynamic address auto-configuration protocol for 6LoWPAN. Ad Hoc Netw. **79**, 72–86 (2018)

22. Mavani, M., Asawa, K.: Resilient against spoofing in 6LoWPAN networks by temporary-private IPv6 addresses. Peer-to-Peer Netw. Appl. **13**, 333–347 (2019)

23. Narten, T., Simpson, W.A., Nordmark, E., Soliman, H.: Neighbor discovery for IPversion 6 (IPv6). RFC 4861 (2007)
24. Protskaya, Y.: Security in the Internet of Things. Ph.D. thesis, Università degli Studi di Parma. Dipartimento di Ingegneria e architettura (2020)
25. Reed, M.G., Syverson, P.F., Goldschlag, D.M.: Anonymous connections and onion routing. IEEE J. Sel. Areas Commun. **16**(4), 482–494 (1998)
26. Selvaraj, S., Sundaravaradhan, S.: Challenges and opportunities in IoT healthcare systems: a systematic review. SN Appl. Sci. **2**(1), 139 (2020)
27. Sen, A.A.A., Eassa, F.A., Jambi, K., Yamin, M.: Preserving privacy in Internet of Things: a survey. Int. J. Inf. Technol. **10**(2), 189–200 (2018)
28. Sha, K., Wei, W., Yang, T.A., Wang, Z., Shi, W.: On security challenges and open issues in Internet of Things. Future Gener. Comput. Syst. **83**, 326–337 (2018)
29. Shelby, Z., Bormann, C.: 6LoWPAN. Wiley, Chichester (2009). https://doi.org/10.1002/9780470686218, https://doi.org/10.1002/9780470686218
30. Snader, R., Kravets, R., Harris III, A.F.: CryptoCoP: lightweight, energy-efficient encryption and privacy for wearable devices. In: Proceedings of the 2016 Workshop on Wearable Systems and Applications, pp. 7–12. ACM (2016)
31. Staudemeyer, R.C., Pöhls, H.C., Wójcik, M.: The road to privacy in IoT: beyond encryption and signatures, towards unobservable communication. In: 2018 IEEE 19th International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM), pp. 14–20. IEEE (2018)
32. Tamò-Larrieux, A.: Technical tools and designs for data protection. In: Tamò-Larrieux, A. (ed.) Designing for Privacy and Its Legal Framework, pp. 101–148. Springer, Cham (2018)
33. Anonymizer Team: IP Rotation for Commercial Enterprises (2020). https://www.anonymizer.com/. Accessed Apr 2020
34. Statista Team: Internet of Things (IoT) connected devices installed base worldwide from 2015 to 2025 (in billions) (2020). https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/. Accessed Apr 2020
35. Tourangeau, R.: Confidentiality, privacy, and anonymity. In: Vannette, D., Krosnick, J. (eds.) The Palgrave Handbook of Survey Research, pp. 501–507. Springer, Cham (2018)
36. Wang, S., Du, Y., Lu, T., Wu, J., Wang, T.: A survey of anonymous communication methods in Internet of Things. In: 2019 IEEE 9th International Conference on Electronics Information and Emergency Communication (ICEIEC), pp. 627–633. IEEE (2019)
37. Wang, X., Mu, Y.: Addressing and privacy support for 6LoWPAN. IEEE Sens. J. **15**(9), 5193–5201 (2015)
38. Wang, X., Mu, Y.: Communication security and privacy support in 6LoWPAN. J. Inf. Secur. Appl. **34**, 108–119 (2017)
39. Yang, Q., Lu, R., Challal, Y., Laurent, M.: Security and privacy in emerging wireless networks. Secu. Commun. Netw. **2017** (2017)
40. Yang, Y., Wu, L., Yin, G., Li, L., Zhao, H.: A survey on security and privacy issues in Internet-of-Things. IEEE Internet Things J. **4**(5), 1250–1258 (2017)
41. Zeitz, K., Cantrell, M., Marchany, R., Tront, J.: Changing the game: a micro moving target IPv6 defense for the Internet of Things. IEEE Wirel. Commun. Lett. **7**(4), 578–581 (2018)

Workflow Security Scheduling Strategy in Cloud Computing

Sarra Hammouti, Belabbas Yagoubi$^{(\boxtimes)}$, and Sid Ahmed Makhlouf

L.I.O. Laboratory, Department of Computer Science,
Faculty of Exact and Applied Sciences, University of Oran1 Ahmed Ben Bella,
P.O. Box 1524, El M'Naouer, Oran, Algeria
b.yagoubi@gmail.com
http://univ-oran1.dz

Abstract. Cloud computing is an effective technology that delivers interesting services to customers over the Internet. It is beneficial for scientific workflow systems in view of its powerful characteristics. However, scheduling workflow system over a cloud platform has become a challenging problem. In this paper, we propose a novel workflow scheduling strategy for the hybrid cloud environments which consists of an economical distribution of tasks between the various cloud service providers, in order to provide customers with high security services. Then, we study the impact of that security services on the total cost and deadline generated by the workflow. This problem is a major gap in the workflow scheduling field and it is still insufficiently explored in the literature. Our proposed scheduling system is composed for three modules. The first module is the Pre-Scheduler, the second is the Security Enhancement Module and the third one is the Post-Scheduler. The system evaluation and the extensive simulations are performed using an extension of Cloudsim simulation tool. The results show that our strategy preserves the same cost however it affects the deadline.

Keywords: Hybrid cloud · Scientific workflow · Scheduling · Cost · Security · Budget · Deadline.

1 Introduction

Cloud computing is a powerful solution that delivers interesting services to customers over the Internet. It refers to provide on-demand computing services and a pool of resource with a pay-per-use system. In addition, it offers a wide network access and fast elasticity. The cloud providers can deploy their services over a public, private, community or hybrid environments. In case of public cloud the infrastructure is located on the premises of the cloud provider and it's open for use by the general public. Private and community cloud infrastructures are intended for use by a single or multiple organizations respectively and they may located either on or off premises. While, the hybrid cloud combines the

two aspects of the public cloud and the private cloud. The delivered services over a cloud platform are either infrastructure as service, platform as service or software as service [17].

These characteristics make the cloud technology beneficial for various scientific fields such as biology, astronomy and chemistry. Where it allows to execute a set of computational tasks or what is called a *scientific workflow* with high performance and minimum cost and time. In fact, scientific workflow is a model that makes it easier for scientists to conceptualize and manage the scientific analysis process in terms of both tasks and data [5], it is generally modeled as Directed Acyclic Graph (DAG) that range a set of computational tasks (nodes) that expose data dependencies between them (edges) in order to solve a scientific problem [9].

Cloud computing offers exceptional opportunities for the complex scientific workflow in terms of cost, performance and reliability. Despite, these opportunities brings various challenges. For instance, it is critical to achieve a high performance and reliability with an optimal use of resources and a minimum of cost and deadline. In the literature, There has been numerous studies that investigate this field by selecting different parameters of Quality of Services (QoS) according to their targeted cloud platform and workflow system. They trying to achieve the best of performance, cost and deadline under different constraints. A closer look to the literature reveals that most of researchers have overlooked the data security problem in the cloud platforms. In fact, this problem is still insufficiently explored, although it is very important for customers to use a secured platforms, especially for their sensitive scientific workflows. In that purpose, we propose in this paper a novel workflow scheduling system for the hybrid cloud environments which consists of an economical distribution of tasks between the various cloud service providers, in order to provide customers with shorter execution times and high security services of within their budget.

Our proposed scheduling system is composed of three modules:

– **Pre-Scheduler** wherein each task or dataset is assigned to be executed or stored either in the private or in the public cloud.
– **Security Enhancement Module** this module is concerned with adding the dataset's required security services while minimizing the cost overhead generated by these services.
– **Post-Scheduler** where each task or dataset is assigned to be executed or stored in the suitable Virtual Machine (VM) while meeting with the budget and deadline constraints.

The performance of our proposed system is evaluated using the Cloud Workflow Simulator (CWS) [14], an extension of the CloudSim simulator [3]. The result shows that our proposed strategy slightly increases the deadline but it does not affect the cost.

The rest of this paper is organized as follows. Sect. 2 presents the related work, we describe the system model and our main assumptions in Sect. 3, the proposed system is explained in Sect. 4, the performance evaluation and the

extensive simulations are detailed in Sect. 5 with a highlight of the main results and at least we conclude the paper with Sect. 6.

2   Related Work

Nowadays, cloud computing becomes a popular solution for many information system issues. In the area of scientific workflow systems, many studies have conducted the important benefits offered by the cloud in term of both performance and cost [8,9].

Thus, with the emergence of this new computing paradigm, many approaches are developed to address the problem of workflow scheduling on the cloud platforms. This challenging problem aims to find an appropriate orchestration of workflow tasks onto resource pool by satisfying the QoS requirements. It is a multi-objective problem because the amount of the allocated resources affect the execution time, cost and performance. It becomes more complex while considering multiple QoS constraints [6,15,22].

Many studies have tackled with the later problem by considering several aspect under different constraints. For example, in [13] authors proposed an approach that aims to improve cost of the workflow by optimizing the use of virtual resources and network bandwidth. While, in [10] the authors focused on how to find an optimized solution to achieve better cost-makespan while maximizing the reliability of executing workflows under user specified deadline and budget constraints. In [24] the authors have proposed a workflow scheduling approach which minimizes cost and makespan at the same time under the constraint of deadline.

In this area, there are some systematic reviews that analyze and describe a large variety of the workflow scheduling approaches and they classify them according to different aspects. For example, In [23] the classification was made regarding to the scheduling process, task and resource. While, in [15] the authors classified the existing approaches based on the type of the algorithm utilized, objectives and properties. Kaur et al. [11] categorized them into heuristic, metaheuristic and hybrid schemes. While, in [16] the authors classified the scheduling schemes according to cloud environments and they described their architecture, key features, and advantages.

In fact, most of the workflow scheduling studies have focused on deadline, budget and other constraints and they ignored the security requirements. Whereas, the majority of companies have private data and they risk irreparable consequences and the loss of trust from their clients if these data are disclosed. Therefore, in the case of applications that handle sensitive data, the main concern is how to provide a secured scheduling strategy [7,11,16].

In the literature, only a few works have dealt with the secured workflow scheduling problem. Among them, we can cite that of Zeng et al. [27] where they presented a security-aware and budget-aware workflow scheduling strategy, in order to provide customers with shorter makespan and meet their security requirements. Li et al. [12] proposed a security and cost aware scheduling

algorithm for heterogeneous tasks of scientific workflow to minimize the total workflow execution cost while meeting the deadline and risk rate constraints. They based on the meta-heuristic optimization technique, particle swarm optimization. While, Shishido et al. [20] examined the effect of both Particle Swarm Optimization (PSO) and Genetic-based algorithms (GA) on attempts to optimize workflow scheduling. An hybrid cloud optimization approach, which combines firefly and bat algorithms was proposed by Arunarani et al. [2].

Chen et al. [4] investigated the problem of scheduling workflows with security-sensitive intermediate data with the objective of minimizing both the makespans and monetary execution costs. While, a cost and energy aware data placement method, for privacy-aware applications over big data in hybrid cloud was proposed by Xu et al. [26]. Wen et al. [25] modeled the problem of scheduling workflows with data privacy protection constraints while minimizing both execution time and monetary cost. In [1], the authors formulated a model for task scheduling and propose a heuristic algorithm which is based on task's completion time and security requirements in which they considered task interaction issues as a security threat. Hammed and Arunkumar [19] proffered a method of scheduling that enabled the users to incur less execution time and cost. They based on the multi-populated genetic algorithm with secured frame work for sensitive data.

Table 1 illustrates a brief comparison between the mentioned works that treat the workflow scheduling problem under the security constrains. The comparison is made according to the objective function, the targeted constraints and the cloud environment model single-cloud or multi-cloud. Among that, we observe that none of these studies were addressed the budget, deadline and security constrains in the multi cloud environments. For that we propose in this paper a security, deadline-, and budget aware workflow scheduling strategy in hybrid Cloud environment.

Table 1. Secured workflow scheduling works.

Work	Objective			Constraints			Model	
	Cost	Makespan	Energy	Budget	Deadline	Security	Single-cloud	Multi-cloud
Zeng et al. [27]		*		*		*		*
Li et al. [12]	*	*			*	*		
Arunarani et al. [2]	*	*			*	*		
Chen et al. [4]	*	*				*	*	
Xu et al. [26]	*	*	*			*		*
Shishido et al. [20]	*	*			*	*		
Wen et al. [25]	*	*				*	*	
Abazari et al. [1]		*				*		*
Hammed and Arunkumar [19]	*	*				*	*	
The present work	*			*	*	*		*

3   System Model and Assumptions

In this section, we extend the system model proposed Makhlouf and Yagoubi[13], as we relied on the same application model and data transfer model proposed by them, and we highlight our main assumptions.

A scientific workflow application consists of a set of computational tasks that expose dependencies between them in order to solve a scientific problem. These dependencies are mainly data, where the output of a task can be the input of another tasks. Scientific workflows are generally modeled as a Directed Acyclic Graph DAG(V,E), where each Vertex (V) represents a task and the Edges (E) represent the task dependencies which indicate the precedence constraints. For example, edge $e(i, j)$ represents that $task_j$ should start its execution after $task_i$ finish, in this case the $task_i$ is called *predecessor* of the $task_j$ and the $task_j$ is called the *successor* of the $task_i$. A task can have one or more predecessors, it cannot start execution until all of its predecessors are finished and all its input files are available. Weight of the edge $e(i, j)$ indicate the amount of data transferred form $task_i$ to $task_j$.

In order to maintain the sensitive data, we assume that each dataset will be accompanied by a security level which reflect its degree of sensitivity. For example, SL_i represent the security level of the dataset D_i, where SL_i takes different values from lowest security level 0 to the highest one 5. A dataset can be manipulated by one or more tasks, and a task can manipulate one or more datasets.

We target the hybrid cloud system as an execution environment, we assume that the client disposes his private cloud and needs to allocate more resources in the public cloud to run his scientific workflow application. A cloud offers an unbounded set of VMs that can be rapidly provisioned and released, it's characterized by On-demand self-service, broad network access, resource pooling, rapid elasticity and measured service according to the NIST[1]. We suppose that a VM can run one task a time and our pricing model is rounded to 1\$/ h in the public cloud. We aim to secure the sensitive data and not to exceed the budget and deadline specified by users.

4   Proposed System

To address the security, deadline, and budget-aware workflow scheduling issue in the hybrid cloud environment, we propose a three-modules system as it's shown in Fig. 2. The first module is the **Pre-Scheduler**, wherein each task or dataset is assigned to be executed or stored either in the private or in the public cloud. The second module is the **Security Enhancement Module**, this module is concerned with adding the dataset's required security services while minimizing the cost overhead generated by these services. The third module is the **Post-Scheduler**, where each task or dataset is assigned to be executed or stored in the suitable VM while meeting with the budget and deadline constraints.

---

[1] National Institute of Standards and Technology.

Figure 1 illustrates an example of whole proposed system, where datasets {D1, D2, D3} and tasks {T1, T2, T4, T6} are assigned to the private cloud and the rest to the public cloud.
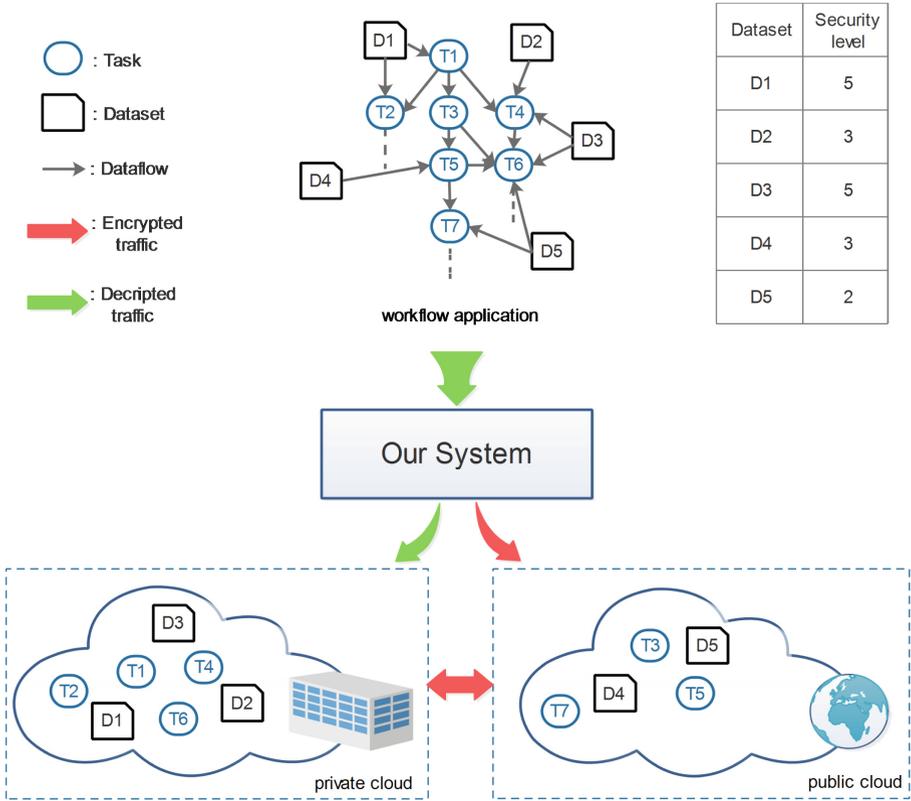
Fig. 1. System model

4.1   Pre-scheduler

Since the private cloud is lower in cost according to the public cloud, in this module we start by assigning the sensitive data (Form the highest security level to the lowest one) and the tasks that manipulate them in the private cloud if the required resources are available.

In order to reduce the data transfer cost, we assign to the private cloud all the datasets manipulated by tasks located in the private cloud.
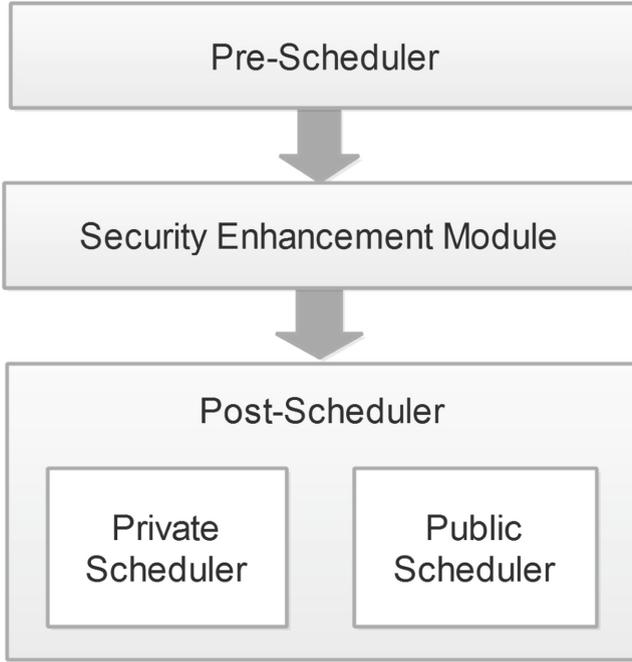
Fig. 2. Principal system components

At least of this phase it remains only the datasets and tasks that require the unavailable resources in the private cloud. So, we assign them to be stored or executed in the public cloud. Simply, this module take as input the workflow and the list of available resources in the private cloud and it gives an execution plan as output. The execution plan refers to the list of tasks which must be executed in the private cloud and that which must be executed in the public one and similarly for the dataset. Algorithm 1 shows by details the principle functioning of our scheduler. And, Table 2 illustrates the used symbols and their descriptions.

4.2   Security Enhancement Module

Considering that the private cloud is secure and that threats come from the public one, we aim in this module to secure the sensitive data that are assigned to be stored in the public cloud, their dependent tasks and the data flowing inter the public/private cloud. To this end, we have proposed adding certain security services and cryptographic functions. Since these services produce a significant overhead to the total cost and may increase the execution time, we have chosen them basing on the data security level, budget and deadline specified by the user.

This module takes as input the execution plan generated by the **Pre-scheduling module** and specifies the suitable encryption/decryption algorithms

Algorithm 1: Pre-Scheduling

Input: $W = \{T, E, D, SL, R\}$; /* W: workflow; T: set of tasks; E: set of edges or task dependancies, D: data, SL: security level for each data, R: required resources for each task   */
$PR = \{pr_1, pr_2, ..., pr_k\}$; /* a set of available resources in private cloud   */
Output: $T_{pri} = \emptyset$; $D_{pri} = \emptyset$; $T_{pub} = \emptyset$; $D_{pub} = \emptyset$;
/* T_{pri}:list of private cloud tasks; D_{pri}=list of private cloud data; T_{pub}=list of public cloud tasks; D_{pub}=list of public cloud data;   */
begin

  foreach $t_i \in T$ **do**
    if $R_i \nsubseteq PR$ /* R_i required resource of t_i */
    then
      $T_{pub} = T_{pub} \cup \{t_i\}$;/* placement of t_i in the public cloud   */
      $T = T - \{t_i\}$;   /* remove t_i from the initial task list   */
      $D_{pub} = D_{pub} \cup \{D_i\}$;   /* D_i all data set manipulated by task t_i */
      $D = D - \{D_i\}$;   /* remove D_i from the initial data list   */

  foreach $d_j \in D$ **do**
    if *(sl$_j$ is high **and** $T_j \nsubseteq T_{pub}$)* /* sl_j : security level of d_j ; T_j:set of tasks that manipulate d_j */
    then
      $D_{pri} = D_{pri} \cup \{d_j\}$;
      $D = D - \{d_j\}$;
      $T_{pri} = T_{pri} \cup \{T_j\}$;
      $T = T - \{T_j\}$;

  while $(\exists t_i \subseteq T_{pri} \wedge D_i \subseteq D) \vee (\exists d_j \subseteq D_{pri} \wedge T_i \subseteq T)$ **do**
    foreach $t_i \subseteq T_{pri}$ **do**
      $D_{pri} = D_{pri} \cup \{D_i\}$;
      $D = D - \{D_i\}$;
    foreach $d_j \subseteq D_{pri}$ **do**
      $T_{pri} = T_{pri} \cup \{T_j\}$;
      $T = T - \{T_j\}$;

  foreach $t_i \subseteq T$ **do**
    if *public_depandencies(t_i)$\leq$private_depandencies(t_i)*
      /* public_depandencies(t_i)/private_depandencies(t_i) return the numbre of tasks palaced in the public/private cloud and have depandencies with t_i */
    then
      $T_{pri} = T_{pri} \cup \{t_i\}$;
      $T = T - \{t_i\}$;
      $D_{pri} = D_{pri} \cup \{D_i\}$;
      $D = D - \{D_i\}$;
    else
      $T_{pub} = T_{pub} \cup \{t_i\}$;
      $T = T - \{t_i\}$;
      $D_{pub} = D_{pub} \cup \{D_i\}$;
      $D = D - \{D_i\}$;

for each sensitive data either located in the public cloud or transfered from the private cloud to the public one or vice versa. Also, it calculates the total overhead generated by these algorithms in order to preserve the user constraints (budget & deadline).

Table 2. Descriptions of symbols used in Pre-Scheduling algorithm

Symbols	Description
W	Workflow
T	Set of tasks
E	Set of edges or task dependencies
D	Data
R	Set of the required resources
SL	Security Level
PR	Set of available resources in private cloud
t_i	$task_i$
D_i	Dataset manipulated by task t_i
R_i	Required resource of t_i
d_j	$dataset_j$
sl_j	Security level of d_j
T_j	Set of tasks that manipulate d_j
T_{pri}	List of tasks affected to be executed in the private cloud
D_{pri}	List of data affected to be stored in the private cloud
T_{pub}	List of tasks affected to be executed in the public cloud
D_{pub}	List of data affected to be stored in the public cloud

4.3   Post-Scheduler

In this phase we aim to schedule tasks inside each cloud platform (private cloud, public cloud). For that, we considered two schedulers **(private scheduler and public scheduler)**. Where, we relied for each of them on the scheduler proposed by Makhlouf and Yagoubi [13] as it's improve its performance. In addition, we extended them in order to take into consideration the cost overhead produced by the security enhancement module.

Private Scheduler schedule tasks assigned to the private cloud, it works basing on the principle proposed in [13]. While, for the total execution cost it consider the additional cost generated by:

1. Encryption algorithms concerning tasks that send sensitive data to the public cloud
2. Decryption algorithms concerning tasks that receive sensitive data from the public cloud

Public Scheduler schedule tasks assigned to the public cloud, it based on the scheduler proposed in [13]. While, for the total execution cost it consider the overhead generated by:

1. Encryption/Decryption algorithms for tasks that manipulate data stored in the public cloud.
2. Encryption algorithms for tasks that send sensitive data to the private cloud.
3. Decryption algorithms concerning tasks that receive sensitive data from the private cloud.

5   Performance Evaluation and Results

In order to evaluate our scheduling model, we implemented it by extending the work in [13]. To do this, we have supposed a security level of data (SL) in the Synthetic Workflow [21,28]. Since synthetic workflows do not support the security level, we have generated the security level of each data following a normal distribution :

$$SL \leftarrow \sqrt{\frac{\sum_{i=1}^{n}(x_i - x_m)^2}{n}}$$

With x_i is the size of the data i and x_m is the average of the n data size. In statistics, the normal distribution is the most important probability distribution. It is used to model unbiased uncertainties random errors of additive type, and symmetric distributions of processes and natural phenomena [18].

The selected applications include LIGO in Fig. 3a (Laser Interferometer Gravitational-Wave Observatory), a data-intensive application, and MONTAGE in Fig. 3b, an I/O bound workflow.

LIGO Workflow   MONTAGE Workflow
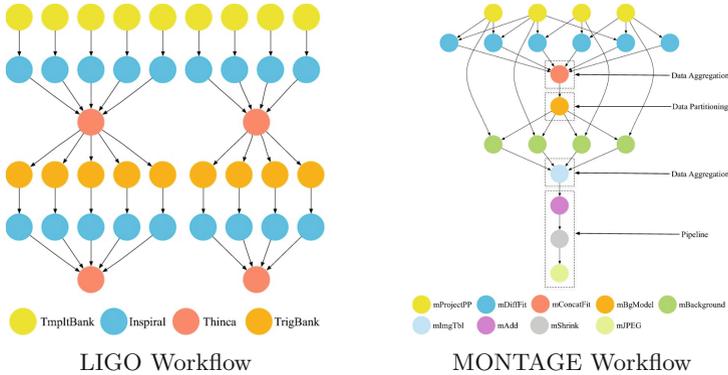
Fig. 3. Scientific workflow structures

As in [13], we simulated Workflows whose size does not exceed 200 tasks. We have fixed budget to 20 virtual machines and measured the impact of security level on Workflow cost and the Deadline. We compared our approach (SLp) with the standard approach (SDp) in [13] which does not support the level of security.

5.1   Impact of the Security Level on the Cost

MONTAGE workflow execution cost

LIGO workflow execution cost
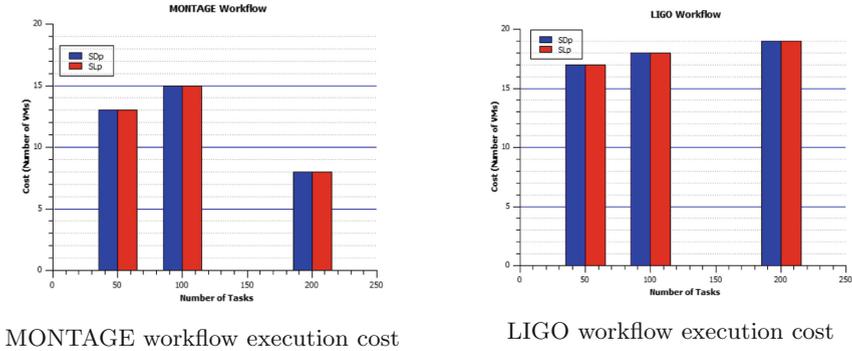
Fig. 4. Impact of the security level on the Cost

From Figs. 4a and 4b, we simulated the execution of the MONTAGE and LIGO workflows respectively and measured the costs in VMs number. We note that regardless of the size of the workflow, our policies give same results as standard policy. It is possible to explain these results by the fact that in our resources Cloud model, a virtual machine can execute only one task at a time and it is charged 1 \$ for each interval of 60 min (one hour) of operation. But a partial usage of a billing interval is rounded. The proof of this assumption will be shown during the next simulation.

5.2   Impact of the Security Level on the Deadline

MONTAGE workflow Deadline

LIGO workflow Deadline
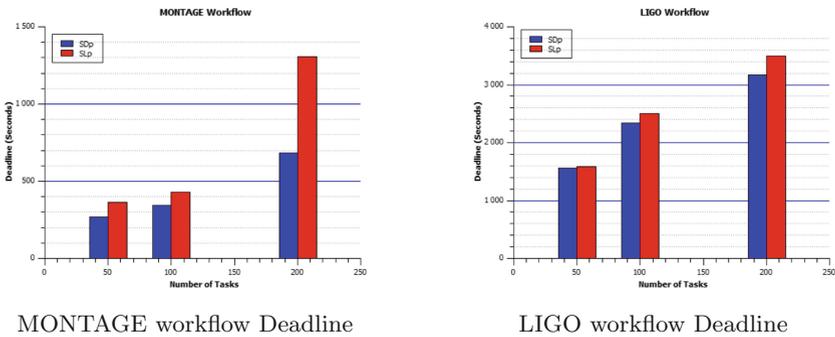
Fig. 5. Impact of the security level on the Deadline

From Figs. 5a and 5b, we simulated the execution of the MONTAGE and LIGO workflows respectively and measured the Deadline. We note that, regardless of

the size of the workflow, our policies give bad results unlike the standard policy which gives excellent results by reducing Deadline and this whatever the size of the workflow. This result is due to the overload of data encryption/decryption operations when they are transferred to the public cloud. This overload increases the Deadline allowing the consumption of the entire of the last 60-min interval. This consumption does not affect the cost but increases Deadline.

6   Conclusion

The present paper introduce a novel workflow scheduling strategy for the hybrid cloud environments which consists of an economical distribution of tasks between various cloud service providers, in order to provide customers with shorter execution times, lower cost and high security services within a limited budget and deadline. The result shows that our strategy increases the deadline but it does not affect the cost. We have shown that the deadline increase is due to the encryption/decryption operations. So that increase does not affect the cost, our strategy exploits to the maximum the virtual machines, especially the last 1-h slot time of virtual machines.

Future studies in this field can deal with the addition of security services at the data and task level at the same time, achieve a more economical cryptographic strategy to ensure the security requirements, find a scheduling strategy that takes into consideration more parameters and more constraints such as energy.

References

1. Abazari, F., Analoui, M., Takabi, H., Fu, S.: MOWS: multi-objective workflow scheduling in cloud computing based on heuristic algorithm. Simul. Model. Pract. Theory **93**, 119–132 (2019). https://doi.org/10.1016/j.simpat.2018.10.004. https://linkinghub.elsevier.com/retrieve/pii/S1569190X18301515
2. Arunarani, A.R., Manjula, D., Sugumaran, V.: FFBAT: a security and cost-aware workflow scheduling approach combining firefly and bat algorithms. Concurr. Comput.: Pract. Exp. **29**(24), e4295 (2017). https://doi.org/10.1002/cpe.4295
3. Calheiros, R.N., Ranjan, R., Beloglazov, A., Rose, C.A.F.D., Buyya, R.: CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. Softw. Pract. Exp. **41**(1), 23–50 (2011)
4. Chen, H., Zhu, X., Qiu, D., Liu, L., Du, Z.: Scheduling for workflows with security-sensitive intermediate data by selective tasks duplication in clouds. IEEE Trans. Parallel Distrib. Syst. **28**(9), 2674–2688 (2017). https://doi.org/10.1109/TPDS.2017.2678507. http://ieeexplore.ieee.org/document/7872483/
5. Davidson, S.B., Freire, J.: Provenance and scientific workflows: challenges and opportunities. In: Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, pp. 1345–1350 (2008)
6. Fakhfakh, F., Kacem, H.H., Kacem, A.H.: Workflow scheduling in cloud computing: a survey. In: 2014 IEEE 18th International Enterprise Distributed Object Computing Conference Workshops and Demonstrations, pp. 372–378. IEEE, Ulm, September 2014. https://doi.org/10.1109/EDOCW.2014.61. http://ieeexplore.ieee.org/document/6975385/

7. Francis, A.O., Emmanuel, B., Zhang, D., Zheng, W., Qin, Y., Zhang, D.: Exploration of secured workflow scheduling models in cloud environment: a survey. In: 2018 Sixth International Conference on Advanced Cloud and Big Data (CBD), pp. 71–76. IEEE, Lanzhou, August 2018. https://doi.org/10.1109/CBD.2018.00022. https://ieeexplore.ieee.org/document/8530818/

8. Hoa, C., Mehta, G., Freeman, T., Deelman, E., Keahey, K., Berriman, B., Good, J.: On the use of cloud computing for scientific workflows. In: 2008 IEEE Fourth International Conference on eScience, pp. 640–645. IEEE, Indianapolis, December 2008. https://doi.org/10.1109/eScience.2008.167. http://ieeexplore.ieee.org/document/4736878/

9. Juve, G., Deelman, E.: Scientific workflows in the cloud. In: Cafaro, M., Aloisio, G. (eds.) Grids, Clouds and Virtualization. CCN, pp. 71–91. Springer, London (2011). https://doi.org/10.1007/978-0-85729-049-6_4

10. Kalra, M., Singh, S.: Multi-criteria workflow scheduling on clouds under deadline and budget constraints. Concurr. Comput.: Pract. Exp. **31**(17), e5193 (2019). https://doi.org/10.1002/cpe.5193

11. Kaur, S., Bagga, P., Hans, R., Kaur, H.: Quality of service (QoS) aware workflow scheduling (WFS) in cloud computing: a systematic review. Arab. J. Sci. Eng. **44**(4), 2867–2897 (2019). https://doi.org/10.1007/s13369-018-3614-3

12. Li, Z., et al.: A security and cost aware scheduling algorithm for heterogeneous tasks of scientific workflow in clouds. Future Gener. Comput. Syst. **65**, 140–152 (2016). https://doi.org/10.1016/j.future.2015.12.014. https://linkinghub.elsevier.com/retrieve/pii/S0167739X15003982

13. Makhlouf, S.A., Yagoubi, B.: Data-aware scheduling strategy for scientific workflow applications in IaaS cloud computing. Int. J. Interact. Multimed. Artif. Intell. **5**(4), 75 (2019). https://doi.org/10.9781/ijimai.2018.07.002. http://www.ijimai.org/journal/node/2496

14. Malawski, M., Juve, G., Deelman, E., Nabrzyski, J.: Algorithms for cost- and deadline-constrained provisioning for scientific workflow ensembles in IaaS clouds. Future Gener. Comput. Syst **48**, 1–18 (2015)

15. Masdari, M., ValiKardan, S., Shahi, Z., Azar, S.I.: Towards workflow scheduling in cloud computing: a comprehensive analysis. J. Netw. Comput. Appl. **66**, 64–82 (2016). https://doi.org/10.1016/j.jnca.2016.01.018. https://linkinghub.elsevier.com/retrieve/pii/S108480451600045X

16. Masdari, M., Zangakani, M.: Efficient task and workflow scheduling in inter-cloud environments: challenges and opportunities. J. Supercomput. **76**(1), 499–535 (2020). https://doi.org/10.1007/s11227-019-03038-7. http://link.springer.com/10.1007/s11227-019-03038-7

17. Mell, P., Grance, T.: The NIST Definition of Cloud Computing, p. 7 (2011)

18. Mishra, S., Datta-Gupta, A.: Distributions and models thereof (chap. 3). In: Mishra, S., Datta-Gupta, A. (eds.) Applied Statistical Modeling and Data Analytics, pp. 31–67. Elsevier (2018)

19. Shahul Hammed, S.Arunkumar, B.: Efficient workflow scheduling in cloud computing for security maintenance of sensitive data. Int. J. Commun. Syst. e4240 (2019). https://doi.org/10.1002/dac.4240

20. Shishido, H.Y., Estrella, J.C., Toledo, C.F.M., Arantes, M.S.: Genetic-based algorithms applied to a workflow scheduling algorithm with security and deadline constraints in clouds. Comput. Electr. Eng. **69**, 378–394 (2018). https://doi.org/10.1016/j.compeleceng.2017.12.004. https://linkinghub.elsevier.com/retrieve/pii/S0045790617312259

21. da Silva, R.F., Chen, W., Juve, G., Vahi, K., Deelman, E.: Community resources for enabling research in distributed scientific workflows. In: eScience, pp. 177–184. IEEE Computer Society (2014)
22. Singh, S., Chana, I.: A survey on resource scheduling in cloud computing: issues and challenges. J. Grid Comput. **14**(2), 217–264 (2016). https://doi.org/10.1007/s10723-015-9359-2
23. Smanchat, S., Viriyapant, K.: Taxonomies of workflow scheduling problem and techniques in the cloud. Future Gener. Comput. Syst. **52**, 1–12 (2015). https://doi.org/10.1016/j.future.2015.04.019. https://linkinghub.elsevier.com/retrieve/pii/S0167739X15001776
24. Wang, P., Lei, Y., Agbedanu, P.R., Zhang, Z.: Makespan-driven workflow scheduling in clouds using immune-based PSO algorithm. IEEE Access **8**, 29281–29290 (2020). https://doi.org/10.1109/ACCESS.2020.2972963. https://ieeexplore.ieee.org/document/8990144/
25. Wen, Y., Liu, J., Dou, W., Xu, X., Cao, B., Chen, J.: Scheduling workflows with privacy protection constraints for big data applications on cloud. Future Gener. Comput. Syst. (2018). https://doi.org/10.1016/j.future.2018.03.028. https://linkinghub.elsevier.com/retrieve/pii/S0167739X17307379. S0167739X17307379
26. Xu, X., et al.: Data placement for privacy-aware applications over big data in hybrid clouds. Secur. Commun. Netw. **2017**, 1–15 (2017). https://doi.org/10.1155/2017/2376484. https://www.hindawi.com/journals/scn/2017/2376484/
27. Zeng, L., Veeravalli, B., Li, X.: SABA: a security-aware and budget-aware workflow scheduling strategy in clouds. J. Parallel Distrib. Comput. **75**, 141–151 (2015). https://doi.org/10.1016/j.jpdc.2014.09.002. https://linkinghub.elsevier.com/retrieve/pii/S0743731514001658
28. Zhou, A.C., He, B., Ibrahim, S.: eScience and big data workflows in clouds: a taxonomy and survey (chap. 18). In: Buyya, R., Calheiros, R.N., Dastjerdi, A.V. (eds.) Big Data, pp. 431–455. Morgan Kaufmann (2016)

An Optimized Energy-Efficient Mission-Based Routing Protocol for Unmanned Aerial Vehicles

Mohamed Skander Daas[1,2]($\boxtimes$) , Zakaria Benahmed[3], and Salim Chikhi[2,3]

[1] Universite Freres Mentouri Constantine 1, 25000 Constantine, Algeria
`daas.skander@umc.edu.dz`
[2] MISC Laboratory, 25000 Constantine, Algeria
[3] University of Abdelhamid Mehri Constantine 2, 25000 Constantine, Algeria
`benahmedzakaria1996@gmail.com, salim.chikhi@univ-constantine2.dz`

Abstract. Routing in unmanned aerial vehicles (UAVs) faces serious energy consumption problems due to the premature battery depletion caused by the intensive exchange of control messages between the UAVs. The frequent use of control messages increases the delivery ratio. However, it increases the energy consumption which prevents the UAVs from completing their mission on time. In this paper, we present OEM-AODV, an adaptive routing protocol based on AODV, whereby the periodic control messages are dynamically sent to reduce energy consumption. We designed a predictive mathematical model through several steps based on performance modeling and multi-objective optimization using the Design of Experiments. Thereby, the frequency of sending periodic control messages is defined according to some mission parameters namely dimensions of the mission area, the maximum velocity of UAVs, and their transmission range. The simulation experiments using the NS-3 simulator show that OEM-AODV reduces significantly both energy consumption and routing overhead when compared with the original AODV and with a recent adaptive version EE-HELLO-AODV.

Keywords: Routing protocols · Unmanned aerial vehicles · Performance modeling · Mutli-objective optimization · Design of experiments.

1 Introduction

Unmanned aerial vehicles, generally called drones, can communicate via wireless links and have the distinction of moving freely in space forming a flying ad hoc network used for a wide variety of military and civilian applications. UAVs are typically used for missions in hostile environments for many reasons such as surveillance, pursuit, or inspection of dangerous or unknown sites.

Two critical problems facing FANETs are (i) data loss caused by frequently lost connection links between UAVs due to node mobility. (ii) The critical shortcoming of the battery capacity of UAVs, where the UAVs batteries are rapidly

depleted. It is not easy to resolve a tradeoff between these problems due to the characteristics of FANETs. However, in order that UAVs can be functional for a longer time and can complete their mission in a sufficient time, FANETs require special attention regarding UAVs' energy saving where energy sources are limited. Much of a UAV's energy consumption is related to the extensive use of control messages of the routing protocols to ensure information forwarding and communication between different UAVs. For this, it is essential to design routing protocols that can determine the best refreshing rate of the control messages that optimizes the energy consumption for a given mission.

In literature, most works adapt the refresh interval according to intuitively developed formulas in terms of the main influential parameters. However, the impact of the interaction between various parameters on the network performance is very hard to understand and to model without the intervention of mathematical modeling and optimization techniques.

In this regard, we propose an optimized energy-efficient routing protocol for Unmanned Aerial Vehicles based on UAVs mission called OEM-AODV. Where many steps based on the Design of Experiments are followed to obtain a mathematical model that predicts the adequate frequency of sending Hello messages to optimize the energy consumption for a given UAVs' mission.

The rest of this paper is organized as follows: Sect. 2 presents the related works. Sect. 3 describes the different steps to design the proposed adaptive routing protocol. Sect. 4 presents the simulation evaluation and performance analysis of OEM-AODV compared with EE-HELLO-AODV [1] and AODV [2]. Finally, the conclusion is presented in Sect. 5.

2   Related Works

In UAVs mission, each node in the network sends control messages periodically as broadcast to all its neighbors. These messages refresh the information of the network links in routing tables. The delay that a UAV waits until a message is sent, is called the interval period or Hello interval. There are several routing protocols for FANET, few are standardized. For example, the standard AODV which is the most used for developing new derivative protocols uses a fixed value for this timer of 1 s. As regards routing protocols for FANETs, the works in [3–5] summarize different proposed techniques.

There have been several adaptive Hello interval techniques for routing protocols in FANETs. Because using beaconing in a short interval needs a high communication overhead for nodes that do not exhibit significant dynamism. On the other hand, using a fixed long interval leads to use outdated locations, which is adverse for routing packets effectively. Consequently, MAAR [6] uses a dynamic location update scheme, which consists of an adaptive beacon refresh and global position update. The first provides velocity and position information to one-hop neighbors while the second makes nodes known all possible destination positions. To achieve both low overhead and position accuracy simultaneously, the transmission rate of beacons in MAAR is adapted to the motion of the nodes. So,

highly mobile nodes require sending updates frequently because their positions are changing rapidly. On the contrary, for nodes that move slowly, the beacon sending rate does not necessitate being frequent because of the relative stable topology.

Authors of [7] propose a routing protocol called Q-learning based Multi-objective optimization Routing protocol (QMR), in which the time interval of the HELLO message can be adapted according to the node velocity. The higher the moving velocity of the node is, the smaller the Hello interval is. On the contrary, the Hello interval becomes larger.

To address the problem of unnecessary energy consumption, authors in [1] proposed an adaptive hello-interval scheme-energy efficient hello (EE-HELLO-AODV) based on mission-related information, such as the allowed airspace volume, number of UAVs, transmission range, and velocity.

In [8], the authors proposed an adaptive control packet mechanism, which is directly related to the periodic hello messages, with vehicles' speeds, causing a significant reduction of control overhead over the network.

Authors in [9] introduced an Adaptive Density-based Routing Protocol (ADRP) for FANETs. The main objective is to calculate the probability of forwarding adaptively to increase the forwarding efficiency in FANETs. ADRP fine-tunes dynamically the probability of rebroadcasting frequency of nodes for routing request packets in terms of the number of neighbor nodes.

Authors in [10] proposed a scheme (LCR) to enhance the performance of the routing protocols used in MANET to reduce the overhead. In the introduced scheme, instead of sending periodic control messages at a fixed frequency, the authors used a link change rate estimation to dynamically adapt the rate at which each node sends the messages. The link change rate estimation is measured according to the connectivity that reflects the network conditions. Table 1 show the considered parameters for adapting the refresh interval of the above cited works.

Table 1. Related parameters in adaptive refresh interval for each protocol

Parameter	MAAR	QMR	EE-HELLO	MA-DP	ADRP	LCR	Proposed
Number of nodes			✓		✓		✓
Velocity	✓	✓	✓	✓			✓
Position		✓		✓			
Transmission range			✓				✓
Mission area			✓				✓
Number of new/lost links						✓	

The above studies focus on an adaptive control message rate to improve performances of routing protocols. However, the implication of many factors that have an impact on the performance makes the task of adapting the frequency of sending messages very complex, uncontrollable, and imprecise without using methods and techniques based on mathematical modeling.

3   Proposed Protocol

In this study, we used the Design of Experiments to optimize the frequency of sending Hello messages in the AODV routing protocol with regard to some UAVs' mission parameters. AODV relies on Hello messages to maintain direct link connectivity. AODV sends periodically Hello messages, the time of which is defined by the fixed AODV Hello interval. The use of HELLO messages by the AODV routing protocol has a significant impact on its performance [10], such as number of routing messages, energy consumption, and delivery ratio. In order to optimize the energy consumption of UAVs in a given mission, it is important to know which value of the Hello interval (T_{hello}) should be used.

Given the implication of several parameters, it is difficult to determine the adequate value of T_{hello} for a given mission to minimize energy consumption without deteriorating the delivery ratio too much. Despite the existence of several optimization methods we chose the RSM technique [11] given its usability not only for an optimization objective but above all for its ability to generate predictive mathematical models that reflect the effect of the different parameters and their interactions by following a systematic plan which requires just a few experiments. Using this methodology allows mathematical estimation of good value of T_{hello} according to the predetermined data for a given mission namely:

- V_{max} (m/s): The maximum velocity of UAVs during the mission.
- T_x (m): The transmission range of UAVs.
- D (UAVs/Km3): the density of UAVs during the mission. It can be defined by the considered number of UAVs (N) in the occupied space (volume) by UAVs. This space is defined by a length (L), width (W), and height (H).

See Fig. 1.
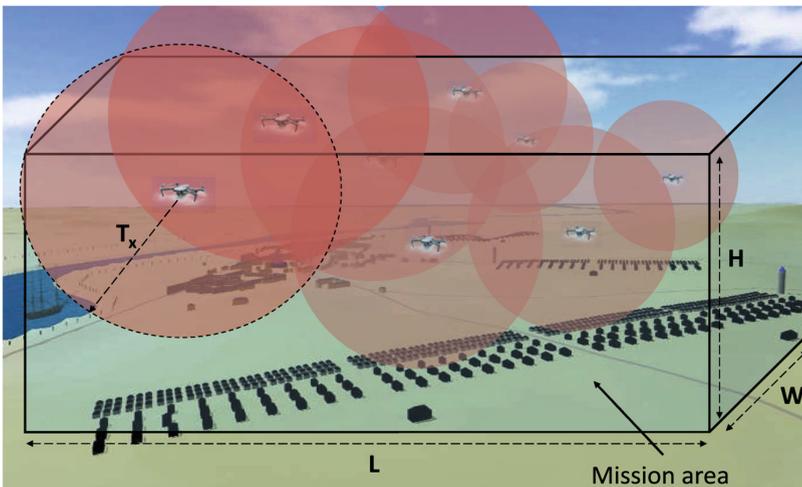
Fig. 1. An example of a group of UAVs accomplishing a mission.

To define the mathematical model we considered this problem as an inverse problem, where its parameters are: packet delivery ratio (PDR), energy consumption (EC), V_{max}, D, T_x, and T_{hello}.

3.1   Mathematical Modeling of PDR and EC Performances Using RSM Methodology

In statistics, response surface methodology (RSM) defines the relationships between several explanatory parameters (input parameters) and one or more response variables (output parameters). Each parameter takes one of three equally spaced values coded as -1, 0, and $+1$. Table 2 shows the adopted parameter levels.

The main concept of RSM is to use a predetermined sequence of experiments in which changes are made in the input parameters to observe the evolution of the output parameters. Most designs use a second-degree polynomial model (see Eq. 1). The mathematical model is used because performances could be mathematically modeled [12,13], such a model is easy to estimate and apply, even when little is known about the process [14].

Table 2. Parameter levels.

Parameter	V_{max} (m/s)	D (nodes/km^3)	T_x (m)	T_{hello} (s)
Level -1	10	20/(0.6 * 0.6 * 0.15)	150	1
Level 0	30	30/(0.6 * 0.6 * 0.15)	300	5
Level $+1$	50	40/(0.6 * 0.6 * 0.15)	450	9

The second order polynomial model can be written by Eq. 1 [11].

$$Y = \lambda_0 + \sum_{i=1}^{k} \lambda_i x_i + \sum \sum_{i<j}^{k} \lambda_{ij} x_i x_j + \sum_{i=1}^{k} \lambda_{ii} x_i^2 + \epsilon \tag{1}$$

Where Y is the expected response, λ_0 is the constant coefficient, λ_i is the ith linear coefficient of the input parameter x_i, λ_{ii} is the ith quadratic coefficient of the input parameter x_i, λ_{ij} is the interaction coefficient between the input parameters x_i and x_j, and ϵ is the error of the model.

The estimated coefficients $\hat{\lambda}$ are obtained by mathematical regression according to the following formula: 2:

$$\hat{\lambda} = (X'X)^{-1} X'y \tag{2}$$

Where X' is the transpose of the matrix X, X is the matrix of the model which depends on the experimental points chosen to execute the design of the considered model, and y is the vector of responses.

We have used Box Behnken design which is an RSM methodology in which the experiments (simulations using NS-3 simulator) are performed using V_{max}, N, T_x, and T_{hello} as input parameters. The measured results of these experiments are PDR and EC performances (outputs). Table 3 shows the experimental design matrix and its measured responses (results).

Table 3. Box Behnken design matrix with responses

Exp	D	V_{max}	T_x	T_{hello}	EC	PDR
1	−1	−1	0	0	4,92286	0,9
2	1	−1	0	0	13,7691	0,8537234
3	−1	1	0	0	5,5828	0,7787234
4	1	1	0	0	16,667	0,73138298
5	0	0	−1	−1	9,37197	0,54680851
6	0	0	1	−1	15,4488	0,95177305
7	0	0	−1	1	8,39825	0,45248227
8	0	0	1	1	4,45175	0,93758865
9	−1	0	0	−1	8,62242	0,86702128
10	1	0	0	−1	27,3192	0,83882979
11	−1	0	0	1	5,37745	0,77553191
12	1	0	0	1	14,7613	0,72287234
13	0	−1	−1	0	8,04565	0,60638298
14	0	1	−1	0	9,51051	0,44326241
15	0	−1	1	0	4,07972	0,96737589
16	0	1	1	0	5,04073	0,90283688
17	−1	0	−1	0	2,79501	0,33723404
18	1	0	−1	0	19,2345	0,48776596
19	−1	0	1	0	2,81767	0,9412766
20	1	0	1	0	7,90213	0,89148936
21	0	−1	0	−1	18,9602	0,90283688
22	0	1	0	−1	18,7583	0,83900709
23	0	−1	0	1	7,55413	0,85531915
24	0	1	0	1	9,98661	0,7141844
25	0	0	0	0	9,10975	0,79219858
26	0	0	0	0	9,10975	0,79219858
27	0	0	0	0	9,10975	0,79219858

Using the Box Behnken RSM methodology, the performances of PDR and EC can be modeled according to Eqs. 3 and 4 (for further details on Box Behnken design, see [11]).

$$EC(D, V_{max}, T_x, T_{hello}) = \beta_0 + \beta_1 D + \beta_2 V_{max} + \beta_3 T_x + \beta_4 T_{hello}$$
$$+ \beta_{12} D V_{max} + \beta_{13} D T_x + \beta_{14} D T_{hello} + \beta_{23} V_{max} T_x + \beta_{24} V_{max} T_{hello}$$
$$+ \beta_{34} T_x T_{hello} + \beta_{11} D^2 + \beta_{22} V_{max}^2 + \beta_{33} T_x^2 + \beta_{44} T_{hello}^2 + \epsilon \qquad (3)$$

$$PDR(D, V_{max}, T_x, T_{hello}) = \alpha_0 + \alpha_1 D + \alpha_2 V_{max} + \alpha_3 T_x + \alpha_4 T_{hello}$$
$$+ \alpha_{12} D V_{max} + \alpha_{13} D T_x + \alpha_{14} D T_{hello} + \alpha_{23} V_{max} T_x + \alpha_{24} V_{max} T_{hello}$$
$$+ \alpha_{34} T_x T_{hello} + \alpha_{11} D^2 + \alpha_{22} V_{max}^2 + \alpha_{33} T_x^2 + \alpha_{44} T_{hello}^2 + \epsilon \qquad (4)$$

The values of coefficients α_i and β_i are shown in Table 5 Figs. 2 and 3 show respectively surface responses of EC and PDR performances according to the changing parameters based on Eqs. 3 and 4.
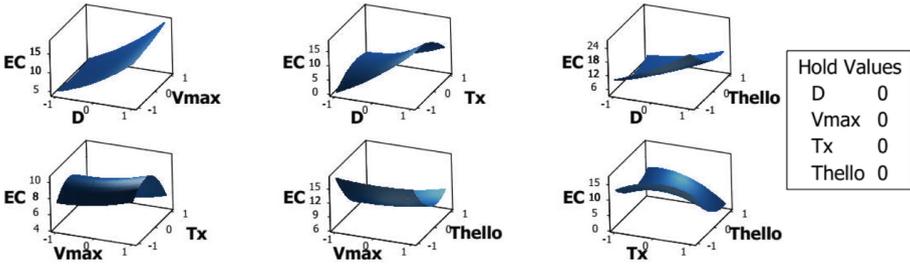
Fig. 2. Response surfaces of the energy consumption (EC) according to Eq. 3.
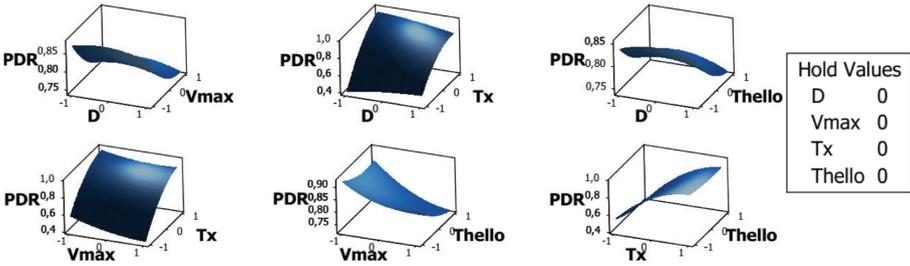
Fig. 3. Response surfaces of the packet delivery ratio (PDR) according to Eq. 4.

3.2 Modelling of T_{hello} as a Function of V_{max}, N, and T_x

To obtain a model that estimates T_{hello} (response) as a function of the three mission parameters (V_{max}, N, and T_x), another Box-Behnken design which is an RSM methodology is considered by screening V_{max}, N, and T_x parameters (inputs) in Eqs. 3 and 4, and then calculating the best value of T_{hello}

(response) by optimizing those models (see Table 4). Using the obtained measurements of (T_{hello}) from Table 4, the resulting mathematical Eq. 5 can be obtained using mathematical multivariable linear regression provided by the Box-Behnken methodology:

$$T_{hello}(D, V_{max}, T_x) = \gamma_0 + \gamma_1 D + \gamma_2 V_{max} + \gamma_3 T_x + \gamma_{12} D V_{max} + \gamma_{13} D T_x$$
$$+ \gamma_{23} V_{max} T_x + \gamma_{11} D^2 + \gamma_{22} V_{max}^2 + \gamma_{33} T_x^2 + \epsilon \qquad (5)$$

To determine the T_{hello} value for each measurement in Table 4, multi-objective optimization of models 3 and 4 is performed using the Minitab optimizer to obtain the most desirable value of T_{hello} by simultaneously minimizing EC and maximizing PDR as a function of the three values of V_{max}, N, and T_x with a constraint of PDR $>\simeq 50\%$ (if PDR can not reach 50%, the maximum of PDR is considered). The desirability function approach is a widely used technique to optimize multiple response processes. For responses $EC(D, V_{max}, T_x, T_{hello})$ and $PDR(D, V_{max}, T_x, T_{hello})$, two desirability functions assign numbers between 0 and 1 to the possible values of EC and PDR.

Figure 4 show a multi-objective optimization of EC and PDR for experiment number 10 in Table 4 where the input parameters are as follow: $D = 0$, $V_{max} = 1$ and $T_x = -1$. The adequate value of T_{hello} that gives the minimum EC and a PDR $>\simeq 50\%$ is $-0{,}85851$ (the corresponding natural value of this coded value is $1{,}56596$ s).
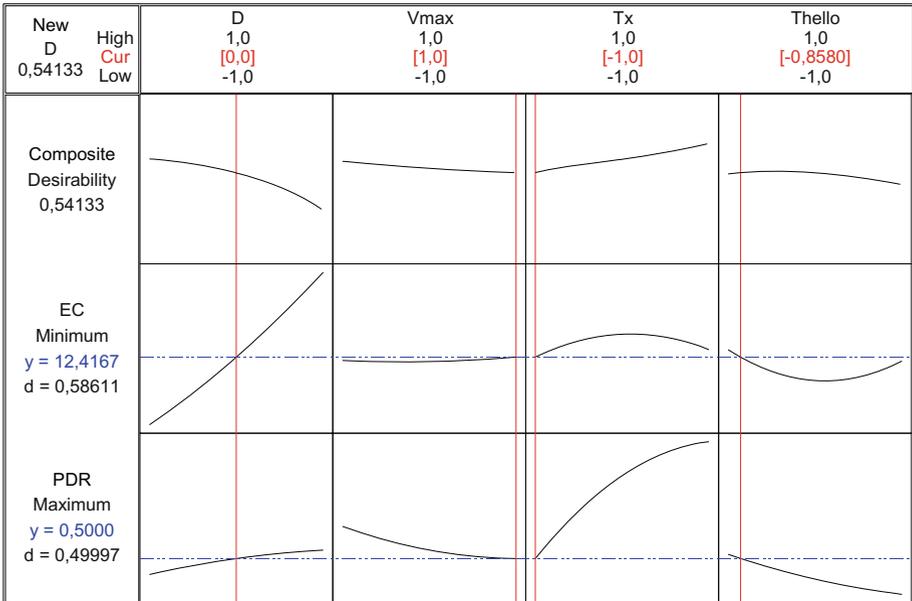
Fig. 4. An example of multi-objective optimization to get the best T_{hello} that minimize EC and have PDR $>\simeq 50\%$ for experiment number 10 in Table 4.

Before each mission launch, the routing protocol uses the density, maximum velocity, and transmission range of UAVs to calculate the T_{hello} value according to the obtained mathematical model.

Table 4. Models' coefficients

Exp	D	V_{max}	T_x	T_{hello}
1	−1	−1	0	6.21212121212
2	1	−1	0	9
3	−1	1	0	5.559397571844
4	1	1	0	8.067465806056
5	−1	0	−1	1
6	1	0	−1	4.763512
7	−1	0	1	7.280620869832
8	1	0	1	9
9	0	−1	−1	6.149531274012
10	0	1	−1	1.56596
11	0	−1	1	9
12	0	1	1	8.165821423084
13	0	0	0	7.182265252804
14	0	0	0	7.182265252804
15	0	0	0	7.182265252804

The coefficients of the three mathematical models 3, 4, and 5 are mentioned in Table 5.

Table 5. The coefficients of the three mathematical models 3, 4, and 5

$Coef_i$	α_i	β_i	γ_i
0	9,10975	0,792199	7,182
1	5,79459	−0,00614362	1,347
2	0,684524	−0,0563534	−0,875
3	−1,46792	0,226534	2,496
4	−3,99595	−0,0406915	/
12	0,55949	−0,00026596	−0,07
13	−2,83876	−0,0500798	−0,511
14	−2,32823	−0,00611702	/
23	−0,125962	0,0246454	0,937
24	0,658595	−0,0193262	/
34	−2,50583	0,0200355	/
11	1,2758	−0,0156058	−0,341
22	0,414074	0,0305423	0,368
33	−2,80854	−0,0980526	−1,33
44	3,68067	0,0191947	/

4   Experimental Evaluation and Discussion of Results

To evaluate the proposed optimized routing protocol, we have compared its performances to those of the original AODV [2] and a recent version for FANETs called EE-HELLO [1], on many scenarios, in which the following effects are studied: Effect of number of UAVs (Density), Velocity of UAVs and transmission range of UAVs.

4.1   Metrics

The considered metrics for the performance evaluation of the compared routing protocols are routing overhead (ROH), average energy consumption (EC), and packet delivery ratio (PDR).

Average Energy Consumption is the average amount of consumed energy by a node in the network. EC is obtained by dividing the sum of differences between initial energy and the final energy level of each node, divided by the number of nodes. It is expressed in Joules.

$$EC = \frac{\sum_{i=1}^{n} IE_i - RE_i}{n} \tag{6}$$

where IE_i and RE_i are the initial and the remaining energy of the ith node, respectively and n is the number of nodes in the network.

Packet Delivery Ratio is the ratio of the total number of successfully delivered data packets at the destination divided by the total number of generated data packets.

$$PDR = 100 \times \frac{\sum_{i=1}^{c} rxPackets_i}{\sum_{i=1}^{c} txPackets_i} \tag{7}$$

where $rxPackets$ and $txPackets$ are the numbers of received and transmitted packets.

Normalized Routing Overhead is the total number of transmitted routing packets, including forwarded routing packets, divided by the total number of received data packets by the destination.

$$ROH = \frac{txRoutingPackets}{\sum_{i=1}^{c} rxPackets_i} \tag{8}$$

where $txRoutingPackets$ is the number of transmitted routing packets.

4.2   Experimental Parameter Settings

The proposed protocol is a modified version of AODV of the NS-3 network simulator. The simulation time is set to 300 s for each scenario run. UAVs are randomly distributed on a 3D space and move according to the Gauss Markov Mobility Model. Each UAV is equipped with a Wi-Fi device. Two UDP applications (Client and Server) are installed on each UAV and each UAV chooses randomly its destination to send it data packets via the client application. The client application sends every five (5) seconds a UDP data packet of 512 Bytes to its destination, its launch time is randomly chosen between the first 50 and 55 s from the beginning of the simulation and ends after 280 s of running. We used ten (10) runs for each experiment, and each obtained performance result is the average of the ten runs. Table 6 shows the simulation parameters.

Table 6. Simulation parameters

Parameter	Value
Simulation time (s)	300
Minimum velocity (m/s)	5
Maximum velocity (m/s)	Default: 30 (10,20, 30, 40, 50)
Number of UAVs	Default: 30 (20,25, 30, 35, 40)
Mission area (Km * Km * Km)	0.6 * 0.6 * 0.15
Mobility model	Gauss Markov 3D
Type of connections	UDP
Data packet size (Bytes)	512
MAC/PHY protocol	IEEE 802.11b
DsssRate (Mb/s)	1
Radio propagation delay model	Constant speed
Radio propagation loss model	Friis
Tx range (m) (Controlled by TxPowerStart and TxPowerEnd)	Default 150 (150, 225, 300, 375, 450)
TxPowerStart, TxPowerEnd (dbm)	Default 4.41 (4.41, 7.93, 10.43, 12.37, 13.95)
wifi device TxCurrent (A)	0.0147
wifi device RxCurrent (A)	0.0097
wifi device SwitchingCurrent (A)	0.000426
wifi device IdleCurrent (A)	0.000426
Initial battery (Joule)	Unlimited (200000)
Protocols	OEM-AODV, EE-HELLO-AODV, AODV

4.3   Results and Discussion

– **Effect of number of nodes:** Figures 5, 6, and 7 show a comparison between the OEM-AODV, EE-HELLO-AODV, and AODV protocols in terms of the impact of the number of UAVs respectively on the routing overhead (ROH), the energy consumption (EC), and the packet delivery ratio (PDR).

First, the higher the density of UAVs is, the greater the energy consumption and the routing overhead are, because in a dense network the number of exchanged control messages is higher which causes greater overhead and consequently more energy consumption.

It can be observed that for 40 UAVs the EC of the proposed OEM-AODV is significantly reduced when compared with EE-HELLO-AODV and AODV (around 10% of energy is saved). The same observation is for the routing overhead where it is smaller in OEM-AODV when compared with the other protocols (around 10% of overhead is reduced).

However, the packet delivery ratio is slightly smaller in OEM-AODV than the other protocols only when the acceptable delivery ratio of $\simeq 50\%$ is reached. These results are explained by using an adequate number of control messages that optimize energy consumption with trying to reach an acceptable limit of the delivery ratio.

– **Effect of velocity and transmission range:** Moreover, to provide further comparisons between the considered protocols. Figures 8, 10, and 12 plot the impact of velocity respectively on EC, ROH, and PDR. Figures 9, 11, and 13 plot the impact of transmission range respectively on EC, ROH and PDR.
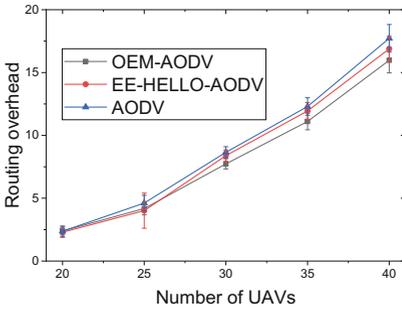
Fig. 5. Impact of density on ROH.
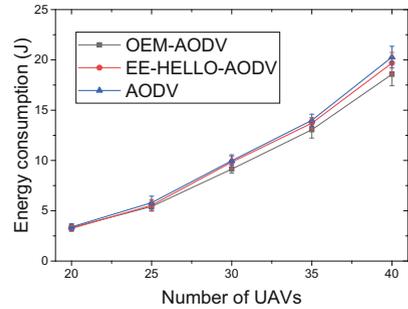
Fig. 6. Impact of density on EC.

For all considered performances the results show that the smaller the velocity is of UAVs the more ROH, EC, and PDR are improved for all the protocols. Similarly, the greater the transmission range the best is the PDR because packets join destination with less forwarding steps and consequently have less chance to be lost. However, Figs. 9 and 11 show that ROH and EC are increased for small to average transmission ranges and decreases for average to longer transmission
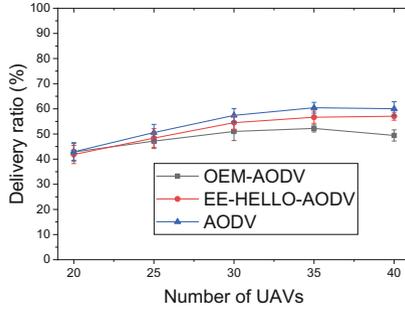
Fig. 7. Impact of density on PDR.

rages. This can be explained by the fact that the number of forwarded control messages is increased from short transmission ranges to average ones because UAVs transmission range cover more neighbors, however, forwarded control messages begin to decrease for longer transmission ranges because forwarded packets join rapidly their destinations. When we compare OEM-AODV with the other protocols, when varying velocity we can observe a significant enhancement of ROH (can reach 11% of reduced overhead) and EC (can reach 12% of saved energy) with keeping an acceptable PDR $> \simeq 50\%$ (reduced at most of 8%). When varying transmission range we can observe more significant enhancement of ROH (can reach 55% of reduced overhead) and EC (can reach 60% of saved energy) with keeping an acceptable PDR $> \simeq 50\%$ (reduced at most of 13%). This is always explained by using an adaptive and optimized frequency of sending control messages based on using mathematical modeling of performances by employing design of experiments techniques.
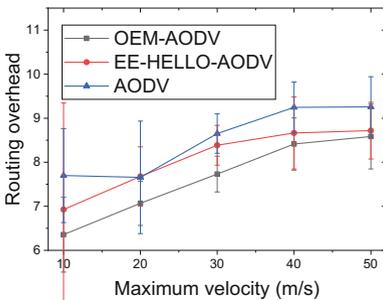
Fig. 8. Impact of velocity on ROH.
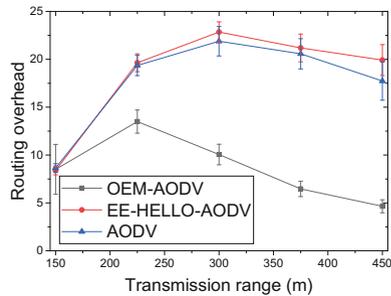
Fig. 9. Impact of Tx range on ROH.
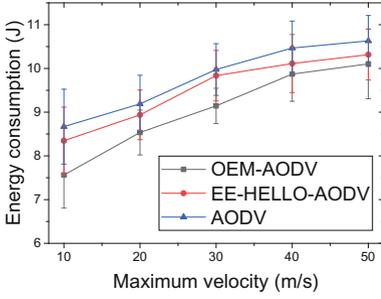
Fig. 10. Impact of velocity on EC.
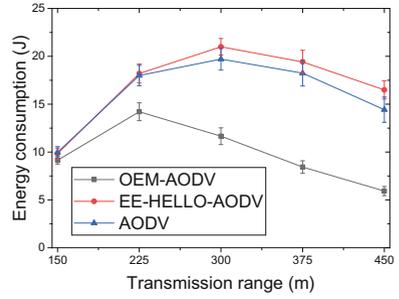
Fig. 11. Impact of Tx range on EC.
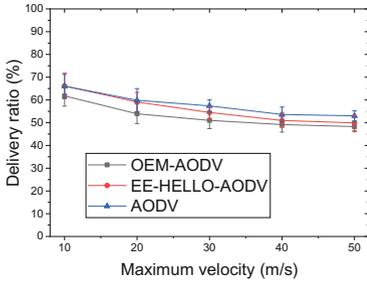
Fig. 12. Impact of velocity on PDR.
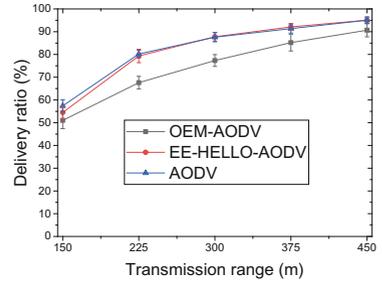
Fig. 13. Impact of Tx range on PDR.

5 Conclusions

In this paper, we considered the problem of energy consumption in mission-based unmanned aerial vehicles. To tackle this issue, we proposed an optimized routing protocol by adapting the Hello interval of AODV protocol using a predictive mathematical model. The proposed solution consists in using the Design of Experiments as a modeling and a multi-objective optimization tool. First, we used the Box-Behnken design to mathematically model the energy and the delivery ratio performances as a function of the influencing factors namely: the number of UAVs, their speed, their transmission range, and the Hello interval. Then, always by using a new Design of Experiments, we treated this problem as an inverse problem where the new objective is to model the Hello interval as a function of the other influencing factors while minimizing the energy and keeping an acceptable delivery ratio ($> \simeq 50\%$). Through different scenarios of the simulation experiments, we measured different metrics and compared the performance of the proposed protocol with the original AODV and with a recent adaptive protocol called EE-HELLO-AODV. Using the proposed protocol, the reduction of the overhead and the energy consumption is as possible minimized against a slightly decreased delivery ratio in data traffic. Given the importance of the energy metric in UAVs networks, we conclude that the tradeoff is considerably profitable and the proposed optimization makes the protocol balanced in favor

of the energy consumption and the overhead while guaranteeing an acceptable minimum delivery rate in any mission scenario.

References

1. Mahmud, I., Cho, Y.: Adaptive hello interval in FANET routing protocols for green UAVs. IEEE Access **7**, 63004–63015 (2019)
2. Perkins, C.E., Royer, E.M.: Ad-hoc on-demand distance vector routing. In: Proceedings WMCSA 1999, Second IEEE Workshop on Mobile Computing Systems and Applications, pp. 90–100 (1999)
3. Nawaz, H., Ali, H.M., Laghari, A.A.: UAV communication networks issues: a review. Arch. Comput. Methods Eng. 1–21 (2020)
4. Khan, M.F., Yau, K.L.A., Noor, R.M., Imran, M.A.: Routing schemes in FANETs: a survey. Sensors **20**(1), 38 (2020)
5. Popescu, D., Stoican, F., Stamatescu, G., Chenaru, O., Ichim, L.: A survey of collaborative UAV-WSN systems for efficient monitoring. Sensors **19**(21), 4690 (2019)
6. Li, X., Deng, F., Yan, J.: Mobility-assisted adaptive routing for intermittently connected FANETs. In: IOP Conference Series: Materials Science and Engineering, vol. 715, p. 012028 (January 2020)
7. Liu, J., Wang, Q., He, C., Jaffrès-Runser, K., Xu, Y., Li, Z., Xu, Y.: QMR: Q-learning based multi-objective optimization routing protocol for flying ad hoc networks. Comput. Commun. **150**, 304–316 (2020)
8. Darabkh, K.A., Judeh, M.S., Salameh, H.B., Althunibat, S.: Mobility aware and dual phase AODV protocol with adaptive hello messages over vehicular ad hoc networks. AEU - Int. J. Electron. Commun. **94**, 277–292 (2018)
9. Zheng, X., Qi, Q., Wang, Q., Li, Y.: An adaptive density-based routing protocol for flying ad hoc networks. In: AIP Conference Proceedings, vol. 1890, p. 040113. AIP Publishing LLC (2017)
10. Hernandez-Cons, N., Kasahara, S., Takahashi, Y.: Dynamic hello/timeout timer adjustment in routing protocols for reducing overhead in MANETs. Comput. Commun. **33**(15), 1864–1878 (2010)
11. Barton, R.R.: Response Surface Methodology, pp. 1307–1313. Springer, Boston (2013)
12. Daas, M.S., Chikhi, S.: Response surface methodology for performance analysis and modeling of MANET routing protocols. Int. J. Comput. Netw. Commun. **10**, 45–61 (2018)
13. Daas, M.S., Chikhi, S.: Optimizing geographic routing protocols for urban VANETs using stigmergy, social behavior and adaptive C-n-F mechanisms: an optimized CLWPR. Veh. Commun. **14**, 97–108 (2018)
14. Yang, P., Chen, H., Liu, W.Y.: Application of response surface methodology and desirability approach to investigate and optimize the jet pump in a thermoacoustic stirling heat engine. Appl. Therm. Eng. **127**, 1005–1014 (2017)

Dynamic Clustering Based Energy Optimization for IoT Network

Mohamed Sofiane Batta[1,2(✉)] , Hakim Mabed[1] , and Zibouda Aliouat[2]

[1] FEMTO-ST Institute/DISC, University of Bourgogne Franche-Comte,
Montbeliard, France
{mohamed.batta,hmabed}@femto-st.fr
[2] LRSD Laboratory, Computer Science Department,
Ferhat Abbas University Setif 1, Setif, Algeria
{mohamedsofiane.batta,zaliouat}@univ-setif.dz

Abstract. The arrival of modern 5G networks is expected to witness a massive increase in internet connections and base stations. Energy optimization is one of the imminent worldwide issues for green computing. Most of the energy efficiency existing works only focus on short-term vision of energy consumption and fails to contemplate the rechargeable battery degradation when evaluating the network lifetime. In this context, we introduce LTE2C, a new Long-Term Energy Efficient Clustering approach for dynamic IoT networks. The objective is to consider the batteries' degradation process and its state of health (SoH) to improve the network lifetime in long-term and reduces the number of required Internet connection. Several simulation scenarios have been conducted to analyze the performance of our clustering scheme. The obtained results show that the proposal reduces the clusters cardinality and significantly improves the network lifetime in long-term.

Keywords: IoT · WSN · Dynamic clustering · Rechargeable battery lifetime · Energy-aware protocols · Network connectivity

1 Introduction

Wireless sensor networks (WSNs) witness a large success over the past decades due to their rapid advances in technology and their promising development. These networks are mainly used to monitor harsh environments, composed of a variety of remotely deployed micro-sensors equipped with limited and non-rechargeable power resources. The emergence of new applications (smart houses, smart cities, smart transportation systems, smart healthcare) requires some intrinsic functionality, other than mere sensing. Therefore, following the WSNs, IoT technology brings a new vision of the next generation wireless networks where diversity of modern and smart heterogeneous devices (things) appear embedded with multiple communication interfaces and use-case scenarios. These

devices collaborate together and communicate with internet services (Cloud computing or data centers) to achieve a common goal. IoT networks are proposed with higher diversities in wireless communication to support a large range of real world applications with the ability to maintain the high dynamicity of the network topology.

The energy storage system assigned to IoT devices can significantly limit network longevity. Energy preservation is one of the dominant and monolithic constraints in wireless networks. Although clustering techniques prolong network longevity, most of the existing energy efficient work in this field only focus on short-term vision based on the state of charge (SoC) of non-rechargeable batteries and fails to address the rechargeable battery degradation aspect in their clustering strategy (long-term vision). Battery aging is one of the most frequent failures that lead to material substitution [1]. Therefore, to fulfill the modern network requirement, saving energy must be considered for rechargeable batteries to encourage the development of ecosystems and reduces the material replacement frequency, which impacts on the green computing environment. Among the various energy efficient proposed techniques, this paper discusses the lifetime extension using an original long-term energy optimization vision that enhances the network lifetime by taking into consideration the rechargeable batteries degradation and their State of Health (SoH) during the clustering of the network. The multi-hop clustering proposal reduces the number of generated clusters, which reduces the number of Base Stations (BSs) connection required to get internet access. Moreover, it balance the load supported by network devices which minimize the battery degradation.

Many stress factors influence the rechargeable battery lifespan [2], depending on how the battery is used such as charge/discharge rate, depth of discharge (DoD), temperature, voltage, etc. In this work, we use a realistic non-linear model of the rechargeable battery degradation adopted from [3] to simulate the battery aging process.

The rest of the paper is structured in the following manner. Section 2 presents some related works. The network concept and battery aging process are described in Sect. 3. Section 4 provides a presentation of our proposal. The simulation settings and the obtained results are described in Sect. 5. We conclude in Sect. 6.

2   Related Works

Clustering techniques have demonstrated many advantages to enhance the scalability and prolong the network lifetime [4]. Energy efficient clustering solutions for wireless networks are highlighted by many researchers, in order to prolong the network longevity [5]. However, the majority of these protocols only handle short-term vision of energy conservation for battery constrained devices. According to the post network structure, clustering approaches can be classified into two categories: single-hop and multi-hop clustering.

2.1   Single-hop Approaches

One of the most traditional and classic single-hop clustering approaches is the low energy adaptive clustering hierarchy (LEACH) [6]. In LEACH protocol, the role of CHs rotates periodically and randomly among network devices for energy perspective. Multiple variants of LEACH have been observed in the literature focusing on various performance constraints [7]. Some approaches integrate the idea of bio-inspired decision making in there clustering proposal. Wu et al. [8] use an ant colony optimization to provide an energy efficient clustering approach. The proposal aims to find a minimal set of CHs and preserve the battery SoC by load-balancing the number of devices among network clusters.

A smart and balanced energy-efficient clustering algorithm (Smart-BEE) for IoT Networks is proposed in [9]. The authors approached the clustering scheme from another perspective, where they adapt the multiple communication interfaces enabled in 5G networks (using Multiple-In and Multiple-Out (MIMO) communications) to perform energy efficient topology, maintain the network coverage and improve the Quality of user Experience (QoE). However, the proposal is assumed to be centralized which decreases the performance in case of a high dynamic network and makes it vulnerable to the central point of failure problem. A fuzzy based energy efficient clustering protocol (FEEC-IIR) for WSN and assisted IoT systems is presented in [10]. A multi-criteria decision-making technique is adopted and the CHs selection is performed based on energy status (SoC), QoS's impact, and node location to reduce packet traffic and save energy.

With the single-hop clustering, connectivity is difficult to guaranty and isolated nodes may appear because of the limited coverage area of network devices. Multi-hop clustering is an appropriate solution for dynamic networks, as the Cluster Members (CMs) and their CHs communicate in multi-hop fashion, cluster heads election and re-affiliations event are limited [4] which improve clusters coherence.

2.2   Muti-hop Approaches

DWEHC [11] is a weight based technique proposed for multi-hop clustering. This proposal is an improvement of a well-known clustering protocol (HEED [12]) by supporting multi-hop intra-cluster communication. The proposal uses devices' battery SoC to perform a load balanced hierarchy. DWEHC control the Cluster Members (CMs) cardinality, and therefore, achieve more balanced power consumption on each device. The goal of the proposal is to generate a minimum energy topology [13]. However, each node is assumed to have a limited number of routing children, which may generate isolated nodes in high density networks. Authors in [14] present a reliable energy efficient manner for intra-clustering in wireless networks (RINtraR). The proposal focuses on optimizing the QoS of wireless communication to reduce energy waste. Each node selects the path that consumes the least energy leading to the BS by assigning a transmission quality value to each communication link that relays it with the neighbors. The clustering process of RINtraR is inspired from LEACH which leads nodes with high connectivity to drain faster since they will be used to relay a high rate of

data. Ghosh et al. [15] presented another scheme of multi-hop clustering where network devices are assumed to be organized into a chain structure on which all data are transmitted. The goal of the proposal is to find the minimal set of dominating CH nodes. The drawback of the proposal is the high time complexity required to arrange network nodes into a chain.

The emergence of rechargeable battery devices also reveals new promising concept such as Rechargeable Wireless Sensor Networks (RWSN), where devices use the ambient energy (sunbeams, wind force) to power their battery. However, the availability of these sources needs to be considered because some devices may stop operating in case of unavailability of renewable resources, which decreases the network performance. Several techniques were proposed to alleviate these problems. For example, in [16] authors assume that devices with discharged batteries broadcast a charging request to a mobile charger that moves to respond to the corresponding requests.

Although several energy-aware clustering protocol were proposed, the overall of these works either consider non-rechargeable batteries or attempts to reduce the energy consumption (based on batteries SOC), whereas the aspect of battery aging is ignored. With the emergence of new complex devices, rechargeable batteries become commonly used and need to be considered. In this context, we introduce a novel long-term vision of energy optimization to extend the endurance of network devices. The proposal takes into consideration the SOC and the SOH of rechargeable batteries and integrates a multi-hop clustering strategy to tolerate the network dynamic topology and improves clusters stability.

3   Network Modelling and Rechargeable Battery Degradation Model

To simplify the understanding of distributed systems, wireless networks are typically modeled into graphs $G = (V, E)$ with a set of vertices V representing the network devices and a set of edges E corresponding to the wireless communicating links between nodes. Devices use a shared wireless communication channels to communicate. Hence, they cannot receive from multiple senders neither send and receive at the same time. In this study, we use the UDG (unit disk graph) [17] connectivity model which assumes that communication links are symmetric. Each node can directly communicate with the BS to establish an internet access and transmit its data or relay its data through multi-hop fashion. In our proposal, the connection with BS is only required for the cluster heads so that the number of internet access can be reduced.

There is confusion when discussing rechargeable and non-rechargeable battery lifetime. Non-rechargeable batteries need to be replaced when their initial charge is completely depleted. Therefore, the battery SoC is used to indicate the remaining energy of the battery. However, rechargeable batteries can withstand multiple recharging cycles. These batteries also have a limited life and require replacement owing to aging. With rechargeable batteries, we use another indicator named SoH that represents the physical condition of the current battery. In general, damage of rechargeable batteries is due to the following parameters:

3.1   Calendar Aging

This factor represents the battery irrevocable degradation over time. It varies according to different stress factors like temperature and SoC, which are related to the device functioning conditions. The temperature accelerates the internal reactions and the battery self-discharge.

3.2   Cycle Aging

It reflects the life lost each time the battery cycles between charging/discharging. It also varies due to the battery utilization pattern. There exist other stress factors that also impact the battery degradation [2], such as the open circuit voltage V and the Depth of Discharge DoD.

For the needs of our tests, the rechargeable battery degradation model Φ is adopted from the model used in [3] and formulated as:

$$\Phi = (\phi_{DoD} + \phi_t) * \phi_T * \phi_\sigma \tag{1}$$

$\phi_{DoD}, \phi_t, \phi_T, \phi_\sigma$ represent the stress factor models (the DoD, calendar aging, the temperature and the SoC).

$$\phi_{DoD} = \alpha_{DoD} * DoD * e^{(\beta_{DoD} * DoD)} \tag{2}$$

$$\phi_t = \alpha_t * t \tag{3}$$

$$\phi_T = e^{\left(\alpha_T * (T - T_{ref}) * \frac{T_{ref}}{T}\right)} \tag{4}$$

$$\phi_\sigma = \alpha_\sigma * e^{\sigma - \sigma_{ref}} \tag{5}$$

Table 1 contains the parameter values used in the stress models.

Table 1. Stress model parameters for the battery degradation

Parameter	Value
α_{DoD}	0.05
β_{DoD}	0.03
α_σ	1.04
σ_{ref}	0.50
α_T	6.93 E-2
T_{ref}	25 °C
α_t	4.14 E-10/s

4   Proposed Approach

This part introduces a new heuristic for the multi-hop clustering in IoT networks (LTE2C: Long-Term Energy Efficiency Clustering). The novelty of the proposal is the integration of an original long-term energy optimization perspective to extend the rechargeable batteries' lifetime. The goal of the proposal is to find the smallest set of CHs to reduce long-range wireless communication and improve energy efficiency in short-term. On the other hand, we also take into consideration the battery degradation of network devices and their SoH to extend the network durability in long-term. The proposal balances the role of CHs and relays among network device to reduce the temperature devoted to the functioning conditions, which reduce the batteries aging and help devices to tolerate longer period. The proposal uses a weight based technique where the node with the highest weight is preferable to be elected as CH. The weight is computed based on the following metric:

1. **The Surrounding Connectivity Ratio** (SCR): This metric, introduced by our proposal, measures the relative connectivity of the current node with its neighborhood. It is performed as:

$$\Omega = \frac{\left| \left[\sum_{i=0}^{|N(i)|} N(j) \ / \ j \in N(i) \right] \cup N(i) \right|}{|N(i)| + 1}$$

$$SCR_i = |N(i)| - \Omega \tag{6}$$

2. **The Residual Energy Ratio** (SoC): It represents the current state of charge of the battery (remaining energy).

$$SoC_{ratio} = \frac{E_{resid}}{E_{init}} \tag{7}$$

E_{init} and E_{resid} are the initial charge and the current energy levels respectively.

3. **The Battery Health** (SoH): It decreases from 1 (new battery) to 0 (completely dead battery). The value of the SoH is performed as follows:

$$SoH = 1 - \Phi \tag{8}$$

Therefore, the weight value W is computed as:

$$W = \alpha * ACR + \beta * SoC + \gamma * SoH \tag{9}$$

The weighting coefficient (α, β, γ) specifies the importance of the corresponding metric. As the objective of the heuristic is to expand the network lifetime in long term, we assign the highest weighting coefficient to the battery SoH, which is considered as the most important metric due to its role in the extension of devices' battery lifetime. Both SCR and SoC have the same importance because they give the CH priority to nodes with high connectivity and residual energy, thus alleviates the interference and collision problems [18,19]. Hence, these two metrics are assigned the same weighting coefficient. After performing several experiment scenarios, we choose $\alpha = 0.6$, $\alpha = \beta = 0.2$.

To support the distribution of network nodes and the dynamicity of the network, LTE2C adopts a distributed clustering technique where nodes have only a local vision of their environment made by their information and those of the nearby neighbors. Received clustering information packets are stored in the neighbors' record list (NRL). The clustering information exchanged between neighbors are illustrated in Table 2. LTE2C consists of three phases: neighbors discovery, clusters formation and maintenance phase.

Table 2. NRL structure

Parameter	Description
ID_i	Identity of the device
W_i	Current weight of i
CH_i	Relative Cluster Head of i
W_CH_i	Weight of the CH dominating i
$Dist(i, CH)$	Distance separating i from its CH (measured in hops)
$|N(i)|$	Cardinality of node i's neighborhood

4.1 Neighbors Discovery

When a node decides to join the network it broadcasts a $HELLO$ discovery message. Neighbors that receive this message, update their local state and replie by sending an LSI (Actual State Information) beacon. Next, the new node updates the local information, computes its weight and broadcasts an LSI beacon, containing the new clustering information, to integrate the network. This technique allows nodes to conserve a coherent vision of their neighborhood.

4.2 Clusters Formation

The cluster formation process is divided into three rules that are described in Algorithm 1. First, nodes execute the first clause C1 and select the node that holds the maximum weight among the current neighbors and their dominating CH, according to NRL list. The greatest value of the selected node is compared with the local weight, the node that holds the highest weight is elected as CH.

Using the second clause C2, the elected CHs update their local state and broadcast an announcement message. The third clause C3 allows network nodes to determine the shortest routing path toward their CH in single-hop or multi-hop fashion. A maximum number of hops constraint is imposed during the formation of the clusters, the selected CH must not be at a distance farther than k hops. This constraint prevents the formation of handicap clusters. In the case where no node in the NRL list satisfies this obligation, the current node elects itself as CH to avoid isolation node scenarios.

Algorithm 1. Clusters formation phase

Code for each node i

Variables:
$N(i)$: set of neighbors node of i
$Highest\_W_j$: temporary variable used to find the weightiest node j in the k-hop neighborhood

(C1): If $Highest\_W_j \neq Max\_weight(\{j \in (N(i) \vee CH_{N(i)}) \wedge Dist(i,j) < k\} \bigcup \{W_i\})$
then

    If $Highest\_W_j = W_i$ **then**
      $CH_i = ID_i$
    else
      $Highest\_W_j = Max\_weight(j \in (N(i) \vee CH_{N(i)}) \wedge Dist(i,j) < k)$
      $CH_i = ID_j$

(C2): If $(CH_i = ID_i) \wedge Dist(i, CH_i) \neq 0$ **then**

    $Dist(i, CH_i) = 0$
    Update NRL list
    Broadcast $CH\_announcement$ beacon

(C3): If $(CH_i \neq ID_i) \wedge Dist(i, CH_i) \neq Min(Dist(j, CH_j) \mid j \in N(i) \wedge CH_i = CH_j) + Dist(i,j)$ **then**

    $Dist(i, CH_i) = Min(Dist(j, CH_j) \mid j \in N(i) \wedge CH_i = CH_j) + Dist(i,j)$
    Update NRL list

Update $W_i = \alpha * MCR + \beta * SoC + \gamma * SoH$
Broadcast LSI Beacon

4.3 Maintenance Phase

The maintenance allows network clusters to be more resistant to disconnection events that may occur during the execution of the distributed system. The exchanged control packet enables nodes to periodically monitor their environment. Therefore, when a node leaves the system subsequent to mobility, lack of energy or coverage failure, the maintenance is locally triggered by the neighbors to avoid the development of crippled topology. The outgoing node is deleted from the neighborhood NRL lists. Next, according to the role interpreted by

the leaving node, nodes either execute the clause C1 to re-elect a new CH or execute C3 to join another nearby CH and perform the necessary routes updates.

Figure 1 illustrates an execution example of LTE2C in a small network represented by 6 nodes (green nodes represent the CHs, node 6 joins the network). The blue tables show nodes current variables.
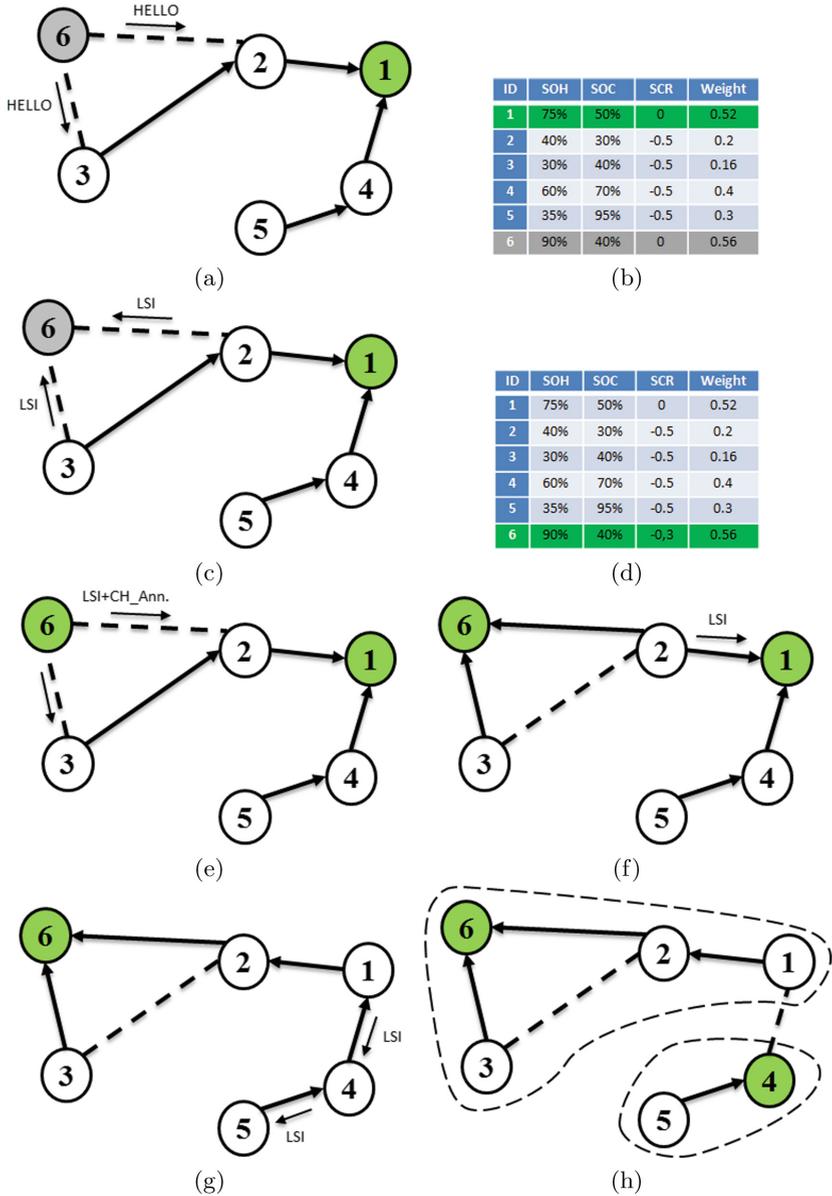
Fig. 1. Execution scenario of LTE2C via a small network

5   Simulation

The network modeling were conducted by simulation using a java based environment implemented to analyse the performances of the different approaches and allow the modeling, analysis and visualization of a wireless network as a graph. The values of the experiment parameters are listed in Table 3. The density of the simulated network is computed as $\lambda = \pi * Tr^2 * n/\Delta^2$. It is imperative to specify that the experiment is performed using the free space propagation model [20], where neither absorbing obstacles or reflecting surfaces are considered in the network area. In a real world environment, other models need to be addressed to satisfy the specification of the intended system. The energy consumption model for radio communication is adopted from [21]. In this study, we rather focus on how batteries are discharged. Network devices are assumed to be equipped with an integrated charger to meet the charging requests. The battery charging process starts when the SoC level falls below 10%. Devices temperature changes depending on the role achieved by the current device. To the best of our knowledge, our proposal is the first clustering technique that integrates the battery degradation and SoH. Hence, we choose to compare it with two protocols that belong to the same family of clustering, notably DWEHC [11] and RINtraR [14].

Table 3. Simulation parameters used in the experiment setting

Parameter	Value
Network size (Δ^2)	500 m × 500 m
Node density (λ)	[0, 30]
Distribution of nodes	Uniform random
Connectivity model	Unit Disk Graph (UDG [17])
Transmitting range (Tr)	25 m
BS position	(50, 50)
Maximum hop constraint k	{1, 2, 3}
E_{elec}	50 nJ/bit
ε_{FS}	10 pJ/bit/M^2
ε_{Mfs}	0.0013 pJ/bit/M^4
Data packet size	50 nJ/bit
Initial SoH	100% (1)
Initial energy	1 J

5.1   Experimental Results

Number of Cluster Heads
We choose the CHs cardinality as a metric to analyze the performance of the proposal because it allows the evaluation of the number of long-range wireless

communication and the number of required internet connection. The maximum number of hops constraint k = 2 and the comparison is performed using different density values. As it can be observed from Fig. 2, the cardinality of the proposal tends to be stable even when the density increases, this is due to the connectivity metric (SCR) used that favors nodes which not only have good connectivity, but their neighborhood also is well connected. Therefore, LTE2C shows an average improvement of 17.4% and 49.9% compared with DWEHC and RINtraR respectively.
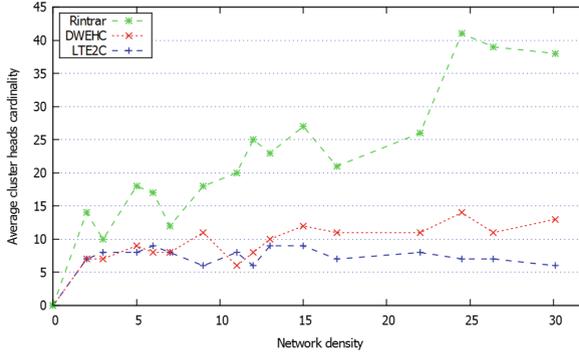
Fig. 2. Cluster heads cardinality under different network density values (k = 2).

The Average Energy Consumed
This metric illustrates the average energy consumed by network devices during the clustering phase in short-term. Following Fig. 3, we observe that the energy consumed increases along with the density. Initially, in the single-hop (Fig. 3(a)), the three approaches consume almost an equal amount of energy relative to the number of data packets exchanged. However, as the number of hops increases, the average energy consumption increases as well. With $k \geq 3$, devices using DWEHC require more energy to elaborate the clustered structure because DWEHC imposes a restriction in the number of routing children per node (limited to 6 children). Thus, more control packets are exchanged to generate the desired topology. On the other hand, RINtraR consumes the lowest amount of energy, thanks to its clustering mechanism based on random probabilities that do not require a lot of clustering information exchange. Our proposal consumes slyly more energy (5.9% more) than RINtraR to elaborate the network structure. However, the result is acceptable because the protocol shows better performance in long-term.

Average Number of Dead Nodes
The number of dead nodes is an important parameter to prove the longevity of the proposed technique and illustrate the average required time before a
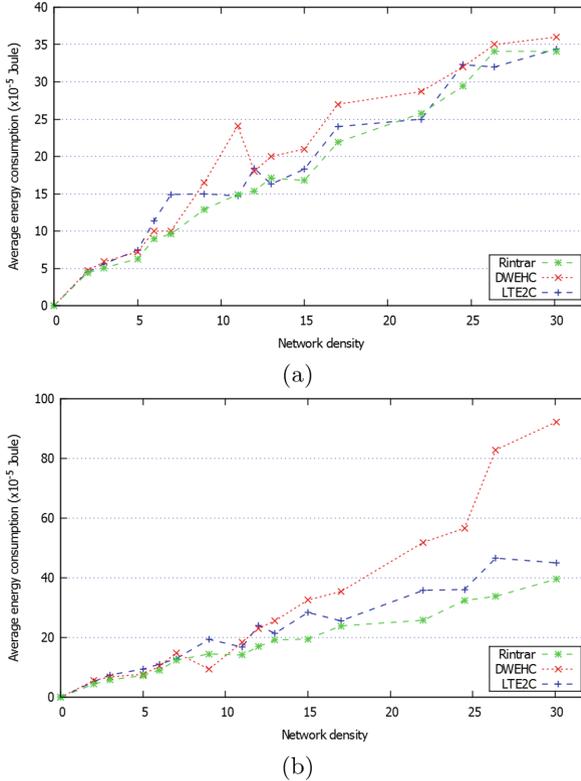
Fig. 3. Average energy consumption of the three approaches under different maximum hop constraints (a) k = 1. (b) k = 3.

rechargeable battery is completely dead and needs to be replaced. It develops a vision of the material substitution frequency. The inclusion of batteries degradation in the clustering process permitted network devices to last for longer period. Indeed, during the simulation, we observe that taking into consideration the batteries SoH helps to better balance the function of the CH among network nodes and manage battery temperature between adjacent devices. Thereby, the velocity of the battery aging process decrease and the durability of network devices increase in long-term. Figure 4 shows the average number of completely dead nodes during the simulation (measured in rounds) by using our proposal against DWEHC and RINtraR, the initial number of alive nodes is 240 ($\lambda = 30$). The usage of the residual energy in DWEHC allows devices using this protocol to resist longer than RINtraR (improvement of 5.8%). The proposed approach generated the smallest set of dead nodes 17.4% and 23.6% lower that DWEHC and RINtraR which reveals the impact on the energy optimization of our perspective in long-term.
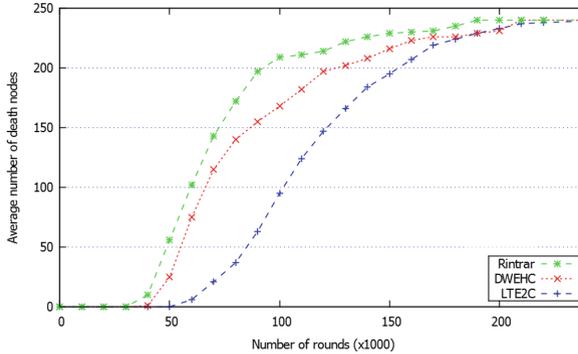
Fig. 4. Average number of dead nodes according to the number of rounds (k = 2)

6   Conclusion

When the vision of the forthcoming generation of IoT networks is fully realized, the scalability and maintainability cost will be enormous. Ignoring these challenges restricts the performance of the IoT solutions. The majority of IoT components are rechargeable battery-powered devices that eventually degrade and require maintenance. Although the battery itself might be cheap, the replacement implies a costly labor. Therefore, the effective management of devices batteries ought be applied to maintain network longevity. In this regard, we formulated the problem of dynamic clustering from a long-term energy efficiency perspective to improve the lifetime of IoT networks (LTE2C). We presented an original long-term vision of energy optimization for dynamic clustering while considering the aspect of rechargeable battery degradation and their State of Health. This work includes the effect of temperature on battery degradation to extend the lifespan of network devices and provide an influence on the green computing environment. The results obtained by simulation reveal that the proposal reduces internet access cardinality and provides network durability. As a future work, we plan to study the effect of increasing the battery charging cycle speed and analyze its impact on the battery aging in long-term to further improve the network lifetime.

References

1. Popovic, P.: Risk assessment of mobile phones using failure mode and effect analysis. In: International Scientific Conference on ICT and E-Business Related Research, pp. 213–220 (2016)
2. Ergun, K., Ayoub, R., Mercati, P., Rosing, T.: Dynamic optimization of battery health in IoT networks. In: IEEE 37th International Conference on Computer Design (ICCD), pp. 648–655 (2019)
3. Xu, B., Oudalov, A., Ulbig, A., Andersson, G., Kirschen, D.: Modeling of lithium-Ion battery degradation for cell life assessment. IEEE Trans. Smart Grid **9**(2), 1131–1140 (2018)

4. Xu, L., Collier, R., O'Hare, G.M.: A survey of clustering techniques in WSNs and consideration of the challenges of applying such to 5G IoT scenarios. IEEE Internet Things J. **4**(5), 1229–1249 (2017)

5. Sucasas, V., Radwan, A., Marques, H., Rodriguez, J., Vahid, S., Tafazolli, R.: A survey on clustering techniques for cooperative wireless networks. Ad Hoc Netw. **47**, 53–81 (2016)

6. Palan, N., Barbadekar, B., Patil, S.: Low energy adaptive clustering hierarchy (leach) protocol: a retrospective analysis. In: International Conference on Inventive Systems and Control (ICISC), pp. 1–12 (2017)

7. Xiao, M., Zhang, X., Dong, Y.: An effective routing protocol for energy harvesting wireless sensor networks. In: IEEE Wireless Communications and Networking Conference (WCNC), pp. 2080–2084 (2013)

8. Wu, C.W., Chiang, T.C., Fu, L.C.: An ant colony optimization algorithm for multi-objective clustering in mobile ad hoc networks. In: IEEE Congress on Evolutionary Computation (CEC), pp. 2963–2968 (2014)

9. Xu, L., O'Hare, G.M., Collier, R.: A smart and balanced energy-effiient multi-hop clustering algorithm (smart-beem) for MIMO IoT systems in future networks. Sensors **17**(7), 1574 (2017)

10. Preeth, L., Dhanalakshmi, R., Kumar, R., Shakeel, M.: An adaptive fuzzy rule based energy efficient clustering and immune-inspired routing protocol for WSN-assisted IoT system. J. Ambient Intell. Hum. Comput., 1–13 (2018)

11. Ding, P., Holliday, J., Celik, A.: Distributed energy-efficient hierarchical clustering for wireless sensor networks. In: International Conference on Distributed Computing in Sensor Systems, pp. 322–339 (2005)

12. Gupta, P., Sharma, A.K.: Clustering-based optimized heed protocols for WSNs using bacterial foraging optimization and fuzzy logic system. Soft Comput. **23**(2), 507–526 (2019)

13. Panda, B.S., Shetty, P., Pandey, A.: k-distinct strong minimum energy topology problem in wireless sensor networks. In: International Conference on Distributed Computing and Internet Technology, pp. 187–192 (2015)

14. Xu, L., O'Grady, M.J., O'Hare, G., Collier, R.: Reliable multihop intra-cluster communication for wireless sensor networks. In: International Conference on Computing, Networking and Communications (ICNC), pp. 858–863 (2014)

15. Ghosh, S., Mondal, S., Biswas, U.: Enhanced pegasis using ant colony optimization for data gathering in WSN. In: International Conference on Information Communication and Embedded Systems (ICICES), pp. 1–6 (2016)

16. Zhao, C., Zhang, H., Chen, F., Chen, S., Wu, C., Wang, T.: Spatiotemporal charging scheduling in wireless rechargeable sensor networks. Comput. Commun. **152**, 155–170 (2020)

17. Lebhar, E., Lotker, Z.: Unit disk graph and physical interference model: putting pieces together. In: IEEE International Symposium on Parallel & Distributed Processing, pp. 1–8 (2009)

18. Batta, M.S., Harous, S., Louail, L., Aliouat, Z.: A distributed TDMA scheduling algorithm for latency minimization in internet of things (2019)

19. Batta, M.S., Aliouat, Z., Harous, S.: A distributed weight-based TDMA scheduling algorithm for latency improvement in IoT. In: IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON) (2019)

20. Medbo, J., et al.: Radio propagation modeling for 5G mobile and wireless communications. IEEE Commun. Mag. **54**(6), 144–151 (2016)
21. Al-Janabi, T.A., Al-Raweshidy, H.S.: Optimised clustering algorithm-based centralised architecture for load balancing in IoT network. In: International Symposium on Wireless Communication Systems (ISWCS), pp. 269–274 (2017)

Machine Intelligence and Data Science

Comparative Analysis of Machine Learning Algorithms for Early Prediction of Diabetes Mellitus in Women

Sumbal Malik[(✉)] [iD], Saad Harous [iD], and Hesham El-Sayed [iD]

College of Information Technology, United Arab Emirates University, Al Ain, UAE
sumbalmalik01@hotmail.com

Abstract. Diabetes is a chronic disease characterized by hyperglycemia where a person suffers from a high level of blood sugar, which leads to complications such as blindness, cardiovascular diseases, and amputation. It is expected that in 2040 the diabetic patients will reach 642 million globally. Hence considering this alarming figure there is a strong need to early diagnose and predict the symptoms of diabetes to save precious human lives. One possible way to diagnose this disease is to leverage machine learning algorithms. Machine learning has swiftly been infiltrating in various domains in healthcare. With the help of diabetes data, machine learning algorithms can find hidden patterns to predict whether a patient is diabetic or non-diabetic. This research aims to provide a comparative analysis of the performance and effectiveness of selected machine learning algorithms in predicting diabetes in women. We develop a predication framework and implemented ten different machine learning algorithms, namely: Naive Bayes, BayesNet, Decision Tree, Random Forest, AdaBoost, Bagging, K-Nearest Neighbor, Support Vector Machine, Logistic Regression, and Multi-Layer Perceptron. Experimental results procured for the Frankfurt hospital (Germany) dataset shows that K-Nearest Neighbor, Random Forest, and Decision Tree outperformed the other algorithms in terms of all metrics. We believe that our diabetes prediction framework will assist doctors to predict diabetes mellitus with high accuracy.

Keywords: Diabetes mellitus · Machine learning · Prediction · Support Vector Machine · BayesNet

1 Introduction

According to the statistics of the World Health Organization (WHO), 422 million people experience diabetes globally [1]. In 2019, the International Diabetes Federation (IDF) report stated that Saudi Arabia, the United Arab Emirates (UAE), and Egypt have the highest rate of diabetes in the Middle East [2]. Especially, the UAE has a diabetes prevalence of 16.3% for adults [3]. Diabetes is one of the chronic diseases characterized by a high sugar level, which leads to damage to heart, kidneys, eyes, blood vessels, and nerves. Therefore, the early prevention and detection of diabetes are very important to save precious human lives and to avoid complications.

Diabetes is dangerous for both men and women. Many symptoms of diabetes are similar in men and women however, some symptoms are more likely to affect women such as oral thrush, urinary tract infections (UTIs), and gestational. Besides, it is extremely difficult to diagnose the complications of diabetes in women [4].

The successful use of machine learning algorithms in health care can assist the medical experts and improves the efficiency of the health care system [5]. Machine learning algorithms have since been used to predict several diseases [6] including mental disorders, diabetes [7], and hypertension. Machine learning models especially, Naive Bayes, K-Nearest Neighbor, Support Vector Machine, Decision Tree, and Random Forest have great classification capabilities.

The goal of this research is to implement a diabetes mellitus prediction system for women, based on machine learning algorithms to diagnose whether a patient is diabetic or non-diabetic. Furthermore, we conduct a comparative analysis of ten machine learning algorithms to identify the most efficient and precise algorithm in predicting diabetes. This paper is organized into five sections. Section 2 presents an overview of the related work. Section 3 discusses the proposed framework and methodology. Section 4 analyzes the experimental results and discusses the effectiveness of our diabetes prediction framework. Finally, Sect. 5 elucidates the conclusion of the study.

2 Related Work

In this section, the state-of-the-art literature related to diabetes prediction is discussed. Several recent research studies are also highlighted in Table 1. In 2019, the research study [15] implemented and compared five data mining classification algorithms namely Multilayer Perceptron (MLP), Fuzzy Lattice Reasoning (FLR), BayesNet, JRip and C4.5 to predict diabetes. The experimental results obtained using the MV dataset highlighted that BayesNet and MLP took less computation time whereas in terms of accuracy the C4.5 and RJip outperformed the other algorithms. The Support Vector Machine (SVM) is one of the promising solutions of machine learning which analyzes the data and recognizes the pattern. Considering the importance and effectiveness of SVM, the authors [16] used the SVM supervised learning algorithm to classify diabetic patients. Firstly, the Principal Component Analysis (PCA) was used to preprocess and reduce the dimensionality of the dataset. Then the output of the preprocessed data was utilized as input for SVM for diabetes classification. The results concluded that diabetes diagnosis using SVM with PCA preprocessing technique showed better accuracy than the existing systems. Similarly, in another research study [17] the researchers used the Decision tree, K-Nearest Neighbors, Support Vector Machine, and Random forest to predict diabetes at different stages. The comparison results of this study stated that Random Forest outperformed other classification techniques.

In 2019, the authors Safae et al. [18] used four classification algorithms namely Neural Networks, Naive Bayes, Decision tree, and Support Vector Machines to predict diabetes in women. The experiments conducted on the Pima Indian Diabetes (PID) [19] dataset indicated that Support Vector Machines obtained the highest accuracy with 76.8%. The limitation of this study was the small PID dataset used to evaluate the findings of this research. Heydari et al. [20] compared five algorithms to predict type II diabetes

Table 1. Literature review of diabetes mellitus

Year	Diabetes type	Dataset	Contribution	Findings
2020 [8]	Type II	PID	Worked on Decision tree and Random Forest to assess type-II diabetes	Got 81.1% accuracy with the help of K-Fold and Cross-Validation
2020 [9]	Type I–II	PID	Did comparative analysis of decision tree, Naive Bayes, Logistic Regression, SVM and KNN to predict diabetes in women	Decision Tree got significant results
2019 [10]	Gestational Diabetes	Hospital Dataset	Used decision tree, logistic regression and DenseNet to predict diabetes in women	Results showed that DenseNet surpassed other algorithms
2019 [11]	Type I–II	PID	Implemented decision tree, SVM and Naive Bayes to detect diabetes at an early stage in women	Obtained results showed that Naive Bayes outperformed with highest accuracy of 76.30%
2019 [12]	Type II	PID	Conducted comparative analysis of four machine learning algorithms, Naive Bayes, Logistic Regression, Random Forest, and J48	Logistic Regression got highest accuracy 83% followed by Random Forest 82%
2018 [13]	Type I–II	PID	Used artificial neural network model with the Levenberg-Marquardt training algorithm	Proposed model got 82% accuracy
2018 [14]	Type I–II	PID	Improved class imbalance using SMOTE and then implemented the decision tree algorithm	Decision tree classifier obtained 94.70% classification accuracy

in Iran. The highest accuracy was showed by artificial neural networks followed by SVM and Nearest Neighbors. Similarly, in another study [21] the authors used Neural Networks on the PID dataset to diagnose diabetes in patients. The experimental results demonstrated that the proposed system performed well with five-fold cross-validation. In another study [4], the authors' used the dataset of a hospital in Luzhou, China to predict diabetes mellitus in patients. The neural networks, decision tree, and random tree were used for prediction. The results concluded that diabetes prediction with random forest could reach the highest accuracy of 80% when all the attributes of the dataset were used.

As per the literature review, we found two important research gaps in the existing diabetes prediction system: (1) less attention has been directed to predict diabetes in women [22] (2) almost 65% of research studies use Pima Indian Diabetes (PID) Dataset to predict diabetes, which is a very small dataset having only 768 instances. This small dataset can manifest issues related to generalization and class imbalance. This research study fills these gaps using the large diabetes dataset provided by Frankfurt hospital and improving the class imbalance problem.

3 Methodology

This research aims to build a predictive model to predict diabetes. Figure 1 illustrates the proposed framework. It consists of the following steps: (1) the diabetes dataset is downloaded. (2) The data is pre-processed and cleaned to build models. (3) The cleaned dataset is then divided into training and testing sets. (4) The predictive models are built for ten machine learning algorithms to predict diabetes. (5) Testing data is given to the model to test the model performance. (6) At this step, the model predicts diabetes. (7) The results of all the algorithms are evaluated. (8) The comparison is performed to determine the best algorithm.
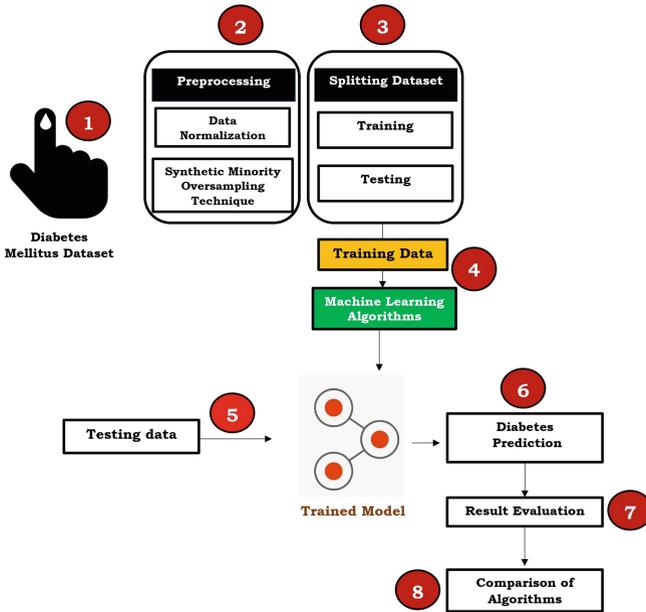
Fig. 1. Diabetes prediction framework.

3.1   Dataset

In this research, the diabetes dataset of Frankfurt hospital, Germany [23] is used to predict diabetes in women. The dataset consists of 2000 instances and 8 attributes. Table 2 gives a brief description of the dataset.

Table 2. Brief description of diabetes dataset

Sr. no	Attribute	Description
1	Pregnancy	Number of times a woman gets pregnant
2	Glucose	Plasma glucose concentration
3	Blood pressure	Reading of diastolic blood pressure (mm Hg)
4	Skin thickness	Triceps skin fold thickness (mm)
5	Insulin	2 h serum insulin
6	BMI	Body Mass Index (weight and height)
7	Age	Age of patient
8	Diabetes pedigree function	Probability to get the disease by extrapolating from patient ancestor's history

3.2   Preprocessing

Data preprocessing and cleaning are important stages to handle the data before using it in the machine learning algorithms. The diabetes dataset is available in CSV format. After downloading the dataset, we normalize it. The normalization is a technique to preprocess the data by bringing all the attributes under the common scale, which is under the common minimum, maximum and medium values without distorting the differences in the ranges of values. To normalize the diabetes dataset, we use the Z-Score Normalization technique [24]. This technique uses the mean and standard deviation of each feature of training data to normalize each input feature vector. Equation 1 shows the mathematical formula for Z-Score normalization, where Z represents the normalized attribute value, x_i original attribute value, μ mean, and σ is the standard deviation.

$$Z = \frac{x_i - \mu}{\sigma} \tag{1}$$

Since the diabetes data is an imbalance, so training a machine learning model with this imbalanced dataset often causes the model to develop a certain biasness towards the majority class. In order to avoid this biasness, we use the Synthetic Minority Over-sampling Technique (SMOTE) to balance the data properly. SMOTE is a very popular oversampling technique proposed by Chawla et al. [14] that operates on feature space. In SMOTE the instances of a minority class of the dataset are increased by creating new synthetic instances rather than by replication, hence it avoids the overfitting problem in machine learning algorithms. These instances are created by considering two parameters (1) oversampling rate, (2) the number of nearest neighbors K. In this study, we applied

the SMOTE with the following parameters: *nearestNeighbours = 5*, and *randomSeed value = 1*. For every minority instance, the algorithm takes $K = 5$ nearest neighbors (closest in distance) to the minority class (Euclidean distance). The SMOTE generates new synthetic instances ($instances_{NEW}$) by taking the difference between variables of minority sample and its nearest neighbors. Finally, the distance is then multiplied by a random number δ (between 0 and 1) and added to the variables value of the minority sample.

3.3 Machine Learning Algorithms for Prediction

After pre-processing the data, we split the data into 70% training and 30% testing datasets. The training data is given to machine learning algorithms to train the model. We build the prediction classifiers and the diabetes class attribute is used as the target variable. Ten machine learning algorithms: Naive Bayes [25], BayesNet, Decision Tree, Random Forest [26], AdaBoost, Bagging [27], K-Nearest Neighbor, Support Vector Machine, Logistic Regression [28], and Multi-Layer Perceptron [29] are trained and compared to predict diabetes mellitus. Table 3 shows the selected parameters used to train the algorithms. After training the classifiers, the model predicts diabetes in women using the testing data.

Table 3. Selected parameters for algorithms.

Algorithm	Parameter
BayesNet	Estimator = SimpleEstimator
Bagging	Classifier = RepTree
Support Vector Machine	Kernel = PolyKernel
Multilayer Perceptron	Batch Size = 100, Learning Rate = 0.3, Momentum = 0.2
K - Nearest Neighbor	k = 2

4 Experimental Results and Discussion

We conduct the experiments using Windows 10 operating system and WEKA 3.8.1 version. We used five performance evaluation metrics to evaluate the results of the diabetes prediction framework. The description and formulation of these metrics are shown in Table 4. These metrics are derived using confusion matrix [30] where True Positive (TP) represents the number of patients correctly classified as diabetic, True Negative (TN) is the number of patients correctly classified as non-diabetic. False Negative (FN) is the number of patients incorrectly predicted as non-diabetic whereas the False Positive (FP) is the number of patients incorrectly predicted as diabetic patients. The experimental results of the diabetes prediction framework and their graphical representations are shown in Table 5 and Fig. 2, respectively.

Table 4. Performance metrics description and formulation.

Metrics	Description	Formula
Time	Calculates the time algorithm takes to run	–
Accuracy	Compute the accuracy of the algorithm which is the rate of correctly classified instances from total instances	$\frac{TP + TN}{TP + TN + FP + FN}$
Precision	Classifier correctness is measured by precision which is the rate of correct predictions	$\frac{TP}{TP + FP}$
Recall	Used to evaluate classifier completeness	$\frac{TP}{TP + FN}$
F-Measure	Weighted average of precision and recall	$2 \times \frac{Precision \times Recall}{Precision + Recall}$

It is evident from Table 5 that K-Nearest Neighbor, Naive Bayes, and BayesNet takes less computational time 0.01, 0.02, and 0.04 s respectively to predict diabetes in women.

Besides, the results also indicate that the Random Forest, K-Nearest Neighbor, Decision Tree, and Bagging outperformed the other algorithms in terms of all performance metrics. It clearly shows that these machine learning algorithms can predict more accurately whether a patient is diabetic or non-diabetic.

To further show the effectiveness of our diabetes prediction framework, we compare the results with five previous research studies. These research studies used the Pima Indian Diabetes Dataset [19], which has the same attributes (Pregnancy, Glucose, Blood

Table 5. Experimental results of ten machine learning algorithms.

Category	Algorithm	Time (s)	Accuracy %	Precision %	Recall %	F-Measure %
Probabilistic	Naive Bayes	0.02	73.95	74.2	70	73.9
	BayesNet	0.04	79.54	81.9	74.8	78.2
Tree	Decision Tree	0.09	93.88	94.2	93.2	93.7
	Random Forest	0.57	**98.8**	**98.4**	**99**	**98.6**
	AdaBoost	0.11	77.16	77.8	74.8	76.2
	Bagging	0.19	93.47	94.5	92	93.3
Miscellaneous	K-Nearest Neighbor	0.01	**98.62**	**98.6**	**98.6**	**98.6**
	Logistic Regression	0.09	78.39	79.2	75.9	77.5
	SVM	0.05	75.29	74.1	76.2	75.2
	Multi-Layer Perceptron	1.08	80.47	82.2	76.7	79.4
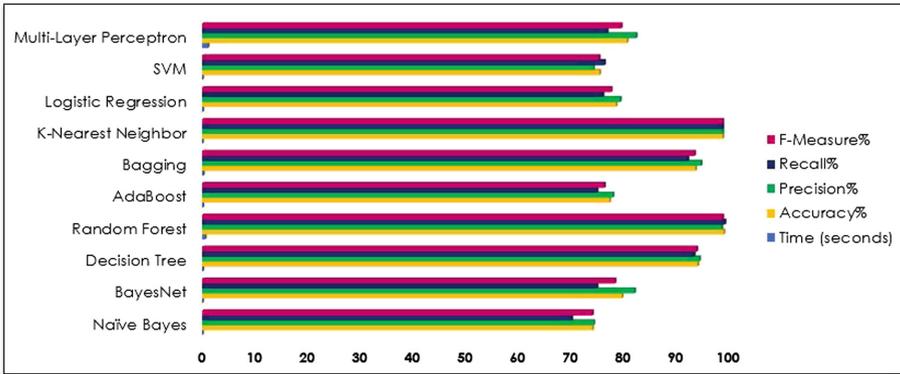
Fig. 2. Graphical representation of experimental results.

Pressure, Skin Thickness, Insulin, BMI, Diabetes Pedigree Function Age) as we have in our dataset but only 768 instances. Table 6 illustrates the extracted results of these studies.

Table 6. Comparison of our experimental results with previous studies.

Sr. no	Year	Machine learning algorithms	Dataset	Performance metrics				
				Time	Accuracy %	Precision %	Recall %	F-Measure %
1	2019 [32]	BayesNet	PID	N/A	73.83	64	57	60
2	2019 [33]	K Nearest Neighbors	Frankfurt Hospital Germany	N/A	97	N/A	100	N/A
		Decision Tree		N/A	95.50	N/A	90.91	N/A
		ANN		N/A	89	N/A	90.91	N/A
		Deep Neural Network		N/A	98	N/A	98.48	N/A
3	2019 [34]	Logistic Regression	PID	N/A	81.81	N/A	N/A	N/A
		K Nearest Neighbors		N/A	79.22	N/A	N/A	N/A
		Decision Tree		N/A	84.41	N/A	N/A	N/A
		AdaBoost		N/A	81.81	N/A	N/A	N/A

(*continued*)

Table 6. (*continued*)

Sr. no	Year	Machine learning algorithms	Dataset	Performance metrics				
				Time	Accuracy %	Precision %	Recall %	F-Measure %
4	2018 [35]	K Nearest Neighbors	PID	N/A	86.69	N/A	N/A	N/A
		SVM		N/A	80.85	N/A	N/A	N/A
		Random Forest		N/A	96.85	N/A	N/A	N/A
		Decision Tree		N/A	84.05	N/A	N/A	N/A
5	2018 [11]	Naive Bayes	PID	N/A	76.30	75.90	76.30	76
		SVM		N/A	65.10	42.40	65.10	51.30
		Decision Tree		N/A	73.82	73.50	73.80	73.60
6	Our study	Naive Bayes	Frankfurt Hospital Germany	0.02	79	82.2	74.7	78.3
		BayesNet		0.04	79.54	81.9	74.8	78.2
		Decision Tree		0.09	93.88	94.2	93.2	93.7
		Random Forest		0.57	98.8	98.4	99	98.6
		AdaBoost		0.11	77.16	77.8	74.8	76.2
		Bagging		0.19	93.47	94.5	92	93.3
		Logistic Regression		0.09	78.39	79.2	75.9	77.5
		SVM		0.05	75.29	74.1	76.2	75.2
		Multi-layer Perceptron		1.08	80.47	82.2	76.7	79.4
		K Nearest Neighbors		0.01	98.62	98.6	98.6	98.6

Note: N/A refers to parameters not considered by the studies

The highest accuracy obtained by implementing the algorithms listed in Table 5 shows that algorithms of tree category surpassed others. The Random Forest attained 98.80% accuracy followed by 98.4% precision, 99% recall, and 98.6% f-measure. The logic behind the Random Forest is the Bagging concept, which enhances the overall performance by adding the additional randomness to the model while growing the trees, which contributes to reaching a high accuracy. Despite the good performance of Random Forest, it takes the second-highest time 0.57 s to build and run the model. Another high-performance model that belongs to the tree category is the Decision Tree.

It achieved 93.88% accuracy, 94.2% precision, 93.2% recall, and 93.7% f-measure. It has been proved that the Decision Tree and Support Vector Machines work excellently in diagnosing diseases and classification problems [31].

The decision tree works on the principle, which iteratively breaks the dataset into two or more sample data. It optimally finds the root node based on the highest entropy value, which helps the tree to choose the most consistent hypothesis among the training set. In contrast to the Random Forest, the Decision Tree takes less time of 0.09 s. The last algorithm of tree class, which significantly performs well, is Bagging. A powerful technique combines the predictions from multiple machine learning algorithms to predict accurately diabetic and non-diabetic patients than the individual model. In our experiments, it also works well by achieving 93.47% accuracy followed by 94.5% precision, 92.0% recall, and 93.3% f-measure. In the probabilistic category, both algorithms, the Naïve Bayes, and BayesNet performed efficiently. In this study, the Naïve Bayes achieved 79% accuracy, 82.2% precision, 74.7% recall, and 78.3% f-measure when used with a supervised discretization filter. BayesNet gives remarkable results in various domains as it works on the concept of Bayesian networks and probabilities. In this research, the BayesNet gained 79.54% accuracy, 81.9% precision, 74.8% recall, and 78.2% f-measure which is significantly better than the study [32] shown in Table 6 where the BayesNet got 73.83% accuracy, 64% precision, 57% recall and 60% f-measure. The results of this study clearly show that our BayesNet algorithm performed best when we use the SMOTE technique to reduce the class imbalance problem. In the miscellaneous category, the K-Nearest Neighbor surpassed all algorithms. We experiment the K-Nearest Neighbor with different K values. It obtained the highest accuracy, precision, recall, and f-measure of 98.62%, 98.6%, 98.6%, and 98.6% respectively when the K value is to set to 2. The research studies in [33] and [34] (summarized in Table 6) show that they also implemented K-Nearest Neighbor to predict diabetes but obtained less accuracy 97% ($K = 2$) and 79.22% respectively than our results.

5   Conclusion

Diabetes mellitus is one of the chronic diseases today. The early detection and prevention of diabetes can help to avoid complications and save human precious lives. In this paper, we developed a prediction framework and analyzed the performance of ten popular machine learning algorithms in predicting diabetes in women. We have also improved the class imbalance problem of the dataset by using the SMOTE technique. Our experimental results show that the performance of the diabetes prediction system is improved with K-Nearest Neighbor, Random Forest, and Decision Tree with 98.62%, 98.8%, and 93.88% accuracy, respectively. The findings of this research demonstrate that leveraging SMOTE with these algorithms enhanced the effectiveness of the framework in helping doctors to predict diabetes accurately. In the future, we plan to extend this work by implementing several deep learning algorithms, which are expected to predict the types of diabetes with high accuracy.

References

1. Diabetes. https://www.who.int/health-topics/diabetes. Accessed 01 Mar 2020
2. Alzaabi, A., Al-Kaabi, J., Al-Maskari, F., Farhood, A.F., Ahmed, L.A.: Prevalence of diabetes and cardio-metabolic risk factors in young men in the United Arab Emirates: a cross-sectional national survey. Endocrinol. Diabetes Metab. **2**(4), e00081 (2019)
3. Alawadi, F., Abusnana, S., Afandi, B., Aldahmani, K.M., Alhajeri, O., Aljaberi, K., Alkaabi, J., Almadani, A., Bashier, A., Beshyah, S., et al.: Emirates diabetes society consensus guidelines for the management of type 2 diabetes mellitus–2020. Dubai Diabetes Endocrinol. J. (2020)
4. Zou, Q., Qu, K., Luo, Y., Yin, D., Ju, Y., Tang, H.: Predicting diabetes mellitus with machine learning techniques. Front. Genet. **9**, 515 (2018)
5. Yuvaraj, N., SriPreethaa, K.: Diabetes prediction in healthcare systems using machine learning algorithms on Hadoop cluster. Cluster Comput. **22**(1), 1–9 (2019)
6. Kourou, K., Exarchos, T.P., Exarchos, K.P., Karamouzis, M.V., Fotiadis, D.I.: Machine learning applications in cancer prognosis and prediction. Comput. Struct. Biotechnol. J. **13**, 8–17 (2015)
7. Dinh, A., Miertschin, S., Young, A., Mohanty, S.D.: A data-driven approach to predicting diabetes and cardiovascular disease with machine learning. BMC Med. Inform. Decis. Making **19**(1), 211 (2019)
8. Younus, M., Munna, M.T.A., Alam, M.M., Allayear, S.M., Ara, S.J.F.: Prediction model for prevalence of type-2 diabetes mellitus complications using machine learning approach. In: Data Management and Analysis, pp. 103–116. Springer (2020)
9. Agarwal, A., Saxena, A.: Comparing machine learning algorithms to predict diabetes in women and visualize factors affecting it the most—A step toward better health care for women. In: International Conference on Innovative Computing and Communications, pp. 339–350. Springer (2020)
10. Du, F., Zhong, W., Wu, W., Peng, D., Xu, T., Wang, J., Wang, G., Hou, F.: Prediction of pregnancy diabetes based on machine learning. In: The 3rd International Conference on Biological Information and Biomedical Engineering, BIBE 2019, pp. 1–6. VDE (2019)
11. Sisodia, D., Sisodia, D.S.: Prediction of diabetes using classification algorithms. Proc. Comput. Sci. **132**, 1578–1585 (2018)
12. Mirza, S., Mittal, S., Zaman, M.: Decision support predictive model for prognosis of diabetes using smote and decision tree. Int. J. Appl. Eng. Res. **13**(11), 9277–9282 (2018)
13. Zhang, Y., Lin, Z., Kang, Y., Ning, R., Meng, Y.: A feed-forward neural network model for the accurate prediction of diabetes mellitus. Int. J. Sci. Technol. Res. **7**(8), 151–155 (2018)
14. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: Smote: synthetic minority over-sampling technique. J. Artif. Intell. Res. **16**, 321–357 (2002)
15. Manimaran, R., Vanitha, M.: Prediction of diabetes disease using classification data mining techniques. Int. J. Eng. Technol. (IJET) (2017). ISSN (Print)
16. Aishwarya, R., Gayathri, P., et al.: A method for classification using machine learning technique for diabetes (2013)
17. Sowjanya, K., Singhal, A., Choudhary, C.: MobDBTest: a machine learning based system for predicting diabetes risk using mobile devices. In: 2015 IEEE International Advance Computing Conference (IACC), pp. 397–402. IEEE (2015)
18. Alaoui, S.S., Aksasse, B., Farhaoui, Y.: Data mining and machine learning approaches and technologies for diagnosing diabetes in women. In: International Conference on Big Data and Networks Technologies, pp. 59–72. Springer (2019)
19. Pima Indians Diabetes Database, Kaggle. https://www.kaggle.com/uciml/pima-indians-diabetes-database. Accessed 12 Mar 2020

20. Heydari, M., Teimouri, M., Heshmati, Z., Alavinia, S.M.: Comparison of various classification algorithms in the diagnosis of type 2 diabetes in Iran. Int. J. Diabetes Dev. Countries **36**(2), 167–173 (2016)
21. Ayon, S.I., Islam, M., et al.: Diabetes prediction: a deep learning approach. Int. J. Inf. Eng. Electron. Bus. **11**(2) (2019)
22. Chasan-Taber, L.: Lifestyle interventions to reduce risk of diabetes among women with prior gestational diabetes mellitus. Best Pract. Res. Clin. Obstet. Gynaecol. **29**(1), 110–122 (2015)
23. Diabetes, Kaggle. https://www.kaggle.com/johndasilva/diabetes. Accessed 04 Mar 2020
24. Jayalakshmi, T., Santhakumaran, A.: Statistical normalization and back propagation for classification. Int. J. Comput. Theor. Eng. **3**(1), 1793–8201 (2011)
25. Raj, R.S., Sanjay, D., Kusuma, M., Sampath, S.: Comparison of support vector machine and Naive Bayes classifiers for predicting diabetes. In: 2019 1st International Conference on Advanced Technologies in Intelligent Control, Environment, Computing & Communication Engineering (ICATIECE), pp. 41–45. IEEE (2019)
26. Zhang, B., Lu, L., Hou, J.: A comparison of logistic regression, random forest models in predicting the risk of diabetes. In: Proceedings of the 3rd International Symposium on Image Computing and Digital Medicine, pp. 231–234 (2019)
27. Parande, P.V., Banga, M.: Bagging for improving accuracy of diabetes classification. In: International Conference on Intelligent Computing and Communication, pp. 125–134. Springer (2019)
28. Zhu, C., Idemudia, C.U., Feng, W.: Improved logistic regression model for diabetes prediction by integrating PCA and K-means techniques. Inf. Med. Unlocked **17**, 100179 (2019)
29. Ahuja, R., Sharma, S.C., Ali, M.: A diabetic disease prediction model based on classification algorithms. Ann. Emerg. Technol. Comput. (AETiC) **3**(3), 44–52 (2019)
30. Luque, A., Carrasco, A., Martín, A., de las Heras, A.: The impact of class imbalance in classification performance metrics based on the binary confusion matrix. Pattern Recogn. **91**, 216–231 (2019)
31. Agrawal, R., Ghosh, S.P., Imielinski, T., Iyer, B.R., Swami, A.N.: An interval classifier for database mining applications. In: VLDB (1992)
32. Larabi-Marie-Sainte, S., Aburahmah, L., Almohaini, R., Saba, T.: Current techniques for diabetes prediction: review and case study. Appl. Sci. **9**(21), 4604 (2019)
33. Daanouni, O., Cherradi, B., Tmir, A.: Type 2 diabetes mellitus prediction model based on machine learning approach. In: The Proceedings of the 3rd International Conference on Smart City Applications, pp. 454–469. Springer (2019)
34. Khurana, G., Kumar, A.: Improving accuracy for diabetes mellitus prediction using data pre-processing and various new learning models (2019)
35. Farooqui, N., Mehra, R., Tyagi, A.: Prediction model for diabetes mellitus using machine learning techniques. Int. J. Comput. Sci. Eng. **6**(3) (2018)

Sentiment Analysis in Google Play Store: Algerian Reviews Case

Asma Chader(✉) , Leila Hamdad , and Abdesselam Belkhiri

Laboratoire de la Communication dans les Systèmes Informatiques (LCSI), Ecole Nationale Supérieure d'Informatique (ESI), BP 68M, 16309 Oued-Smar, Algiers, Algeria
{aa_chader,l_hamdad,fa_belkhiri}@esi.dz

Abstract. In mobile application stores, users very often rely on the opinions of others before downloading an application and its reputation could depend entirely on them. This makes analysis of users' reviews very interesting for application owners to take future decisions. In this paper, we are interested in analyzing Algerian reviews on application store using sentiment analysis. To the best of our knowledge, this is the first study that explores the Algerian context where reviews have the particularity of being written using different languages (French, Arabic and Algerian Dialect) making them difficult to process. We analyzed these reviews according to two existing approaches: Automatic approach based on machine learning and Lexicon-based one. Evaluation of the proposed solution is conducted on more than 50 000 reviews collected from popular Algerian applications on Google play store. The obtained results are very promising, we achieved an accuracy of 80% using the lexicon-based approach and of 72% for SVM on Dialect reviews.

Keywords: Reviews · Google play store · Sentiment analysis · Arabizi · Arabic · French

1 Introduction

In the highly competitive world of marketing we witness, understanding user sentiment towards products/services and reacting accordingly has become crucial to ensure future-proof businesses. This becomes ever more challenging in the context of mobile application (app) stores where most of the applications are free to download and a large variety of choices is offered for a specific type of apps. According to a recent study[1], the Google Play Store counts over 2,56 million apps, followed by the Apple's App Store with almost 1,85 million available apps as of the first quarter 2020. Therefrom, users often rely on others' reviews or recommendations to finalize their purchasing decisions. Those reviews are also valuable for developers to get feedback (e.g. feature requests, functional complaints, and privacy issues) and for app store owners to improve search ranking and detect malicious behaviors [1]. The rapidly increasing volume of reviews

---

[1] https://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores/.

(misleading/spam ones as well) makes it almost impossible to analyze them manually. Hence, over the last several years, app reviews were evaluated in different ways, from general exploratory studies [2] to classification [3], feature extraction [4], review filtering [5], to summarization [6]. Moreover, reviews usually contain user sentiments/emotion that can be seen as additional meaningful meta-data [7]. For instance, bug reports or privacy issues are often associated to a negative emotion while feature requests tend to be neutral; which may have a great value to identify priority messages notably. Therefore, automatic extraction and analysis of opinions, aka sentiment analysis (SA) is getting increasing interest. SA field can be seen as a classification task of whether a text is positive, negative or neutral [8] and have been applied to various kinds of texts including emails, blogs, newspaper headline, tweets and social media posts/comments. Existing approaches can be categorized along three axes [8, 9]: supervised approaches that are based on machine learning algorithms using manually annotated data, lexicon based approaches that rely on opinion vocabularies to detect polarity (by score computation according to opinion words present in the text) and hybrid approach that combines the two by, for instance using the lexicon based method to annotate data (instead of doing it manually) and then apply machine learning classifiers.

Reviews of mobile apps have some unique characteristics; they are generally short in length and specific to a particular app version and may contain device- or technology-specific information. Vocabulary remains one of the biggest challenges (for SA in general and in app store data mining particularly) because of its variations and different meanings a single word can have in different contexts and domains; informal and unstructured texts are also open and challenging tasks [1]. In the Algerian context, which is the focus of this paper, things are more complex. Reviews are in different languages: Modern Standard Arabic (MSA), French and Algerian dialect (AlgD) in both Arabic and Romanized transcriptions. This latter, called Arabizi, consists on using Latin alphabet, numbers or punctuation to write an Arabic word. Numbers are used to symbolize Arabic letters that have no phonetic Latin equivalent, such as number 3 for Arabic letter "ع" or number 7 for Arabic letter "ح" [8]. Unlike standardized languages such as MSA, French or English, there are no specific and predefined rules to write Arabizi; the variety of Algerian accents that differ from one region to another increases as well the complexity of its analysis.

Despite the growing interest on Arabic sentiment analysis (ASA) in recent years, the app store context is not well explored yet [9], even less the Algerian one; to the best of our knowledge, there is not any work done on Algerian app reviews. Moreover, only limited work addressed Arabizi [8]. In this paper, we propose two approaches (machine learning and lexicon-based) to classify app store reviews. We consider different languages (French, MSA and AlgD in both Latin and Arabic scripts) and describe the process followed to perform sentiment analysis, starting by data fetching from the Google play store and applying different preprocessing (depending on language) and classification methods to finally comparing result among the different algorithms and approaches. The next section describes works most related to ours and Sect. 3 our proposed approaches. Section 4 presents evaluation results and Sect. 5 concludes the paper with some future directions.

2 Related Work

We focus the related work discussion in two areas: mining user feedback from application store data, and sentiment analysis field that we present with respect to the three main approaches, lexicon based, machine learning based and hybrid approach.

Many studies have focused on reviews from mobile app stores [1–7]. Topic modeling and sentiment analysis are often applied with various focuses such as review categorization, summarization, feature extraction, users' ratings or even user/developer dialogue analysis [10]. Hoon et al. [11] analyzed 8,7 million user reviews of mobile apps and found that when users give a negative review to an app, the length of the feedback is greater. Wang et al. [12] studied mobile app updates and use a k-means clustering algorithm on over 17,5 million reviews to identify seven patterns based on the feature intensity trend between two neighboring updates. McIlroy et al. [13] studied the nature of more than 600 000 reviews from 20 mobile apps, and proposed an automatic multi-labelling approach.

Regarding sentiment analysis, Genc et al. provide a systematic literature review on opinion mining from Play Store Reviews [1]. Martens and Johann [7] performed an exploratory study of the emotional sentiment of 7 million reviews from the Apple App Store. Similarly, Li et al. proposed an approach to analyze the changes of user's sentiment through continuous app updates by detecting the similar topics before and after update of up to 150 000 reviews [14]. In a recent study, Soumik et al. performed sentiment analysis using text classification and various machine-learning algorithms on Bangla dataset which consisted of about 10 000 reviews and achieved a 76.48% accuracy with SVM [15]. Saudy et al. [9] surveyed the field of Arabic app store sentiment analysis for Requirements Evolution and underlined the lack of studies in this direction. The Arabic app store context is not well explored yet and no work has been done on Algerian app reviews yet.

Current approaches to sentiment analysis include machine learning methods that use annotated datasets to train classifiers, lexicon based methods that use a vocabulary of opinion words to predict text polarity, and hybrid methods that combines the two.

Wang et al. [16] compared three ensemble techniques viz. bagging, boosting, and random subspace methods on five algorithms: NB, ME, DT, kNN, and SVM for classification of sentiment on an 8 030 instances corpus. They reported 79% accuracy with the TF-IDF representation for SVM classifier. Likewise, [17] examined different classifiers on Arabic corpus of 3 700 tweets manually annotated and showed that SVM with 72% F-measure outperformed k-NN, NB and DT. Few work addressed Arabizi dialects. There are for example [18] in Maghrebi and [19] in Tunisian dialects. In AlgD Arabizi, Chader et al. [8] proposed a supervised approach using different classifiers and reported an F-measure of 87% using SVM on a corpus of 16 580 messages. This work also investigated the impact of several preprocessing techniques, dealing with dialect specific aspects.

As for lexicon based methods, Taboada et al. [20] proposed a method using a dictionary of sentiment words and their semantic orientations varied between -5 and $+5$. They also incorporated amplifiers and negation; which achieved an accuracy of 90% on 5 100 text reviews. [21] presented an approach for AlgD using three lexicons (keywords

lexicon, negation, and intensification words lexicon) enriched by dictionary of emoticons. The approach was tested under different configurations and achieved a 79.13% accuracy on about 7 700 Facebook comments.

Finally, in the hybrid method, [22] presented an approach that consists of annotating the training corpus using a look-up table stemming technique (lexicon-based phase) and then using the corpus to train SVM and NB classifiers, they achieved an accuracy of 96% on the MIKA corpus consisting of about 4 000 topics. Similarly, bettiche et al. [23] used a hybrid approach to Arabizi AlgD comments sentiment analysis. They reported an F-measure of 93.7% on a corpus of 2 650 messages about politics from YouTube videos.

As can be noticed from related work, different approaches have been proposed in both sentiment analysis and mining app store reviews fields. However, only limited research looked at Arabic sentiment analysis of app store reviews; which is the focus of this paper.

3   Proposition: Sentiment Analysis of Algerian App Store Reviews

Our approach aims at performing sentiment analysis on Algerian app store reviews using both lexicon-based and machine leaning based methods. As already mentioned, we consider the different existing languages, namely French, MSA Arabic and AlgD in both Arabic and Latin transcriptions. We follow the usual process of sentiment analysis (illustrated in Fig. 1) starting by *Data collection* that is done mostly from Google Play store and *Language detection* where we propose a novel process aiming at better identifying AlgD.
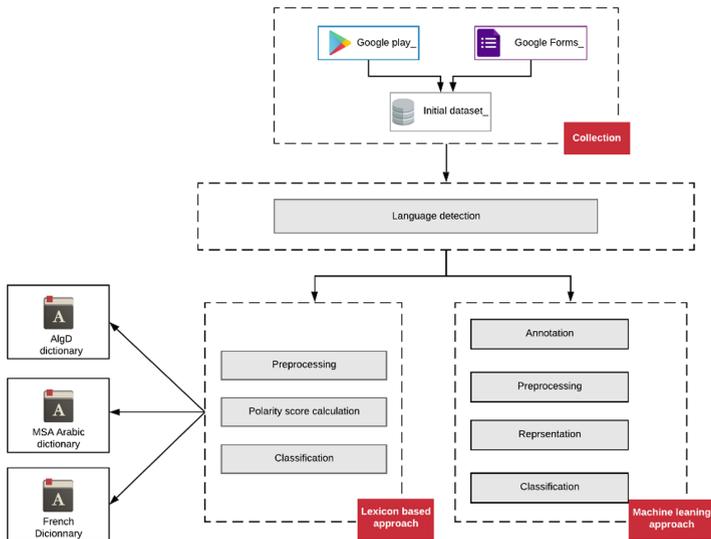

Fig. 1. Overview of the proposed approach

For the machine learning approach, the first step is *Data annotation* in which polarity labels are associated to all reviews in the dataset, followed by *Data preprocessing*

as pipelines of various steps specific to each language, *Data representation* allowing to convert text to numeric vectors, where we used TF-IDF being the most popular representation and finally *Data classification* in which the classifier is defined based on machine learning models to detect reviews' class {positive, negative}.

The lexicon based approach, for its part, requires opinion word vocabularies to be constructed first (one opinion dictionary per language). It start by a reduced (compared to machine learning) *Data preprocessing* step followed by *Data classification* by score computation according to opinion words present in reviews. We detail in what follows each step of the process for both machine leaning and lexicon based approaches starting with language detection which is common to both.

3.1 Language Detection

Previous studies like [8] used a process of elimination to identify AlgD reviews (if the text is Latin and not French then it is AlgD, and the same for Arabic). For our dataset, this process is observed to generate many misclassifications; many Arabizi AlgD were classified as French because they are interspersed with French words (which is very common among Algerian people writing). Thus, we propose to identify AlgD reviews using some of its characteristic traits. We first separate Latin (French and Arabizi ALgD) from Arabic and then apply the following processes.

Arabizi AlgD. We identified four features to distinguish Arabizi/French languages:

Presence of Digits. One distinctive feature of Arabizi text is the use of numbers and/or punctuation to symbolize Arabic letters that have no phonetic equivalent in Latin but resemble in shape. Digits 3, 5, 7 and 9 are used for example to represent sound of the letters "ع", "خ", "ح" and "ق" respectively. Moreover, when followed by «'», digits such "3", "7" and "9" change their interpretations and become "غ", "خ" and "ض".

Presence of "kh" or "gh". Another feature corresponds to the use of combinations "kh" and "gh" that map to Arabic letters "خ" and "غ". Many people use them instead of digits. Moreover, these combinations are rarely used in French and does not create any ambiguities.

Verb's Negation. In AlgD, the negation is usually attached to the word and expressed using the prefix "ma", the verb and suffix "ch". For instance, "**ma**3ajbetni**ch**" which means "I do not like it" or "**ma**temchi**ch**" meaning "it does not work". We use this later to identify Arabizi AlgD.

Succession of Letters. We also observe the frequent use of successive vowels and consonants in written AlgD, which is not allowed in French language (e.g. "mat7awsch tefhem" meaning "don't try to understand"). Thus, if a word contains more than three successive consonants (or vowels), it will be considered as Arabizi word.

As the Algerian dialect is characterized by the use of numbers, we filter out some words which can create ambiguity/interference (e.g. 3G, 4G, 20Go…) and construct an "interfering terms" dictionary. We also collected a list of AlgD words frequently found in reviews (such as "kifach" or "3lach" meaning respectively "how" and "why") and
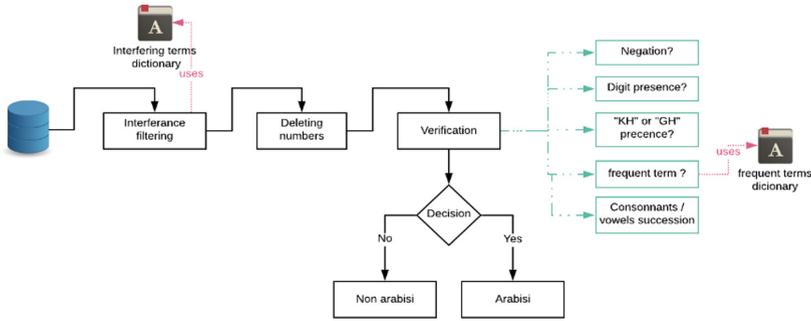
Fig. 2. Arabizi AlgD identification process

use them as an Arabizi AlgD "frequent terms" dictionary. The process of Arabizi AlgD identification is presented in Fig. 2.

Arabic AlgD. Different from Arabizi, Arabic AlgD reviews are very standardized and do not comprise any additions (such as numbers, punctuation). We note the absence of all the characteristic features mentioned above; which makes them difficult to dissociate from MSA Arabic reviews.

Our approach is to construct an Arabic dialectal dictionary by browsing the Arabic reviews to extract the words specific to AlgD. Each review is then compared to the dictionary and classified in MSA Arabic or Arabic AlgD. The AlgD has no orthographic rules, and thus several spellings can be observed for the same word. To handle this, we used the Levenshtein distance [24] for comparison. This later gives a measure of the similarity between two words and is equal to the minimum number of characters to be deleted, inserted or replaced to pass from one string to another. The words "علاش", "وعلاش", "علاه" (meaning "why") for instance are recognized as the same. The constructed dictionary includes *personal pronouns* such as "نتا", "نتوما", "هوما" different from MSA Arabic ones "أنت", "أنتم", "هم", respectively "you", "you" in plural and "they", *adjective and interrogative pronouns* which are very discriminative (e.g. "واش", "كيفاش", "علاش" meaning "why", "how" and "what"), *opinion words* in Arabic AlgD and some *frequent words* in app store reviews such as "ابليكاسيون" for "application". Our final vocabulary includes around 1000 distinct terms.

3.2 Machine Learning Approach

Data Annotation. The annotation step consists on associating a polarity label (positive, negative) to each review, which represents the expressed sentiment. We opted for manual annotation, which is more reliable especially for AlgD.

Data Preprocessing. This phase consists of applying multiple techniques to take out unnecessary information from reviews and keep maximum of variant informative vocabulary. Since we consider different languages, preprocessing will vary according to each.

It is worth noting that Arabic AlgD reviews are transliterated and combined with Arabizi ones to make one dialectal corpus. Transliteration allows to transform words from one script to another by swapping letters and without representing sounds. Systems such as Din 31635, Buckwalter or Qalam[2], are elaborated in Arabic-Latin-Arabic transliteration. Like previous studies [8], we use the Qalam system which is close to AlgD. We have also added some rules, notably to handle Arabic letters that have no Latin equivalent. Thus "ﺍ" is transliterated as 2 ("ع" as 3, "خ" as 5, "ح" as 7 and "ق" as 9) in order to have results most similar to Arabizi AlgD reviews. Our dataset includes, henceforth, reviews on three languages, French, MSA Arabic and Arabizi AlgD. We will detail in what follows the preprocessing steps.

Basic Preprocessing. Are applied to reviews of all languages, it includes removing of URLs, Hashtags, numbers and punctuation since they do not provide any contribution to the given sentiment. Emoticons are also removed because their usage is rare in the collected reviews. Moreover, shortening of elongated words is performed. These later are commonly present in users' reviews and can play the role of a feeling intensifier. For instance, "Foooooort" is positive word stronger than "Fort". Although, the focus of this study is positive/negative classification and the same meaning is provided in either way. Thus, we shortened words having more than two-character repetitions into a single one. For Latin script reviews (French and Arabizi) two additional steps are performed in order to keep single and unified instance of each word: *Dis-accentuation* which consists on removing accentuation from some letters like (é, è, à, ù) and substitute them with (e, a, u); and *Capitalized words processing* where all words are transformed to lowercase form.

Stop Words Removing. One of the important tasks of preprocessing is to filter out useless information referred as stop words that are commonly used and highly frequent in texts such as prepositions "the", "a", "an", "in"…etc. For regular languages viz. French and MSA, there are lists of well-known stop words available in several tools such as NLTK[3] Python platform. However, for AlgG no predefined resources are available hence the obligation to create them.

Stemming. Stemming (or rooting) is one of the core steps in the process. It consists on reducing inflectional (sometimes derivationally related) forms of a word to a common base form. Plenty of stemmers exist for standardized language such as Snowball[4] used to root French or English or ArabicStemmer[5] used for stemming MSA Arabic text. However, there is no stemmer elaborated for AlgD. In fact, dialect is not regular, nor standardized and lacks syntactic and grammatical rules (affixes attaching notably) allowing rooting. Thus, as done in related work [8, 23] we perform additional processing to handle the several spellings that can be observed for the same word, e.g. "mahabtch", "mahbtch", "mahebtch", "mhbtch", "mahebetch", "ma7abtch", "ma7btch", "ma7ebetch" meaning "she does not want" in english. The following steps are thus specific to AlgD. We recall that Arabic AlgD was already transliterated and all AlgD reviews are now in Arabizi.

---

2 http://qalam.info/.

3 https://www.nltk.org/.

4 https://snowballstem.org/.

5 https://www.arabicstemmer.com/.

Phonetic Grouping. Which consists of two parts. The first is *Digit substitution* with corresponding letters to unity words containing numbers. After this step, "ma**7**ebetch" for instance is converted to "ma**h**ebetch". The second step is *Phonetic grouping* performed using the Soundex algorithm [25]. It aims to group words that are pronounced in the same way based on their phonetic code. For instance, "mahabtch", "mahbtch", "mahebtch", "mhbtch", "mahebetch", referring all to the same word, having all the same meaning, the same pronunciation but different spellings. At the end of this step, a phonetic dictionary where each word is associated to a list of its corresponding phonetic codes, is created and then used to substitute certain words by their most common form. Those of the previous example will be associated to "mahabtch".

Vowels Removing. This step was proposed in our previous study [8] and proved to produce accurate results on social media comments. In fact, in written AlgD texts, the usage of vowels is often confused. For instance, users do not differentiate between the 'a' and the 'e' and write indifferently "mahabtch", "mahebtch" or "mahbtech". Thus, in our process we delete all vowels from a given word written in Arabizi; "mahabtch" and are converted to "mhbtch".

In addition, analyzing AlgD app store reviews, we notice that they were interspersed with French words. Accordingly, we propose to process them by applying the same pipeline as French reviews. This step allows also to reduce the size of the corpus vocabulary. Its contribution will be assessed in experiments.

Data Representation. This phase refers to the conversion of textual data into numeric vectors understandable by classifiers. To do so, we use the well-known TF-IDF (Term Frequency-Inverse Document Frequency) being the most popular and giving constantly good results [8]. The effectiveness of identifying features plays an essential role in obtaining high classification performance. TF-IDF has tackled the issue of most frequent words in the corpus assigning them less weight and thus let keeping only most relevant features.

Classification. Once the reviews are preprocessed and properly represented, the last phase is classification where our prediction model is created based on most robust and accurate classifiers in the NLP[6] field, which are: Naive Bayes, Support Vector Machine, k-Nearest Neighbors, Artificial Neural Networks and Decisional Trees. The model should be parametrized (iteration number, Depth of DT…) before its predictions evaluated.

3.3   Lexicon-Based Approach

The lexicon-based approach relies on pre-constructed opinion vocabulary and infer expressed sentiment via a semantic analysis of words. That is, the polarity of a review is determined according to the ones of the words that compose it. Each positive word get assigned $(+1)$ score and each negative (-1). After calculating the score of each review, we classify it as positive (if its score is bigger than 0), and negative otherwise.

---

[6] Natural Language Processing.

The vocabulary is manually constructed from opinion words present in our corpus. We consider different languages, opinion words of each are grouped into two distinct dictionaries, one for those referring to positive sentiment and another for negative opinion words. The following table presents an extract from the created vocabulary:

To compare review word to vocabulary, we use the Levenshtein distance that compares two strings by calculating a cost of transition from one to another (see Sect. 3.2), consequently we apply only basic preprocessing steps, in addition to stemming for standardized languages (French and MSA). Moreover, reviews in Arabic AlgD are not transliterated but directly included with MSA Arabic ones. The Arabic vocabulary contains opinion words of both MSA and Arabic AlgD (e.g. "شابة", "تعيي" in Table 1).

Table 1. Extract of constructed opinion vocabulary

Language	French		Arabic		Arabizi AlgD	
# terms	4 654		3 096		1 590	
	Positive	Negative	Positive	Negative	Positive	Negative
Extract	Bien	Nul	الأمثل	الفقيرة	Raw3a	Maderna
	Réussi	Bug	عزم	اتهم	Top	Ndemt
	Magnifique	Problème	مزايا	نزاع	Hayla	Compliqué
	Facile	Abuser	تفتح	يفقد	Khelwia	Nsit
	Excellent	Difficile	المثالية	حابسة	3ajbetna	Pubs
	Confiance	Court	مليح	الظلم	Intik	5edma
	Riche	Lente	شابة	تعيي	Bien	Wa3ra

Negation Treatment. Plays an essential role in lexicon-based classification as negation can change the polarity of any sentiment word to the opposite. Consequently, we propose a process that changes the polarity of opinion words. Wherever a negation is detected close to an opinion word, its polarity score will be reversed (multiplied by -1). To do so, negation words for each language (e.g. "ne", "jamais", "pas", "machi", "لا", etc.) have been collected and stored in a negation list.

4   Experiments

In this section, we evaluate the performance of our proposal. Firstly, we predict the polarity of reviews based on ML approach, hence we assess the Lexicon-based one. Experiments are conducted on new collected dataset with more than 50 000 reviews; using HP Omen machine with i7, 9th generation processor and 16 Go RAM. Python was used to perform machine learning algorithms. The model is evaluated by two appropriate and widely used metrics: accuracy and the F-measure. In what follows, we present the results obtained after each step of SA processing.

4.1 Data Collection

For our study, we collected a dataset of more than 50 000 reviews using two ways:

Data Scraper (Google Chrome Extension). We have collected most of the comments from the Google Play Store so as to obtain a domain related corpus. The data comprised reviews from a large number of popular Algerian apps such as mobile operators, VTC and delivery services…etc. we collected more than 48 000 reviews, mostly in French and MSA Arabic.

Google Forms. We also created a google Form in order to collect AlgD reviews with their polarities (positive or negative). Our aim was to offset the lack of dialectal reviews compared to French and MSA Arabic in the first dataset. Using the form, we collected around 2 032 reviews (1 046 negative and 986 positive reviews).

4.2 Machine Leaning Approach

Data Annotation. Before proceeding to annotation, we first cleaned up the raw reviews for duplication and non-textual (notably emoji) information. The dataset was reduced to about 34 838 reviews, 22 325 in French, 7 264 in Arabic and 5 429 in AlgD, as shown in Fig. 3 (orange bars).
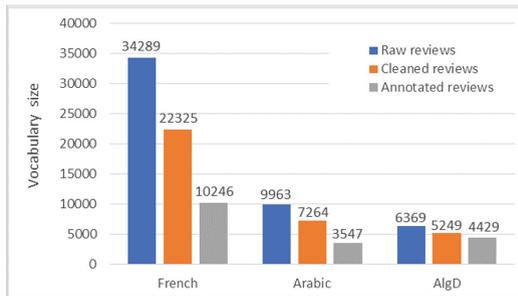
Fig. 3. Dataset size after data cleaning and annotation

To achieve annotation, we used an online platform: Tagtog[7], where we defined three labels: positive, negative and other (which contains irrelevant reviews such as personal information). We only kept positive and negative ones for analysis. Total number of such reviews attained 18 222 distributed as shown in Fig. 3. Note that reviews collected from google form was annotated by default as aforementioned. Results of the data annotation are given in Fig. 4.

---

[7] https://tagtog.net/.
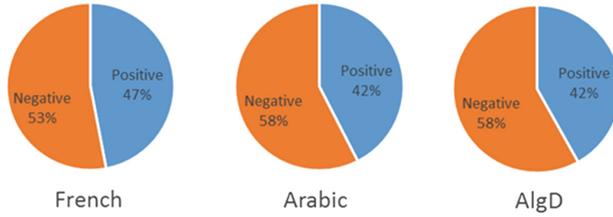
Fig. 4. Data annotation results

Data Preprocessing. As we deal with three languages, we conducted preprocessing for each of them. The Algerian dialect is the most complicated to deal with between the three. At the end, the process generates a corpus of 16 125 unique words, the before-after vocabulary size is illustrated in the following barplot (Fig. 5) where we can see that dimensionality is reduced to almost half for the three languages. This reduction allows keeping features most relevant to our analysis.
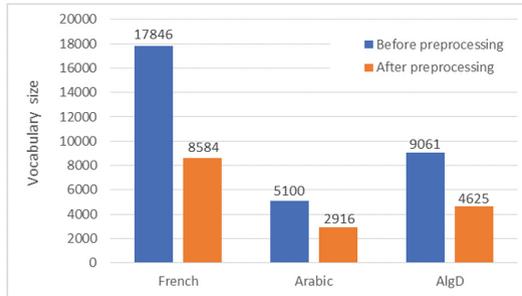
Fig. 5. Vocabulary size before/after preprocessing

As already pointed out, we noticed that AlgD reviews were interspersed with French words and accordingly proposed to perform an additional stemming step to handle them. Figure 6 (left) shows the repartition of words in Arabizi reviews; we can see that French words represent a significant proportion of 35%. The right side of Fig. 6, for its part, presents vocabulary size before and after applying French stemming. We observe a non-negligible decrease of 38% (from 7 387 to 4 625 features) in dimensionality, mainly due to grouping words representing the same instance.

Results and Discussion. To evaluate the proposed model, the dataset is split into two parts: 80% is inducted to train the classifiers and 20% kept for testing purposes. We trained different ML algorithms, namely Support Vector Machine (SVM), Naive Bayes (NB), k-Nearest Neighbors (KNN), Artificial Neural Networks (ANN) and Decisional Trees (DT). We also explored several settings to properly parametrize the model (iteration number, Depth of DT…) before its predictions evaluated. The following table shows the results of the different classifiers for the three languages (Table 2):
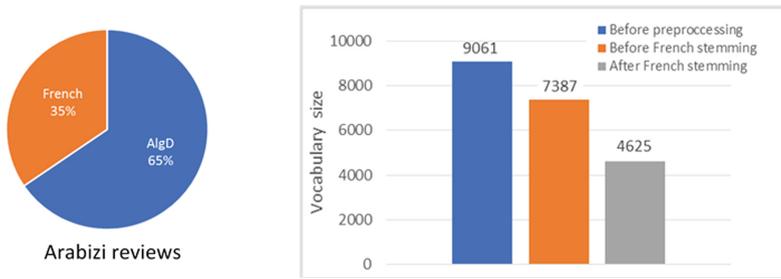
Fig. 6. Words repartition in Arabizi reviews (left) and results of stemming French words (right)

Table 2. Classification model evaluation

	French		Arabic		AlgD	
	Accuracy	F-measure	Accuracy	F-measure	Accuracy	F-measure
SVM	0,74	0,76	0,75	0,72	0,72	0,66
NB	0,75	0,73	0,62	0,76	0,67	0,72
KNN	0,56	0,72	0,61	0,77	0,54	0,7
ANN	0,66	0,71	0,7	0,75	0,59	0,66
DT	0,69	0,74	0,62	0,74	0,62	0,69

Among the applied ML algorithms, SVM has almost the highest performance, followed by Naïve bayes, with more significant advance over other algorithms on AlgD. This is not surprising that these two classifiers are the most used in sentiment analysis studies. In terms of accuracy, SVM largely outperformed NB for MSA and AlgD languages with respectively 75 and 72% against 62 and 67% correctly classified reviews. Regarding F-measure, NB achieved better results of 76 and 72% for MSA and AlgD languages. ANN and DT achieved satisfactory results in terms of F-measure (contrary to accuracy) especially for French and MSA Arabic. KNN produced the worst results in terms of accuracy regardless language; we observe, for instance an accuracy of 56% with a loss of about 25% compared to SVM for French.

To evaluate the impact of French processing in AlgD reviews, we performed analysis with several learning algorithms (SVM, NB and DT) in terms of accuracy in two cases, with and without the stemming step. The results are illustrated in Fig. 7:
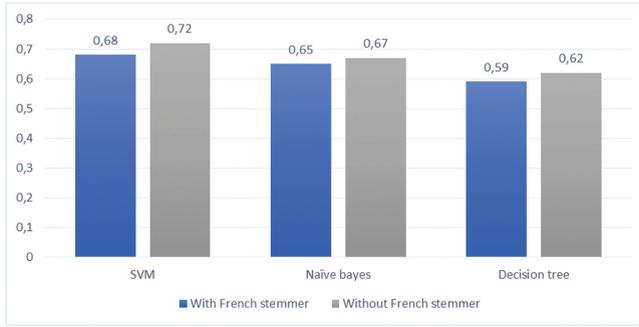
Fig. 7. Accuracy results before/after French word stemming

As can be seen in the above figure, the added step of stemming French words in AlgD reviews enhanced performance results and allowed us to lift the accuracy of SVM to 72% (with 4% improvement over results without stemming), and accuracy of NB and DT to 67% and 62% (against 65 and 59%) respectively; which is considerable and demonstrates the relevance of our proposition.

4.3   Lexicon-Based Approach

As already mentioned, to enable lexicon-based approach, we entirely constituted the opinion vocabulary of the Algerian dialect, given that there are no publicly available resources. French and Arabic vocabularies on the other hand are available, we had just to browse online dictionaries. The following figure illustrates the number of opinion words counted in the three languages (Fig. 8):
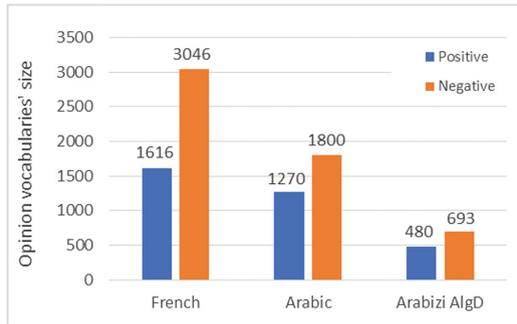
Fig. 8. Words repartition in opinion vocabulary

For evaluation, we took the reviews we had annotated before for supervised learning approach and ran the Lexicon-based categorization; which achieved encouraging results and even outperformed machine learning approach performance by correctly classifying 83% of French, 77% of MSA Arabic and 80% of AlgD reviews. Lexicon-based approach results in terms of accuracy are summarized in the following table (Table 3):

Table 3. Lexicon-based approach results

Language	French	Arabic	Arabizi AlgD
Accuracy	0.83	0.77	0.80

5   Conclusion

In this paper, Algerian Google play store reviews were subject to study by SA techniques; following two approaches: model based on general machine learning algorithms and lexicon-based method. The two approaches showed promising results on a dataset of more than 18 200 reviews which was also created as a part of this study. Different languages were handled properly and the proposed solution achieved interesting results even for AlgD having many specific aspects as a non-standardized language. On this later, we achieved an accuracy of 80% using the lexicon-based approach and of 72% for SVM algorithm where the proposed stemming step for French words in AlgD reviews allowed to improve results by about 7%.

Overall, this study enabled us to set a cornerstone for sentiment analysis on the Algerian mobile app store context and opens up to investigate many other areas such as subjectivity analysis or Irony/Sarcasm detection.

References

1. Genc-Nayebi, N., Abran, A.: A systematic literature review: opinion mining studies from mobile app store user reviews. J. Syst. Softw. (JSS) **125**, 207–219 (2017)
2. Pagano, D., Maalej, W.: User feedback in the appstore: an empirical study. In: 2013 21st IEEE International Requirements Engineering Conference (RE), pp. 125–134 (2013)
3. Panichella, S., Sorbo, A.D., Guzman, E., Visaggio, C.A., Canfora, G., Gall, H.C.: How can I improve my app? Classifying user reviews for software maintenance and evolution. In: 2015 IEEE International Conference on Software Maintenance and Evolution (ICSME), pp. 281–290 (2015)
4. Guzman, E., Maalej, W.: How do users like this feature? A fine grained sentiment analysis of app reviews. In: IEEE 22nd International Requirements Engineering Conference (RE), pp. 153–162 (2014)
5. Chen, N., Lin, J., Hoi, S.C., Xiao, X., Zhang, B.: AR-miner: mining informative reviews for developers from mobile app marketplace. In: ICSE 2014 (2014)
6. Fu, B., Lin, J., Li, L., Faloutsos, C., Hong, J.I., Sadeh, N.M.: Why people hate your app: making sense of user feedback in a mobile app store. In: KDD 2013 (2013)
7. Martens, D., Johann, T.: On the emotion of users in app reviews. In: 2017 IEEE/ACM 2nd International Workshop on Emotion Awareness in Software Engineering (SEmotion), pp. 8–14 (2017)
8. Chader, A., Lanasri, D., Hamdad, L., Belkheir, M.C., Hennoune, W.: Sentiment analysis for Arabizi: application to Algerian dialect. In: IC3K, pp. 475–482 (2019)
9. Saudy, R.E., Nasr, E.S., El-Ghazaly, A.E., Gheith, M.H.: Use of Arabic sentiment analysis for mobile applications' requirements evolution: trends and challenges. In: AISI (2017)

10. Hassan, S., Tantithamthavorn, C., Bezemer, C.P., Hassan, A.E.: Studying the dialogue between users and developers of free apps in the Google play store. Empir. Softw. Eng. **23**, 1–38 (2017)
11. Hoon, L., Vasa, R., Schneider, J.G., Mouzakis, K.: A preliminary analysis of vocabulary in mobile app user reviews. In: Proceedings of the 24th Australian Computer-Human Interaction Conference, pp 245–248. ACM (2012)
12. Wang, S., Wang, Z., Xu, X., Sheng, Q.Z.: App update patterns: how developers act on user reviews in mobile app stores. In International Conference on Service-Oriented Computing, pp. 125–141. Springer, Berlin (2017)
13. McIlroy, S., Ali, N., Khalid, H., E. Hassan, A.: Analyzing and automatically labelling the types of user issues that are raised in mobile app reviews. Empir. Softw. Eng. **21**(3), 1067–1106 (2016)
14. Li, X., Zhang, Z., Stefanidis, K.: Sentiment-aware analysis of mobile apps user reviews regarding particular updates. In: Proceedings of the Thirteenth International Conference on Software Engineering Advances (ICSEA), p. 109 (2018)
15. Soumik, M.M., Farhavi, S.S., Eva, F., Sinha, T., Alam, M.R.: Employing machine learning techniques on sentiment analysis of Google play store Bangla reviews. In: 22nd International Conference on Computer and Information Technology (ICCIT), pp. 1–5 (2019)
16. Wang, G., Sun, J., Ma, J., Xu, K., Gu, J.: Sentiment classification: the contribution of ensemble learning. Decis. Support Syst. **57**, 77–93 (2014)
17. Hadi, W.: Classification of Arabic social media data. Adv. Comput. Sci. Technol. **8**(1), 29–34 (2015)
18. Zarra, T., Chiheb, R., Moumen, R., Faizi, R., Afia, A.E.: Topic and sentiment model applied to the colloquial arabic: a case study of Maghrebi Arabic. In: Proceedings of the 2017 International Conference on Smart Digital Environment, pp. 174–181. ACM (2017)
19. Ali, C.B., Mulki, H., Haddad, H.: Impact du pretraitement linguistique sur l'analyse des sentiments du dialecte tunisien. In: Actes de la Conference Traitement Automatique de la Langue Naturelle, TALN 2018, p. 383 (2018)
20. Taboada, M., Brooke, J., Tofiloski, M., Voll, K., Stede, M.: Lexicon-based methods for sentiment analysis. Comput. Linguist. **37**(2), 267–307 (2011)
21. Mataoui, M., Zelmati, O., Boumechache, M.: A proposed lexicon-based sentiment analysis approach for the vernacular Algerian Arabic. Res. Comput. Sci. **110**, 55–70 (2016)
22. Mustafa, H.H., Mohamed, A., Elzanfaly, D.S.: An enhanced approach for Arabic sentiment analysis. Int. J. Artif. Intell. Appl. (IJAIA) **8**(5), 1–14 (2017)
23. Bettiche, M., Mouffok, M.Z., Zakaria, C.: Opinion mining in social networks for Algerian dialect. In: International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, pp. 629–641 (2018)
24. Levenshtein, V.I.: Binary codes capable of correcting deletions, insertions, and reversals. Sov. Phys. Dokl. **10**(8), 707–710 (1966)
25. Holmes, D., McCabe, M.C.: Improving precision and recall for soundex retrieval. In: International Conference on Information Technology: Coding and Computing, Proceedings, pp. 22–26. IEEE (2002)

Meta-learning to Select the Best Metaheuristic for the MaxSAT Problem

Souhila Sadeg[1]([✉]) [iD], Leila Hamdad[2], Omar Kada[1], Karima Benatchba[1],
and Zineb Habbas[3]

[1] Ecole nationale Supérieure d'Informatique, LMCS, Oued Smar, Algiers, Algeria
{s_sadeg,bo_kada,k_benatchba}@esi.dz
[2] Ecole nationale Supérieure d'Informatique, LCSI, Oued Smar, Algiers, Algeria
l_hamdad@esi.dz
[3] Université de Lorraine - LORIA, Metz, France
zineb.habbas@univ-lorraine.fr

Abstract. Several metaheuristics can be considered for solving a given optimization problem. Unfortunately none of them is better on all instances. Selecting *a priori* the best metaheuristic for a given instance is a difficult task which can be addressed using meta-learning. In this work, we propose a method to recommend, for a MaxSAT instance, the best metaheuristic among three: Genetic Algorithm (GA), Bee Swarm Optimization (BSO) and Greedy Randomized Adaptive Search Procedure (GRASP). Basically, a learning model is trained to induce associations between MaxSAT instances' characteristics and metaheuristics' performances. The built model is able to select the best metaheuristic for a new MaxSAT instance. We experiment different learning algorithms on different instances from several benchmarks. Experimental results show that the best metaheuristic is selected with a prediction rate exceeding 80% regardless the learning algorithm. They also prove the effect of instances used in training on the model performance.

Keywords: Meta-learning · Algorithm selection · Metaheuristics · MaxSAT

1 Introduction

During the last decade, there has been a growing interest in the use of algorithms from data mining and machine learning fields to enhance metaheuristics' performances has been observed. Interesting papers propose taxonomies of works in this emerging field by classifying them following different criteria [3,4,17]. The most recent categorization we found in the literature is the one proposed in [3] in which works are classified into specifically-located hybridizations (where machine learning is applied in a specific procedure) and global hybridizations (in which machine learning has a higher effect on the metaheuristic design).

The first group comprises parameter fine-tuning, initialization, evaluation, population management, operators, and local search. On the other hand, reduction of search space, hyper-heuristics, cooperative strategies, and algorithm selection are considered as parts of the second category.

We are interested in this work by the algorithm selection problem (ASP) which was first defined by John Rice in 1976 [14] and aims to predict, for a given instance of a problem, the most suitable algorithm among a set. Indeed, a large number of algorithms are continuously proposed to solve numerous problems in many application fields such as data mining and optimization but the task of choosing the best one is challenging. Actually, It is admitted that there is no algorithm better for all problems or even for all instances of the same problem [18]. In fact, algorithms show different performances from one problem to another and from one instance to an other as explained by the "*No Free Lunch Theorem*" which states that no algorithm is better than others and across all performance measures. To select the most appropriate algorithm we need to take into consideration the properties of the problem or instance and determine the suitable features that can be easily calculated and are likely to correlate with algorithm performance as aknowledged by Rice[14].

ASP is generally addressed as a learning problem. Its goal is to learn a model that makes a mapping between the characteristics of a set of instances and the performances of a set of algorithms so that it can choose the most suitable one for a new given instance. It was first applied by the machine learning community to solve classification problems and used the term meta-learning: learning about learning.

Meta-learning has been successfully applied to the selection of algorithms for various types of problems [2]. Indeed, although the research on meta-learning has been mainly focused on the selection of machine learning algorithms, studies applying meta-learning in other areas such as optimization are continuously proposed for various well-known optimization problems such as Traveling Salesman Problem (TSP) [10], Quadratic Assignment Problem (QAP) [16] and Graph Coloring Problem (GCP) [11].

In our work, we apply algorithm selection using machine learning to select the best metaheuristic for solving an instance of the Maximum satisfiability problem (MaxSAT). It is an optimization variant of the well-known satisfiability (SAT) that is extensively studied because many complex theoretical and practical problems can be expressed as MaxSAT. Max-cut, max-clique, model checking, routing, bioinformatics and scheduling are some examples. Given a set of n Boolean variables $X = \{x_1, x_2, ..., x_n\}$ and a set of m clauses $C = \{c_1, c_2, ..., c_m\}$, where each clause is a disjunction of literals, and each literal is a variable in X or its negation, the goal of MaxSAT is to find an assignment of true variables that maximises the number of satisfied clauses. We propose to apply meta-learning approach to select among a portfolio of metaheuristics the most appropriate for a MaxSAT instance. We study the impact of instances used at the meta-learning level by selecting only the most discriminating ones.

The remainder of this article is organized as follows: Sect. 2 introduces the algorithm selection problem as formalized by Rice [14]. Meta-features for MaxSAT problem are described in Sect. 3. Section 4 presents the experiments

methodology and the portfolio of metaheuristics used in our work. Numerical results are presented in Sect. 5 before we conclude.

2   Algorithm Selection Problem

Algorithm selection as defined by [14] aims to predict which algorithm from a portfolio is the most suitable for a given instance. This prediction is based on characteristics of a set of instances and the performances of each algorithm on each instance. As illustrated in Fig. 1, the model is composed of four important components:

– P: a set of instances of the problem
– F: a set of features calculated for each instance x in P as a vector $f(x)$
– A: a set of candidate algorithms
– Y: the performance space

The collection of data describing P, A, Y, F is known as the meta-data and the objective is to find for a given problem instance $x \in P$ with a feature vector $f(x)$ a selection mapping $S(f(x))$ into algorithm space A such that the selected algorithm $\alpha \in A$ maximizes the performance $y(\alpha, x) \in Y$ [15].
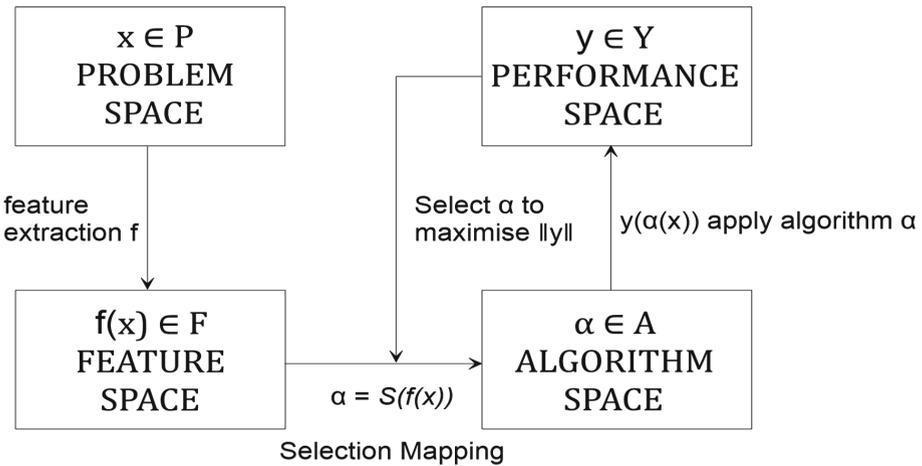
Fig. 1. Schematic diagram of Rices's algorithm selection problem model [14]

Finding the appropriate features that allow the mapping function to predict the best algorithm is a challenging task for which no automatic method exists. It depends strongly on the problem domain and requires deep domain knowledge and analytical skills [11].

Algorithm selection is generally addressed using a meta-learning approach which is defined by Brazdil et al. in [1] as the study of methods that exploit meta-knowledge to obtain efficient models and solutions by adapting machine learning

and data extraction processes. Meta-knowledge is composed of algorithms' performances of every algorithm on each instance and meta-features which describe the instances. Usually, selection procedure is either based on classification or regression techniques [11]. Classification techniques classify a new instance into one class among a set of classes labeled each one by the algorithm that performs better on the instances it contains. On the other hand, regression techniques predict the algorithms' performances on a new instance following a model built in the learning step.

3   Meta-features for MaxSAT

A crucial step in algorithm selection is the identification of meta-features, the set of features that characterize the instance set. An obvious requirement for features in this context is to ensure that they can be computed faster than running all of the algorithms in the portfolio [15]. Concerning the satisfiability problem, large amounts of features have already been proposed in literature [12,13]. We opted in this work for the set of features presented in Table 1 proposed in [12] because they can be very quickly computed and may serve well for algorithm selection.

Table 1. Set features for MaxSAT

Problem Size Features
1–3. Number of clauses c, Number of variables v, Ratio v/c
Variable-Clause Graph Features
4–8. Variable nodes degree statistics: mean, variation coefficient, min, max, and entropy
9–13. Clause nodes degree statistics: mean, variation coefficient, min, max, and entropy
Balance Features
14–16. Ratio of positive and negative literals in each clause: mean, variation coefficient, and entropy
17–21. Ratio of positive and negative occurrences each variable: mean, variation coefficient, min, max and entropy
22–23. Fraction of binary and ternary clauses
Proximity to Horn Formula
24. Fraction of Horn clauses
25–29. Number of occurrences in a Horn clause for each variable: mean, variation coefficient, min, max, and entropy

4   Experimental Framework

This section describes our experimental framework. Mainly, we present the set of metaheuristics forming the algorithms' portfolio, the set of MaxSAT instances used in our tests and also the meta-learning process.

4.1   Set of Metaheuristics

Three metaheuristics were used in our experiments: Genetic Algorithm (GA), Greedy Randomized Adaptive Search Procedure (GRASP) and Bee Swarm Optimization (BSO). Each of them is briefly presented in this section.

Genetic Algorithm (GA). It is one of the most used evolutionary methods. Developed by Holland in 1975 [8], it is inspired by the natural selection and genetic evolution and generally composed of two main steps [7]: 1) generating the initial population either randomly or using a heuristic and 2) iterating a process of four operations (selection, crossover, mutation and update of population) until a stopping criterion is reached. The general GA algorithm is presented in Algorithm 1

Algorithm 1. GA general algorithm

1. generating initial population;
2. Evaluation of population
3. **While** (*nonstoppigcriterion*) **Do**
   (a) Selection
   (b) Crossover
   (c) Mutation
   (d) Evaluation of population
4. **return** the best population

Greedy Randomized Adaptive Search Procedure (GRASP). GRASP was first proposed in 1989 [6] and several variants were proposed in the literature. In our experiments we used the original one which consists of two steps: solution construction phase and the improvement phase where the solution obtained in the first phase is improved by performing a local search. The generic algorithm of GRASP is as follows.

Algorithm 2. GRASP general algorithm

While non stopping criterion

1. Construct a solution S randomly;
2. improve S by a local search;

EndWhile
return the best solution found;

Bee Swarm Optimization (BSO). BSO is an iterative search process based on a population of artificial bees cooperating to solve an instance of an optimization problem by imitating the foraging behavior of natural bee [5]. It is based on 3 key steps:

– Selection of a reference solution
– Determination of the search region;
– Bees' local searches.

In the first step, a reference solution is selected either randomly (in the first iteration) or among a set of good solutions (in the following iterations). After that, a set of candidate solutions are generated from the reference solution and each is assigned to a bee as the starting point of its local search. Each bee communicates its best found solution and a new reference solution is selected based on quality and diversification criteria. For a good balance between diversification and intensification, a maximum number of chances is granted to each search area and reference solutions are stored in a *Tabu* list to avoid cycles. The BSO general algorithm is presented in Algorithm 1.

Algorithm 3. BSO General algorithm
---
1. Generate the first reference solution *RefSol* randomly or via a heuristic
2. **While** non stopping criterion **Do**
    (a) Determine *SearchRegion* from *RefSol*
    (b) Assign a solution from *SearchRegion* to each bee
    (c) **for** each bee k
        i. Perform a local search
        ii. return the best found solution
    (d) **EndFor**
    (e) Choose the new *RefSol*
3. **EndWhile**
4. Return the best solution found

4.2   Set of Instances

The set of instances is composed of 1534 instances from five diverse types taken from the *SATLIB*[1] and two types of instances taken from the MaxSAT 2015 competition[2]:

---

[1] http://www.cs.ubc.ca/~hoos/SATLIB/benchm.html.
[2] http://www.maxsat.udl.cat/15/.

1. *Uniform Random 3-SAT.* Instances are generated randomly such that for an instance composed of n variables and k clauses, each clause consists of 3 literals drawn randomly and equiprobably among the $2n$ possible literals (n variables and their negations).
2. *Morphed graph Coloring problems.* This type of instance is the result of a transformation of graphs, resulting from a combination of ring graphs and random graphs, into SAT instances.
3. *flat graph coloring problem.* In this case, the learning instances are obtained after a 3-colorable flat graph transformation into SAT instances.
4. *Random instances with variable length clauses.* These instances are generated randomly such that for an instance composed of n variables and k clauses, each variable is included in a clause with a probability p and its negation is then included with a probability of 0.5. The set of used instances contains 50 instances, where 40 are composed of 100 variables and 800 clauses and 10 instances contain 100 variables and 900 clauses. Only 16 instances among the 50 are satisfiable while the others are not satisfiable.
5. *Artificially generated random instances.* They represent instances of the 3-SAT type, generated randomly to be different from the instances generated in a deterministic way and those obtained by translating other problems, such as the problem of coloring graphs. 72 instances of this type are used, where 48 are satisfiable and 24 are not.
6. *Instances of the MaxSAT 2015 competition.* Two types of instances are used, some are manually built while the others are randomly generated.

Table 2 presents the set of learning instances described above.

Table 2. Learning instances

Benchmark	Type of Instances	#instances
Uuf75-325	Uniform random 3-SAT instances	100
Uuf200-860		100
Sw100-8-lp0-c5	Morphed graph Coloring problems	100
Sw100-8-lp5-c5		100
Flat125-301	Flat 3-colouring graph problems	100
Flat200-479		100
AIM	Random Instances with variable Clauses	72
JHN	Artificially generated random instances	50
Ms_crafted	Manually constructed instances	62
Ms_random	Randomly generated instances	750

4.3   Meta-learning Process

Using meta-learning for algorithm selection involves three essential phases:

1. *Knowledge Acquisition*: during this step we select learning instances (which are representative of the problem space), extract their features - also called *meta-features* in the context of meta-learning) - and execute each algorithm on each instance. The features and algorithms performances will constitute what is called *metadata*.
2. *Building the meta-model*: the meta-model is built using a machine learning algorithm on the *metadata* obtained in the first step. More specifically, a supervised classification algorithm is used to categorize instances into a given number of classes equal to the number of algorithms.
3. *Validating the meta-model*: finally, to evaluate the meta-model, we test it on a set of new instances where the classifier predicts classes of new given instances. Predictions are evaluated by executing the algorithm on each instance.

Two questions related to the main aspects of the meta-learning process are studied in our work:

1. Are all the instances used in the learning phase sufficiently discriminant to allow a good learning?
2. How much the classifier can influence the performances of the model (recommendation algorithm)?

We try to answer these questions in the following section by analysing experimental results.

5   Tests and Results

In this section, we present the results obtained after performing several tests on the set of MaxSAT instances described in the previous section. Algorithms are implemented in java and tests were performed on Intel(R) Xeon(R) CPU E5-2650 with a memory of 2.00 GHz. Algorithms have been implemented in java and Weka[3] has been used to perform data mining tasks.

5.1   Metaheuristics' Parameters

Fixing the values of algorithms' parameters is a crucial task that has a direct impact its performances. Considering the three metaheuristics, we have 11 parameters to adjust following two performance measures: solution quality and execution time. Manual tuning is a hard task that would take several weeks. We therefore opted for an automatic adjustment using the *ParamILS*Framework proposed in [9].[4]

The optimal values returned by *ParamILS* for the three algorithms are given in Table 3.

---

[3] http://www.cs.waikato.ac.nz/ml/weka/.
[4] http://www.cs.ubc.ca/labs/beta/Projects/ParamILS/.

Table 3. Values of metaheuristics' parameters

Algorithm	parameters' values
GA	maxIter : 200,
	mutationRate : 0.005,
	sizePop: 500,
	tournamentSize : 40
GRASP	maxIter : 60,
	alpha : 0.75
BSO	chance : 7,
	flip : 10,
	maxIter : 25,
	nBees : 10,
	nbSearchIter : 15

5.2 Data Acquisition and Preparation

In order to test our recommendation algorithm, we have to perform the data acquisition step by:

1. Extracting the features of each instance in the learning set
2. performing for each metaheuristic ten(10) executions on each instance

After that, data is normalized and labeled in order to perform a supervised classification. The label of an instance is the most appropriate algorithm following the desired performance.

The performance of an algorithm can be seens as the quality of the returned solution, the execution time or a combination of quality and time. In order to offer the three possibilities we use the following formula

$$performance = \alpha * time + (1 - \alpha) * quality \qquad (1)$$

where α is the weight granted to execution time. For example, if only solution quality is considered as metaheuristic performance α is assigned the value 0, and each instance is labeled by the algorithm that has given the best solution quality.

Another important aspect of data preparation concerns the relevance of instances used in the learning step. Indeed, for some of them, metaheuristics give very close performances. Consequently, they are not discriminating enough to allow a good learning. Hence, we introduce a threshold parameter named t which belongs to the interval [0,1] and is used to decide if an instance is kept in the learning set or not. As an example, consider a problem instance for which the metaheuristics give the following performances: 0.33, 0.33 and 0.34. Without the use of parameter t, this instance will be labeled by the algorithm which obtained 0.34, whereas actually the three algorithms are equivalent. To overcome

this problem, the third algorithm is chosen as a label for this instance only if its performance (0.34 in our example) is greater than the value of t, otherwise this instance is removed from the learning instances' set. In other words, the value of t indicates how much better must be the algorithm to be considered more suitable than the others.

Learning. In the learning phase, a learning algorithm is used as a meta-learner but the choice of the most appropriate classifier is in itself an algorithm selection problem. Therefore, we use in our experiments three learning algorithms: random forests, neural networks and k nearest neighbors.

Learning step produces a classification model whose efficiency depends on threshold's value t and the learning algorithm (classifier). Their impact on the performance of our recommendation algorithm is studied in the remainder of this section.

Effect of the Parameter Threshold "t". In order to evaluate the effect of parameter t, we performed two series of tests with three values of $alpha$: 0.25, 0.5 and 0.75. In the first set of tests, $t = 0$ which means that no selection is performed on instances used in the learning step while in the second set, $t = 0.5$ which means that only instances for which an algorithm obtained a performance greater than or equal to 0.5 are considered. The obtained results illustrated in Fig. 2 show that the best classification rates are obtained when a selection of instances used in the learning step is performed. We can also observe that the effect of this selection is stronger when the performance advantages execution time criteria.
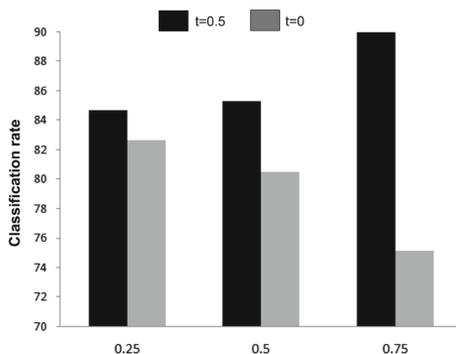
Fig. 2. Effect of the parameter t on the classification rate

Effect of the Learning Algorithm. As it is difficult to choose a priori the best learning algorithm to use in classification task, we performed tests on three

classifiers: Artificial neural networks (ANN), random forests and k-nearest neighbors (K-NN). Figure 3 shows a comparison of the results following the same values of α used previously $(0.25, 0.5, 0.75)$. We can see that the three classifiers give equivalent classification rates when $\alpha = 0.75$ (execution time is given more importance than quality). In the other cases random forests algorithm is outperformed by k-NN and neural networks.
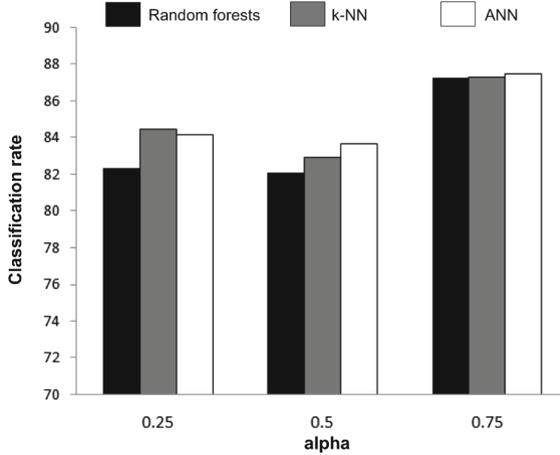
Fig. 3. Classification rate obtained by the learning algorithms following different values of $alpha$

5.3   Evaluation of the Recommendation Algorithm

In general, the classification rate is the most representative indicator of the performance of a learning algorithm. However, we were also interested by the global performances of the recommendation algorithm that we measured by the total number of satisfied clauses and the total execution time calculated using the following formulas.

$$Quality = \sum_{i=1}^{numbre\ of instances} number\ of\ satisfied\ clauses\ in\ instance_i \qquad (2)$$

$$time = \sum_{i=1}^{numbre\ of instances} execution\ time\ of\ instance_i \qquad (3)$$

The results of metaheuristics on all instances presented in Table 4, show that GA gives better execution times while BSO is more interesting in terms of quality.

Table 4. Results of the three metaheuristics

Algorithm	Quality	Time (ms)
GRASP	560719	727540
GA	510126	**150345**
BSO	**562133**	486259

The two Figs. 4 and 5 present respectively the obtained results in terms of quality and execution time following several values of *alpha*. They show a comparison between the metaheuristics, the recommendation algorithm and an optimal classifier that would give a classification rate of 100%, ie, an oracle that always selects the best metaheuristic whatever the value of alpha. In these tests $t = 0.5$ and ANN is used for learning.

We can easily see that results of our recommendation algorithm are almost identical to the optimal selection algorithm. Indeed they are so merged that it is difficult to distinguish them. This means that the recommendation algorithm selects the best metaheuristic for almost all instances. In addition, we can observe that:

- for $0 < alpha < 0.5$: the recommendation algorithm is more efficient in terms of quality than the three metaheuristics. Moreover, its execution time (484600 ms) is smaller than BSO (486200) which is the best one in terms of quality.
- for $0.5 < alpha < 0.8$: our recommendation algorithm gives a good compromise between quality and execution time.
- for $0.8 < alpha < 1$: The recommendation algorithm is better than the three metaheuristics in terms of execution time. Moreover, it shows better quality ($514000 satisfied clauses$) than GA which is the fastest one (GA allowed to satisfy only 510000 clauses).
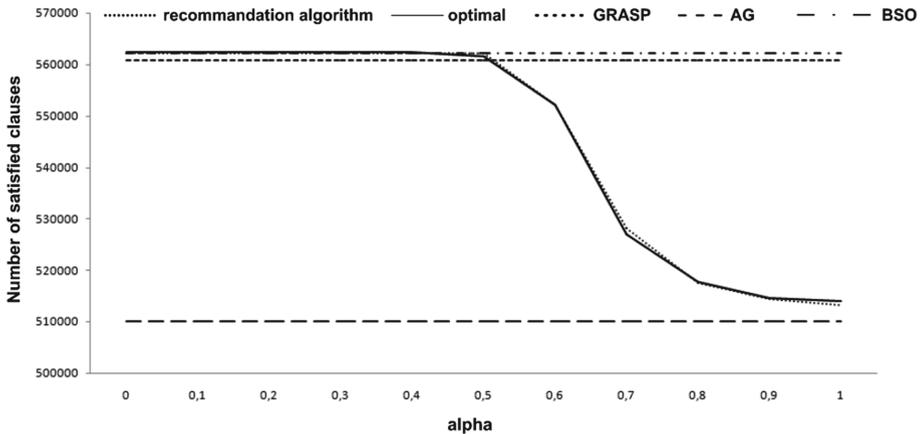
Fig. 4. Evaluation of the recommendation algorithm in terms of quality following the value of *alpha*
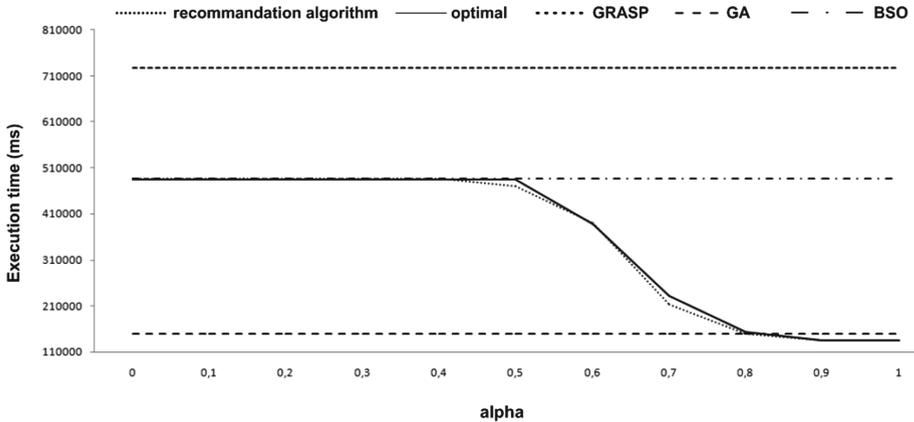
Fig. 5. Variation of execution time according to *alpha*

6   Conclusion

There are several algorithms that can be used successfully to solve an instance of a hard combinatorial optimization problem but it is very difficult to choose the best one a priori. Indeed it is admitted that there is no algorithm performing better than the others on all instances of a problem let alone on all problems. We addressed in our work this problem known as *algorithm selection* to propose a recommending algorithm that selects the best metaheuristic for a given MaxSAT instance. We used a meta-learning approach that considers the problem as a classification one: A learning model is built by training it on meta-data composed of instances defined by a set of features and metaheuristics for which we know their performances on each instance. During the training phase the model learns to relate metaheuristic performance to instance characteristics and becomes able to predict the most appropriate metaheuristic for a new instance. In order to improve the prediction rate we proposed to fix a threshold for algorithm performance that decides if an instance is kept or removed in the training set. Only instances for which a metaheuristic gives a performance higher than or equal to the threshold value are considered in the training step. Experiments performed on more than 1500 instances using three different learning algorithms showed that, regardless of the learning algorithm, the best metaheuristic is selected with a prediction rate exceeding 80% when the performance considered is solution quality and approaches 95% in case of execution time. The results confirmed the importance of choosing well the instances of the learning set since a significant increase in the prediction rate is obtained when the most discriminating ones are used. In our future works, we plan to enrich the portfolio with other metaheuristics and study the algorithms' performances on each benchmark instead of the whole instances set.

References

1. Brazdil, P., Carrier, C.G., Soares, C., Vilalta, R.: Metalearning: Applications to Data Mining. Springer, Heidelberg (2008)
2. Brazdil, P., Giraud-Carrier, C.: Metalearning and algorithm selection: progress, state of the art and introduction to the 2018 special issue (2017)
3. Calvet, L., de Armas, J., Masip, D., Juan, A.A.: Learnheuristics: hybridizing metaheuristics with machine learning for optimization with dynamic inputs. Open Math. **15**(1), 261–280 (2017)
4. Corne, D., Dhaenens, C., Jourdan, L.: Synergies between operations research and data mining: the emerging use of multi-objective approaches. Eur. J. Oper. Res. **221**(3), 469–479 (2012)
5. Drias, H., Sadeg, S., Yahi, S.: Cooperative bees swarm for solving the maximum weighted satisfiability problem. In: Computational Intelligence and Bioinspired Systems, pp. 318–325. Springer (2005)
6. Feo, T.A., Resende, M.G.: A probabilistic heuristic for a computationally difficult set covering problem. Oper. Res. Lett. **8**(2), 67–71 (1989)
7. Gen, M., Cheng, R.: Genetic Algorithms and Engineering Optimization, vol. 7. Wiley, New York (2000)
8. Holland, J.H.: Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence (1975)
9. Hutter, F., Hoos, H.H., Leyton-Brown, K., Stützle, T.: Paramils: an automatic algorithm configuration framework. J. Artif. Intell. Res. **36**(1), 267–306 (2009)
10. Kanda, J., de Carvalho, A., Hruschka, E., Soares, C., Brazdil, P.: Meta-learning to select the best meta-heuristic for the traveling salesman problem: a comparison of meta-features. Neurocomputing **205**, 393–406 (2016)
11. Musliu, N., Schwengerer, M.: Algorithm selection for the graph coloring problem. In: Learning and Intelligent Optimization, pp. 389–403. Springer (2013)
12. Nikolić, M., Marić, F., Janičić, P.: Simple algorithm portfolio for sat. Artif. Intell. Rev. **40**(4), 457–465 (2013)
13. Nudelman, E., Leyton-Brown, K., Hoos, H.H., Devkar, A., Shoham, Y.: Understanding random sat: Beyond the clauses-to-variables ratio. In: International Conference on Principles and Practice of Constraint Programming, pp. 438–452. Springer (2004)
14. Rice, J.R.: The algorithm selection problem. Adv. Comput. **15**, 65–118 (1976)
15. Smith-Miles, K., van Hemert, J.: Discovering the suitability of optimisation algorithms by learning from evolved instances. Ann. Math. Artif. Intell. **61**(2), 87–104 (2011)
16. Smith-Miles, K.A.: Towards insightful algorithm selection for optimisation using meta-learning concepts. In: IEEE International Joint Conference on Neural Networks, 2008. IJCNN 2008, (IEEE World Congress on Computational Intelligence), pp. 4118–4124. IEEE (2008)
17. Talbi, E.G.: Combining metaheuristics with mathematical programming, constraint programming and machine learning. Ann. Oper. Res. **240**(1), 171–215 (2016)
18. Wolpert, D., Macready, W.: No free lunch theorems for search (technical report sfi-tr-95-02-010). Santa F e Institute, Santa F e, NM (1995)

Ontological Relation Classification Using WordNet, Word Embeddings and Deep Neural Networks

Ahlem Chérifa Khadir[1,2(✉)], Ahmed Guessoum[1], and Hassina Aliane[2]

[1] NLP, Machine Learning and Applications (TALAA) Research Group, Laboratory for Research in Artificial Intelligence (LRIA), Department of Computer Science, University of Science and Technology Houari Boumediene (USTHB), Algiers, Algeria
{akhadir,aguessoum}@usthb.dz

[2] Research and Development in Digital Humanities Division, Research Centre on Scientific and Technical Information (CERIST), Algiers, Algeria
ahassina@cerist.dz

Abstract. Learning ontological relations is an important step on the way to automatically developing ontologies. This paper introduces a novel way to exploit WordNet [16], the combination of pre-trained word embeddings and deep neural networks for the task of ontological relation classification. The data from WordNet and the knowledge encapsulated in the pre-trained word vectors are combined into an enriched dataset. In this dataset a pair of terms that are linked in WordNet through some ontological relation are represented by their word embeddings. A Deep Neural Network uses this dataset to learn the classification of ontological relations based on the word embeddings. The implementation of this approach has yielded encouraging results, which should help the ontology learning research community develop tools for ontological relation extraction.

Keywords: Semantic relation classification · Word embeddings · Deep learning · Ontologies · WordNet

1 Introduction

Semantic relation classification is an important task in the area of natural language processing. The state-of-the-art solutions for this problem are often based on patterns and traditional machine learning techniques (Support Vector Machines, Bayesian Networks, Decision Trees, etc.). The performances of these approaches strongly depend on the quality of the used patterns (mainly based on Hearst lexico-syntactic patterns [10]) and the extracted features (Part-Of-Speech tags, frequencies, distance measures, etc.). The latter are often obtained from existing Natural Language Processing (NLP) systems, which leads to the

propagation of potential errors. On the other hand, there are recent works that use classification models based on deep neural networks [12] which are outperforming the state-of-the-art results. However, these models usually suffer from the lack of labeled datasets. Indeed, such datasets are tedious to produce and are, consequently, often small in terms of the number of samples. Also, they are rarely specifically built for the case of ontological relations.

In this paper, we are interested in classifying semantic relations, specifically ontological WordNet-like relations, relying on pre-trained word vectors and WordNet knowledge. These two sources of information will be used to build our dataset. The assumption is that word vectors that are pre-trained on large text corpora encapsulate the context information of a defined word, as reported by Collobert et al. in [5], and this information can be taken as a basis to represent this word as an entity which is involved in a relation. More precisely, the assumption is that the type of semantic relation that links two words in WordNet can be approximated using the word vectors that represent them.

The remainder of this paper is organized as follows: Sect. 2 presents the related work. The proposed approach is described in Sect. 3. Section 4 explains the tests that have been done and the results are discussed in Sect. 5. In Sect. 6, a conclusion is presented where the contribution of this work is highlighted and directions for future work are stated.

2   Related Work

In the context of the shared task 8 called "Multi-Way Classification of Semantic Relations Between Pairs of Nominals" of the SemEval-2010 competition [11], a dataset of 9 relations (Cause-Effect, Instrument-Agency, Product-Producer, Content-Container, Entity-Origin, Entity-Destination, Component-Whole, Member-Collection and Message-Topic.) was developed by crawling the web using patterns, and validated by human experts. The dataset includes 10,717 sentences containing annotations showing the nouns that are related by one of the 9 above relations. Although relatively small, the dataset allowed the comparison of several approaches that tackled the aforementioned task, in the context of the competition and beyond. At that time, the best F1-score (82.19%) was achieved by a team that used a Support Vector Machine model which considered several features (hypernyms, POS, distance, etc.). Recent works [13,18,22,24] used the same dataset and outperformed the F1-score previously obtained. The work in [24] showed the contribution of word embeddings and deep learning models in improving the results of the relation classification task and that this combination outperformed the approaches that used traditional machine learning models. The authors used a convolutional deep neural network (CNN) to extract lexical and sentence level features. The features were fed into a softmax layer to predict the relationship between two marked nouns. They obtained 82.7% F1-score on the aforementioned dataset.

The work in [18] yielded better results relying on pre-trained embeddings and a CNN deep learning architecture. They showed that when considering only

the text segment between related target nominals instead of the whole sentence provided by the dataset, they obtained the best results with an 84.8% F1-score.

In [22], the authors proposed to enrich the pre-trained embeddings with the information contained in the sentence the related target nominals appear in. To do so, they used BERT (Bidirectional Encoder Representations from Transformers), a recent embedding model developed by Google [6]. The sentence vector as well as the target entity representations that denote the pair of nominals were used to perform the classification task. Using special tokens, the target entity representations were obtained by specifying to the BERT module the location of the two entities in the sentence. The global architecture of their classifier consists of a deep neural network of fully connected layers. They achieved 89.25% on the SemEval-2010 task 8 dataset.

Similarly, the authors of [13] proposed a new pre-trained CNN network architecture called XM-CNN which uses a reinforcement mechanism [21] to capture the relevance of words with respect to the target entities and then an attention pooling mechanism [19] to determine the most useful convolved features for relation classification. This approach achieved the best F1-score, 91.6% on the SemEval-2010 task 8 dataset.

The SemEval-2018 shared task "Semantic relation extraction and classification in scientific papers" [8] addressed the classification of relations specifically in the scientific domain. Six categories of relations (usage, results, model, part-whole, topic and comparison) were considered. The dataset consisted of 350 annotated abstracts of scientific documents for training and 150 for testing. Three subtasks were defined: "Relation classification on clean data", "Relation classification on noisy data" and "Relation extraction and classification scenario". The best scores were obtained by systems that combined the use of word embeddings and deep learning models with an average F1-score of 80% and 70%, respectively, for the first two subtasks. However, for the third subtask the best F1-score was only about 50%.

With the aim of classifying antonymy and synonymy relations, the authors of [4] proposed a deep neural network model, called DVASNet, which relies on embedding representations of words, co-occurrence contexts and specific patterns of Vietnamese words structure. They performed their tests using a dataset of 400 antonym pairs and 600 synonym pairs and obtained a 78% F1-score.

In [20] the authors addressed the semantic relation classification shared task, presented in [23]. This task consists in classifying the relation that exists in the Chinese language between any pair of words regardless of the sentences they may appear in. Since the test set provided for the task consisted of only 2000 annotated pairs, the participants had to build their own training data using dictionaries. Four relations were considered: hyponymy, meronymy, synonymy and antonymy. The authors of [20] achieved an F-score of 76.8% using a CNN deep learning model enriched with an embedding layer fed with pre-trained word vectors of the involved nominals. This work is the closest one if compared to our approach. Nevertheless, the dataset we provide covers many more instances and

relation types and our system gave better results even with a simpler architecture as will further be explained in the sequel.

After reviewing all these works, it is noticeable that word embeddings combined with deep neural networks are almost the trend to follow when tackling the semantic relation classification task.

3   Proposed Approach

The approach proposed in this paper exploits WordNet as a dataset to learn how to classify selected ontological relations, using pre-trained word embeddings of the words[1] involved in these relations. First, there was a need to build the dataset, which will be explained in detail below. Then, the built dataset is used by a deep neural network to learn how to classify the semantic relation that exists between two words based only on their pre-trained word vectors.

In the sequel, we present the steps followed in building the dataset and the architecture adopted for the learning model.

3.1   Building the Dataset

To build the dataset, first we relied on WordNet. WordNet [16] is a lexical database that represents a semantic network where the nodes are synsets. A synset represents a concept in the database and it is formed by a group of unordered synonyms. For example: the concept "peace" represents a nominal synset formed with the synonyms "peace", "peacefulness", "peace_of_mind", "repose" and "serenity". The database contains words for different Part-Of-Speech (noun, verb, adverb and adjective) and covers several relations. Two types of relations could be distinguished: lexical links and semantic links. Lexical links are relationships that stand between word entities (which compose a synset). For example: the "antonym" relation is a lexical one. On the other hand, semantic links stand for relations between synsets (concepts). This is the case, for example, for the "hypernym" (is-a) relation.

Second, we used Word2vec word embeddings [14,15]. Word2vec covers two types of models for learning word embeddings, Continuous Bag Of Words (CBOW) and Skip-gram. Their architecture consists of a shallow two-layer neural network. Word2vec allows to model a vector space of several dimensions so as to represent a vocabulary of words by means of a vector of real numbers (a word embedding) for each word. Based on a large corpus of text, Word2vec builds a vocabulary of unique words with different contexts associated to them. With the CBOW model, a word embedding is produced by predicting the word from a window of its surrounding context words, while with Skip-gram, it is the window of surrounding context words that is predicted from the word. In both cases, word embeddings are represented by the learned weights

---

[1] The "words" are actually terms, which could be multi-word terms. For the sake of simplicity, we use "word" to refer to term/lemma in this article.

of the hidden layer and words with common contexts in the corpus have close vectors in the produced vector space. The CBOW model is faster to train if compared to Skip-gram, but the latter works better with small corpora and appropriately represents rare words. Word2vec embeddings have contributed to greatly improving the results of several natural language processing tasks.

To summarize, the building of the dataset is done following two steps:

– Select the relations to exploit from WordNet and then extract all the pairs of words linked by each of the considered relations in the format (relation type - $word_1$ - $word_2$).
– Replace the involved words by their pre-trained word vectors and generate the final dataset that consists of all these triplets where words are represented as word embeddings.

WordNet Relation Selection. WordNet contains a number of different semantic relations but not all of them are present in a large enough number of instances. Consequently, we have selected from WordNet only those relations that are reasonably well represented in the dataset so that an automatic training of a model that classifies the relations be possible. When the relations are very sparsely represented in the dataset, such a machine learning of the relations becomes impossible. The selected relations are:

– Hypernymy: syn_1 is a syn_2; e.g. "mammal" is an "animal".
– Instance Hypernymy: syn_1 is an instance of syn_2; e.g. "Barack_Obama" is an instance of "President".
– Part Holonymy: syn_1 is a component of syn_2; e.g. "branch" is a component of "tree".
– Member Holonymy: syn_1 is an element of syn_2; e.g. "tree" is an element of "forest".
– Substance Holonymy: syn_1 is a material of syn_2; e.g. "wood" is a material of "door".
– Similar: this relation holds between adjective and satellite adjective synsets. Satellite adjectives are adjectives associated with central ones in terms of meaning. For example, the adjective "accurate" has the satellite adjectives: "straight", "surgical", "true" and "veracious".
– Antonymy: $word_1$ is an antonym of $word_2$.
– Synonymy: $word_1$ is a synonym of $word_2$.

We have used a MySql[2] backup file of the WordNet database to facilitate the access to all WordNet relations[3], with the simple use of SQL queries. Table 1 presents a global description of the selected relations with indexes and statistics.

---

[2] https://www.mysql.com/fr/.
[3] https://vandanphadke.wordpress.com/2017/02/15/using-WordNet-as-a-lexical-database-in-applications/.

Table 1. The relations selected from WordNet to build the dataset.

Index	Relation name	Semantic link/Lexical link	Number of cases
0	Hypernymy	Semantic link	89,172
1	Instance hypernymy	Semantic link	8,589
2	Part holonymy	Semantic link	9,104
3	Member holonymy	Semantic link	12,288
4	Substance holonymy	Semantic link	797
5	Similar	Semantic link	21,434
6	Antonymy	Lexical link	7,981
7	Synonymy	Lexical link	133,692

It is important to mention that in Table 1 the semantic links involve synsetIDs, but our aim is to have links between words (wordIDs). In order to obtain pairs of words instead of pairs of synsets, we have retrieved all the words composing each synset in a pair and combined them in a cross-product-like fashion. For instance, let (synsetID$_1$, synsetID$_2$) be a pair of synsets linked with a specific relation in WordNet, and the words associated with synsetID$_1$ and synsetID$_2$ be, respectively, w_{11}, w_{12}, w_{13}, and w_{21}, w_{22}. The resulting pairs of words are (w_{11}, w_{21}), (w_{11}, w_{22}), (w_{12}, w_{21}), (w_{12}, w_{22}), (w_{13}, w_{21}) and (w_{13}, w_{22}).

It should be pointed out here that the size (133,692) in Table 1 for the Synonymy relation was fixed empirically. We have formed this set of pairs by combining wordIDs of the same synset, and since WordNet contains more than 117,000 synsets, the actual number of synonym pairs would be too big and would greatly imbalance the dataset if considered entirely.

Table 2 shows the number of word (lemma) pairs for each relation type after applying the process explained above.

Table 2. Statistics about the number of word pairs per semantic relation type.

Index	Relation name	Number of word (lemma) pairs
0	Hypernymy	329,011
1	Instance hypernymy	34,841
2	Part holonymy	39,162
3	Member holonymy	60,415
4	Substance holonymy	2,658
5	Similar	50,602
6	Antonymy	7,975
7	Synonymy	133,062

Replacing Words with Pre-trained Word Vectors. In this step, we have chosen to use word vectors that had been pre-trained on Wikipedia with Facebook's AI Research (FAIR) lab FastText library[4]. These word vectors have a dimension of 300 and were obtained using the skip-gram model described in [3] with the default parameters. This library could be seen as a dictionary where the keys are terms and the values their pre-trained vectors.

To build our final dataset, in the format (relation type - word_vector$_1$ - word_vector$_2$), we have retrieved from the FastText library the pre-trained vectors of all the words composing the triplets (relation type - word$_1$ - word$_2$) obtained in the previous step. In other words, we have to find, for each word (lemma) extracted from WordNet, a keyword that matches it in the library. However, since WordNet lemmas are not necessarily written as the automatically extracted words of the pre-trained vectors library, even when referring to the same meaning, the retrieving step was not a trivial task. For example, when performing a manual search, the WordNet lemma "aaland islands" was found in the FastText library as "aaland", "genus ustilaginoidea" was found as "ustilaginoidea", etc. Actually, there are many similar cases where the automatic matching operation was not obvious and managing it was out of the scope of this article. Nonetheless, we made some format changes such as adding/deleting space or hyphen (-) between words that compose the same lemma to match the way they may be written in the dataset. For example, "west bank" in WordNet was found as "westbank" in the FastText library. Anyway, there are still some WordNet lemmas that may not be found neither automatically nor manually, as is the case for the lemma "genus sarcorhamphus".

After replacing the words that compose the pairs described in Table 2 with their pre-trained word vectors, if found (automatically) in the FastText library, the size of the different sets naturally dropped. This is due to the constraints explained above. But, if compared to the sizes of the existing datasets in this

Table 3. Statistics about the semantic relations dataset (using word embeddings).

Index	Relation name	Number of word vector pairs
0	Hypernymy	165,157
1	Instance hypernymy	11,315
2	Part holonymy	13,860
3	Member holonymy	10,314
4	Substance holonymy	1,026
5	Similar	38,126
6	Antonymy	6,730
7	Synonymy	61,328
Total		**307,856**

---

field (see Sect. 2), the produced dataset has a size that is large enough to make it suitable for our purposes. Table 3 describes the final dataset format and statistics.

Finally, the dataset generated has dimension [307,856: 601], where 307,856 is the number of instances of word relations and 601 is the size resulting from the concatenation of two word embeddings in addition to an integer that represents the index of the class.

3.2   Learning Model

To train a classifier which learns from the built semantic relations dataset, we have used a Deep Neural Network (DNN) with two hidden layers. The model was built using Keras[5], which is an open source Python[6] library offering several implementations to simplify deep learning experimentations. We reshaped the input to (2, 300) to allow the neural network to detect parallel correlations between the two word vectors. Each of the two hidden layers contains 512 neurons and uses the ReLU activation function [1]. After each hidden layer, a dropout [9] of 0.2 is applied to prevent over-fitting. We used the RMSprop optimizer [2] with a batch size of 128 and categorical cross-entropy as the loss function. The target being the relation class, it has been encoded as 1-hot, so the output layer has the same number of neurons as that of semantic relations classes and
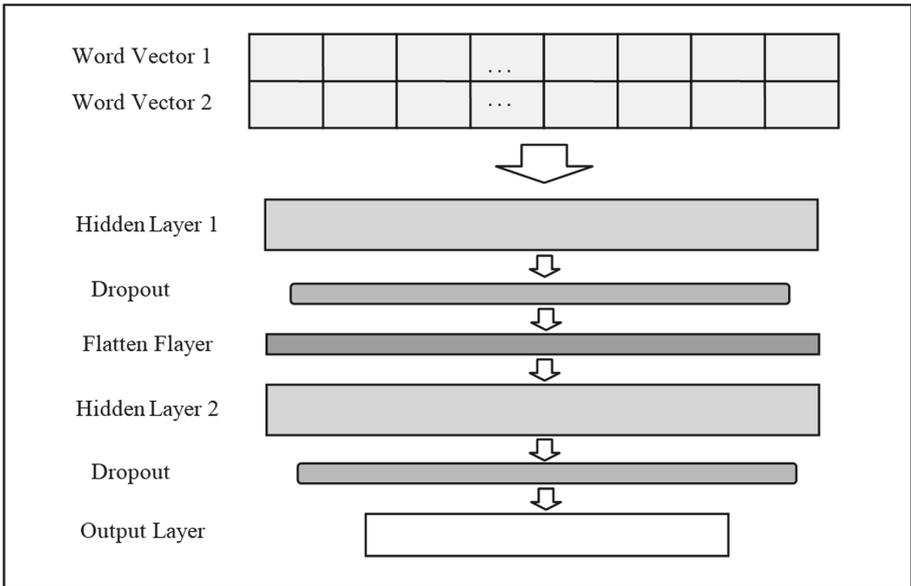

Fig. 1. Architecture of the deep neural network used for the classification of semantic relations.

---

[5] https://keras.io/.

[6] https://www.python.org/.

uses the Softmax activation function [7]. The hyper-parameters have been fixed empirically. Figure 1 illustrates the architecture of the DNN.

4   Experiments and Results

For the evaluation of the approach, three experiments have been performed. The purpose of these different experiments is to show different ways of exploiting the produced dataset. The first experiment aimed at classifying all the relation types represented in the dataset. The second, targeted four relation classes only (hypernymy, part holonymy, antonymy and synonymy), which represent the most widely used relations when aiming at building the structures of lexical ontologies. Finally, in the third experiment, we were interested in classifying the hypernym class against the others, given the fact that this relation plays an important role in building any ontology, since it constitutes its backbone.

4.1   Experiment 1: Full Unbalanced Dataset

Tests were performed using the entire, unbalanced dataset. The data was partitioned as follows: 232,000 instances for training, 58,000 for validation and 17,856 for testing.

On the validation set, an accuracy of 84.73% and a loss of 0.45 were obtained, and with the test set we obtained 84.87% accuracy and same loss. The detailed results are given in Table 4.

Table 4. Classification results for the first experiment.

	Precision		Recall		F1-score		Validation/Test set size	
Class index	Validation set	Test set	Validation set	Test set	Validation Set	Test set	Validation set size	Test set size
0	0.85	0.85	0.95	0.95	0.90	0.90	31,313	9,529
1	0.97	0.97	0.92	0.94	0.94	0.96	2,182	627
2	0.86	0.87	0.75	0.77	0.80	0.81	2,611	816
3	0.95	0.96	0.90	0.92	0.92	0.94	1,881	615
4	0.83	0.89	0.48	0.58	0.61	0.70	179	71
5	0.83	0.83	0.96	0.96	0.89	0.89	7,077	2,232
6	0.73	0.70	0.40	0.39	0.52	0.50	1,231	381
7	0.80	0.80	0.55	0.55	0.65	0.65	11,526	3,585
Avg[a]/Total	0.84	0.84	0.85	0.85	0.84	0.84	58,000	17,856

[a]The average is weighted according to the data distribution of validation and test sets since they are unbalanced in this case.

4.2   Experiment 2: Hypernymy, Part Holonymy, Antonymy and Synonymy Classes

The data was partitioned as follows: $(62,407_0 + 11,110_2 + 4,330_6 + 58,228_7)$ for training[7] (the subscript numbers represent the class indexes), $2,000 * 4$ for vali-

---

[7] For the hypernym class, we did not take all the available data in this experiment; this was done in order to make the data set more balanced with respect to the rest of the considered classes.

dation and $400*4$ for test. We note that we were able to balance the validation and test sets in this case, contrary to experiment 1.

On the validation set, the accuracy and loss were 77.76% and 0.89, respectively; and on the test set they were respectively 77.25% and 0.84. The detailed results are given in Table 5.

Table 5. Classification results for the second experiment.

	Precision		Recall		F1-score		Validation/Test set size	
Class index	Validation set	Test set	Validation set	Test set	Validation set	Test set	Validation set size	Test set size
0	0.74	0.73	0.80	0.80	0.77	0.76	2,000	400
2	0.97	0.98	0.83	0.82	0.90	0.90	2,000	400
6	0.98	0.98	0.61	0.59	0.75	0.74	2,000	400
7	0.60	0.60	0.86	0.88	0.71	0.71	2,000	400
Avg/Total	**0.82**	**0.82**	**0.78**	**0.77**	**0.78**	**0.78**	**8,000**	**1,600**

4.3 Experiment 3: Binary Classification - Hypernymy or Non-hypernymy Class

In this experiment we were interested in classifying the hypernymy class against all the other classes. There was one exception though which is the instance hypernymy class, which we considered to be too close to the hypernymy class to be part of the non-hypernymy class. As such, the non-hypernymy class was formed with a mix of the classes of indexes 2 to 7, which has led to a class of size 131,384.

We partitioned the data as follows: $96,000*2$ for training, $24,000*2$ for validation and $11,384*2$ for test. On the validation set the accuracy and loss were respectively 87.7% and 0.34; and on the test set they were respectively 87.98% and 0.34. The detailed results are given in Table 6.

5 Discussion

The results of the three experiments are encouraging. Indeed, in all cases the F1-score was above 78%, and even reached 88% in the third experiment. The built dataset may be useful for multi-way classification of several ontological relations,

Table 6. Classification results for the third experiment.

	Precision		Recall		F1-score		Validation/Test set size	
Class type	Validation set	Test set	Validation set	Test set	Validation set	Test set	Validation set size	Test set size
H. (0)	0.84	0.84	0.94	0.94	0.88	0.89	24,000	11,384
Non-H. (2–7)	0.93	0.93	0.82	0.82	0.87	0.87	24,000	11,384
Avg/Total	**0.88**	**0.88**	**0.88**	**0.88**	**0.88**	**0.88**	**48,000**	**22,768**

and also for binary classification to detect hypernym relations. The latter represents the backbone relation of ontologies and thus an ontology structure can be built using this binary classification. However, it should be noted that some relations were better classified than others. For example, in experiment 2, the precisions for synonymy (class 7) and hypernymy (class 0) are less than those for antonymy (class 6) and part holonymy (class 2). This could be explained by the fact that the former two relations may overlap in some samples. Also, we note that the ambiguity is ignored in these experiments, since we used pre-trained word embeddings, i.e. each word has one vector that corresponds to it, even if the word is polysemous. This aspect may also have affected the results.

6   Conclusion

This work falls under multi-way relation classification problems, with a more specific focus on the task of word semantic relation classification, in the context of ontological relations. We have proposed a way to exploit WordNet and pre-trained word embeddings to produce a dataset to be used in the problem of semantic relation classification. The dataset contains 307,856 labeled word pairs and 8 relation types (Hypernymy, Instance Hypernymy, Part Holonymy, Member Holonymy, substance Holonymy, Similar, Antonymy and Synonymy). To the best of our knowledge, it is the first time a dataset of this kind is built and we presented in detail the steps to construct it, so as to allow the process to be repeated with WordNets of other languages. We have implemented a deep neural network classifier which we trained and tested using the dataset. We performed three different experiments that show several ways of exploiting the produced dataset. In the first experiment, we exploited all the dataset so as to observe the capability of the proposed learning model to classify all the considered relations. In the second one, we focused on four relations (Hypernymy, Part Holonymy, Antonymy and Synonymy), which are frequently used relations when aiming at building the structures of lexical ontologies. Finally, the third experiment narrowed the classification of the Hypernymy class against the others to tackle a specific task when learning ontologies, which is the construction of its taxonomy (its backbone). The results were very encouraging and support our assumption that pre-trained word vectors encapsulate information which is useful for classifying relations between pairs of words.

For future work, we intend to refine our experiments by highlighting the way each class is learned by the system, testing our classifier on other resources like BabelNet [17], and proposing a way to take into account word ambiguity when handling the semantic relation classification problem. Furthermore, we plan to develop a system to supply our classifier to perform semantic relation extraction. This system will aim at extracting related word pair candidates from raw text and passing them to the classifier to detect the relation type that may exist between the word pairs.

References

1. Agarap, A.F.: Deep learning using rectified linear units (ReLU). arXiv preprint arXiv:1803.08375 (2018)
2. Dauphin, Y.N., de Vries, H., Bengio, Y.: Equilibrated adaptive learning rates for non-convex optimization. arXiv preprint arXiv:1502.04390 (2015)
3. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. Trans. Assoc. Comput. Linguist. **5**, 135–146 (2017)
4. Bui, V.T., Nguyen, P.T., Pham, V.L., Ngo, T.Q.: A neural network model for efficient antonymy-synonymy classification by exploiting co-occurrence contexts and word-structure patterns. Int. J. Intell. Eng. Syst. **13**(1), 156–166 (2020)
5. Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., Kuksa, P.: Natural language processing (almost) from scratch. J. Mach. Learn. Res. **12**, 2493–2537 (2011)
6. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
7. Dunne, R.A., Campbell, N.A.: On the pairing of the softmax activation and cross-entropy penalty functions and the derivation of the softmax activation function. In: Proceedings of the 8th Australian Conference on the Neural Networks, Melbourne, vol. 181, p. 185. Citeseer (1997)
8. Gábor, K., Buscaldi, D., Schumann, A.K., QasemiZadeh, B., Zargayouna, H., Charnois, T.: Semeval-2018 task 7: semantic relation extraction and classification in scientific papers. In: Proceedings of the 12th International Workshop on Semantic Evaluation, pp. 679–688 (2018)
9. Gal, Y., Ghahramani, Z.: Dropout as a Bayesian approximation: representing model uncertainty in deep learning. In: International Conference on Machine Learning, pp. 1050–1059 (2016)
10. Hearst, M.A.: Automatic acquisition of hyponyms from large text corpora. In: Proceedings of the 14th Conference on Computational Linguistics, vol. 2, pp. 539–545. Association for Computational Linguistics (1992)
11. Hendrickx, I., et al.: SemEval-2010 task 8: multi-way classification of semantic relations between pairs of nominals. In: Proceedings of the 5th International Workshop on Semantic Evaluation, pp. 33–38. Association for Computational Linguistics, Uppsala (2010). https://www.aclweb.org/anthology/S10-1006
12. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**(7553), 436–444 (2015)
13. Li, Q., Li, L., Wang, W., Li, Q., Zhong, J.: A comprehensive exploration of semantic relation extraction via pre-trained CNNs. Knowl.-Based Syst. **194**, 105488 (2020). https://doi.org/10.1016/j.knosys.2020.105488. ISSN 0950-7051
14. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
15. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems, pp. 3111–3119 (2013)
16. Miller, G.A.: WordNet: An Electronic Lexical Database. MIT Press, Cambridge (1998)
17. Navigli, R., Ponzetto, S.P.: Babelnet: the automatic construction, evaluation and application of a wide-coverage multilingual semantic network. Artif. Intell. **193**, 217–250 (2012)

18. Qin, P., Xu, W., Guo, J.: An empirical convolutional neural network approach for semantic relation classification. Neurocomputing **190**, 1–9 (2016)
19. dos Santos, C., Tan, M., Xiang, B., Zhou, B.: Attentive pooling networks. arXiv preprint arXiv:1602.03609 (2016)
20. Shijia, E., Jia, S., Xiang, Y.: Study on the Chinese word semantic relation classification with word embedding. In: National CCF Conference on Natural Language Processing and Chinese Computing, pp. 849–855. Springer, Cham (2017)
21. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press, Cambridge (2018)
22. Wu, S., He, Y.: Enriching pre-trained language model with entity information for relation classification. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management, pp. 2361–2364 (2019)
23. Wu, Y., Zhang, M.: Overview of the NLPCC 2017 shared task: Chinese word semantic relation classification. In: National CCF Conference on Natural Language Processing and Chinese Computing, pp. 919–925. Springer, Cham (2017)
24. Zeng, D., Liu, K., Lai, S., Zhou, G., Zhao, J.: Relation classification via convolutional deep neural network. In: Proceedings of COLING 2014, The 25th International Conference on Computational Linguistics: Technical Papers, pp. 2335–2344. Dublin City University and Association for Computational Linguistics, Dublin, August 2014. https://www.aclweb.org/anthology/C14-1220

Gender Identification from Arabic Speech Using Machine Learning

Skander Hamdi[1]([⊠]) , Abdelouahab Moussaoui[1] , Mourad Oussalah[2] ,
and Mohamed Saidi[1]

[1] Department of Computer Science, University of Ferhat Abbes Setif I, Setif, Algeria
{skander.hamdi,abdelouahab.moussaoui,mohamed.saidi}@univ-setif.dz
[2] Department of Computer Science and Engineering, University of Oulu,
Oulu, Finland
mourad.oussalah@oulu.fi

Abstract. Speech recognition is becoming increasingly used in real-world applications. One of the interesting applications is automatic gender recognition which aims to recognize male and female voices from short speech samples. This can be useful in applications such as automatic dialogue systems, system verification, prediction of demographic attributes (e.g., age, location) and estimating person's emotional state. This paper focuses on gender identification from the publicly available dataset Arabic Natural Audio Dataset (ANAD) using an ensemble-classifier based approach. More specifically, initially we extended the original ANAD to include a gender label information through a manual annotation task. Next, in order to optimize the feature engineering process, a three stage machine learning approach is devised. In the first phase, re restricted to features to the two widely used ones; namely, MFCC and fundamental frequency coefficients. In the second phase, six distinct acoustic features were employed. Finally, in the third phase, the features were selected according to their associated weights in Random Forest Classifier, and the best features are thereby selected. The latter approach enabled us to achieve a classification rate of 96.02% on the test set generated with linear SVM classifier.

Keywords: Speech features · Feature selection · Arabic speech · Gender identification · Machine learning · Deep learning

1 Introduction

Speech recognition is the process by which a computer system maps an acoustic speech signal to some form of abstract meaning of speech using a set of predefined acoustic features. Speech recognition is becoming increasingly used in real-world applications such as speaker identification for securing access to confidential information or virtual spaces, automatic reading of text or dictionaries, recognizing and understanding of speech for control applications. Therefore, speech

recognition has become a growing field of research in data science and machine learning. Gender identification is another application of voice recognition. The differences in a human speech concerning gender are basically due to physiological characteristics: vocal fold thickness, vocal tract length. We can observe these differences in the speech signal and also use them to make a gender-based classification model that distinguishes between male and female from a short voice recording.

In this paper, we will build a gender classification speech-based model using a modified version of an arabic speech database where utterances are pronounced by only Arabic native speakers. Arabic language was chosen for its special characteristics compared to the Latin languages used in all related works [25]. Low levels descriptors have been extracted during the pre-processing phase. Our methodology uses a set of machine learning classifiers, including KNN, Naives Bayes, Decision Tree, Logistic Regression, Random Forest, Linear SVM, Kernel SVM, ANN, CNN, which enable comprehensive comparison. Furthermore, in order to investigate the feature engineering process, a three step-strategy has been developed where the feature set is selected according to the two most widely used features, all potential acoustic features, selected according to their weights in Random Forest classifier. We achieved 96.02% of test accuracy using Linear SVM. The second section of this paper will give a brief introduction to the used speech features while the third section will give a review of related works in gender identification. The fourth section will present the used database. The Methodology, experimental results are highlighted in the fifth and sixth section, respectively. Finally, discussion and conclusions are drawn in the emphasizes the methodology, the sixth will present and discuss the experiments results and the last one conclude the work with some perspectives.

2   Acoustic Features

Speech recognition tasks involve extracting a set of speech features from the speech dataset. One distinguishes three groups of features: spectral, excitation and acoustic features [2]. Spectral and acoustic features are the most relevant to our work.

2.1   Intensity

Sound Intensity models the loudness of the sound signal [1,2]. It is known as the power of the sound waves in an area unit in a perpendicular direction. Measured by watt per square meter (W/m^2). Sound intensity can be calculated with the following equation [15]:

$$I = \frac{P}{A} \tag{1}$$

where P is the power of sound and A is a distance.

In order to measure the sound intensity relatively to a reference, we can calculate the **acoustic intensity level** with the following formula [16]:

$$L = 10 \times \log_{10} \frac{I}{I_0} (dB) \qquad (2)$$

where I is the sound intensity and I_0 is a reference value. Sound intensity level is expressed in decibels (dB).

2.2   Zero-Crossing Rate

Zero-crossing rate is the rate at which a signal changes its sign during the frame (measure the duration) [1,2]. Formally defined as following [17]:

$$zcr = \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{I}\{s_t s_{t-1} < 0\} \qquad (3)$$

2.3   Fundamental Frequency

Fundamental frequency (F0) measures the pitch. It is defined as the vibration frequency of vocal chords. Defined as the lowest frequency of a periodic waveform [18]. F0 can be extracted using different methods such as Pitch Detection Algorithms [19,20].

2.4   Mel Frequency Cepstral Coefficients (MFCC)

MFCC takes into account human perception for sensitivity at appropriate frequencies by converting the conventional frequency to Mel Scale (4). To calculate MFCCs, some steps should be followed [21]:

1. Calculating the Fourier transform of a signal.
2. Mapping the result of the previous step to the mel scale (4) using triangular overlapping windows.
3. Calculating the logarithm of the powers at each of the calculated mel frequencies.
4. Calculating the discrete cosine transform of the mel logarithm powers.
5. MFCCs are the amplitudes of the resulting spectrum.

$$mel = 2595 \times \log_{10} \frac{1 + hertz}{700} \qquad (4)$$

2.5   Probability of Voicing

It evaluates the probability that the voice is *voiced* or *unvoiced*. The voiced type is characterized by the periodic structure when the unvoiced presents basically a noise. Some methods to estimate the probability of voicing were discussed in [22].

2.6   Line Spectral Frequency (LSP)

Line Spectral Pairs (LSP) are popular alternative representation of Linear Prediction Coefficients (LPC). LSPs are useful for speech coding as they have some properties that make them superior to direct quantization of LPCs [23].

3   Related Works

Several works have been published in order to identify gender from speech using machine learning and deep learning approaches. Most of works we surveyed used MFCC and F0 as features and different classification methods. For instance Rami and Alkhawaldeh [3] used *Mel-spectrogram*, *MFCCs*, *Chroma-STFT*, *Spectral_Contrast* and *Tonnetz* features for a gender recognition task through a neural network based approach. For the learning and classification methods, they provide several experiments using machine learning techniques family: Bayesian Network, Naive Bayes, MLP (Multi-Layer Perceptron), Logistic Regression, SMO (sequential minimal optimization for SVM), Linear SVM, Kernel SVM (Polynomial, Radial), Latent Dirichlet allocation (LDA), Lazy learners (IBk and KStar), AdaBoost, Decision trees, random forest and three rule-based methods (OneR, Ridor and Rough Set). For deep learning approaches, they used 1D Convolutional neural network architecture composed of three convolution layers with 32, 48,120 neurons with ReLu activation function, one maxpooling layer and two fully connected layers with 128 and 64 units and an output layer with softmax activation function with two units according to class labels *male,female*. They used also some feature selection techniques to choose the relevant ones, *Evolutionary search*, *PSO search* and *Wolf search* were used. The ROC curve (AUC) was used to verify if a classifier can separate the two classes. Using a dataset of artificial voices consisting of 20 languages, each one composed of 16 voice samples (8 for male, 8 for female), they obtained the following results: as precision/recall metric, Kernel SVM (Polynomial) yields 100% while Logistic Regression yields 99.7% and 99.4% for Random Forest method.

Kabil et al. [5] proposed a deep learning approach using two datasets called AVspoof and ASVspoof 2015, different convolutional neural network (CNN) architectures have been proposed by giving the raw speech signal to the network's input to make automatically the feature learning phase by the convolution layers. To make comparisons, two artificial neural network (ANN1 + ANN2) architectures have been proposed after acoustic features extraction. The first one with MFCC feature 342 as input dimension and the second with MFCC+F0 and 351 as input dimension. Three CNN architectures with different hyper-parameters have been tested, the last one (cnn3) outperforms cnn1,cnn2, and the first baseline proposed system MFCC+F0 based. After CNN analysis, it has been shown that CNN learn formant and fundamental frequency.

Doukhan et al. [6] focused on the French corpus called REPERE where raw audio streams were used. The authors compared Gaussian Mixture Models (GMM), i-vector, and CNNs models for automatic gender identification task. In [7], a CNN based speech segmentation was used to eliminate music and empty

speech segments, low energy frames using fixed thresholding-based approach. Next, for the feature extraction step, the authors used SIDEKIT. As result, the CNN model gave the best Recall measure, 98.04% for male, 95.05% for female, and 96.52% for F1 measure.

Levitan et al. [9] used pitch and spectral features by comparing each one to others. To add more information to the features; minimum, maximum, median, mean, and standard deviation have been calculated for each value of f0 trajectory, and 21 MFCCs used as features. A database called HMIHY composed of 5002 from 1654 speakers is used with four classifiers: Logistic Regression, linear regression, random forest, AdaBoost. Logistic regression model gave the high classification rate of 95.2% using the fundamental frequency and MFCCs. In another experiment, a benchmark called aGender for german speech, has three class labels: *male, female* and *children* and to compare the previous results with german speech data classification, f0 statistics have been supplemented with another statistics (f0+). Using all features (f0, f0+, and MFCCs), random forest classifier gave the best classification rate of 85.0% close to the best achieved results in the challenge [1]. Compared to the previous results, this degradation is due to the existence of a third class and the challenge *children vs female* because of the high pitched speech in both classes which makes the problem more complicated. To validate and ensure the model performance, a cross-lingual gender detection has been presented by training different learners on the english corpus HMIHY and test the model with non-english data (aGender by eliminating children samples). The cross-lingual model with Logistic Regression learner gave 92.1% while german-only model gave 93.0% using random forest classifier.

Ioannis et al. [4] proposed a new ensemble semi-supervised self-labeled algorithm to build a more accurate gender classifier by exploiting the problem of the non-existence of sufficient data to build an efficient model. They proposed an algorithm called *iCST-Voting* which combines the predictions of *Co-training, Self-training*, and *Tri-training* using an ensemble as base learning. The main methodology of the proposed framework is composed of two steps: **Training** step where three classifiers are trained using *Co-training* C_{co}, *Self-training* C_{self} and *Tri-training* C_{Tri} algorithms. **Voting** step where each trained classifier (C_{co},C_{self} and C_{Tri}) is applied to every unlabeled sample on the test set, the final label will be the majority voting. Two datasets were used, Voice gender dataset and Deterding dataset, pre-processed by acoustic analysis using **seewave** and **tuneR** libraries in **R**. As classifiers: SMO, decision trees C4.5, and kNearest Neighbor were used. Comparing to the state-of-the-art, iCST-Voting outperforms all previous work in self-labeled algorithms for both datasets.

Kaushik et al. [13] used a database of 200 voice samples for celebrities 50% for males and 50% for females. They proposed a combination of two methods, called combo-classifier which use the Modified autocorrelation method and average magnitude difference function by weighting each of them. Combo-classifier achieved 99%.

Ali [14] proposed a system with front-end and back-end, the first one extracts useful feature using First Fourier Transform (FTT) algorithm and the second as

a classifier which classifies each speech signal ("A" or "B") as male/female. Classification was based on the frequency at maximum power which was extracted from estimated power spectrum and gave an average recognition accuracy of 80%.

In [8], a feed-forward multi-layer perceptron FF-MLP has been proposed after feature extraction and speech analysis pipeline. Also, different order of Linear Prediction Coefficients (LPC) filter is applied using autocorrelation function (ACF) to produce LPC coefficients where they are used as neural network input. The training was done using Levenberg-Marquardt learning algorithm. Different LPC orders have been tested from 8 to 20 and the LPC-18 gave the best rate of 95.5%.

There is no paper yet that proposes arabic data to build a gender recognition model speech-based, in this paper, we will propose a model based on machine learning and deep learning techniques using the pitch and spectral features discussed in the state-of-the-art. We will use a modified version of an arabic speech database used for emotions recognition.

4   Arabic Natural Audio Dataset (ANAD)

In the following section, we will present the used database, our modification which allows us to use it for the aim of building a gender recognition speech-based model.

4.1   Dataset Description

Arabic Natural Audio Dataset (ANAD) is used in [1,11] for emotion recognition, the dataset is available online in Kaggle[1]. Live calls between an anchor and a human outside the studio were downloaded from online Arabic talk shows from different dialects (Egyptian, Gulf, Jordan and Lebanese). Each video was divided into turns: callers and receivers. To label each video, 18 listeners were asked to listen to each video and select an emotion: happy, angry, or surprised. Silence. Laughs and noisy chunks were removed. Every chunk was divided into 1 s speech units forming the final corpus composed of 1383 records.

4.2   Modified Version

To build a gender recognition model using this database, the label *male, female* of each record was manually added by giving all the voice recordings to 3 listeners, each one of them listen and note the corresponding class label to each voice recording, in case of disagreement, we took the majority vote as the correct class. The following table presents the gender label distribution statistics.

---

[1] https://www.kaggle.com/suso172/arabic-natural-audio-dataset.

Table 1. Gender class label in the new ANAD.

Class label	Samples count
Male	736
Female	647

5   Methodology

The dataset has been pre-processed to build an emotion recognition model in [1] by extracting pitch and spectral features or what is called low-level descriptors [12]. These features are: fundamental frequency f0, f0 envelope, intensity, zero crossing rates, 12 MFCCs, 7 LSP frequencies, probability of voicing. Some statistics are calculated for each one of these descriptors: maximum, minimum, range, absolute position of maximum, absolute position of minimum, arithmetic of mean, Linear Regression1, Linear Regression2, Linear RegressionA, Linear RegressionQ, Standard Deviation, kurtosis, skewness, quartiles 1, 2, 3 and, inter-quartile ranges 1–2, 2–3, 1–3. These features have been used in three stages. In the first stage, we used only MFCCs, F0 features as reported in [5,9], while all features have been employed in the second stage. In order to improve the results of the two stages, we used the feature weighted provided by the Random Forest algorithm to select the best and important features by measuring the impurity (*Gini impurity* in our case). The more the impurity decreases, the more important the feature is [24].

Different classifiers have been proposed to build a gender classification model and used for the three stages, as machine learning methods family: **k-Nearest Neighbor**, **Naive Bayesian**, **Decision Trees C4.5**, **Logistic Regression**, Ensemble learning using **Random Forest** with hyper-parameter search and 5-fold cross-validation to find the best parameter for the algorithm, **Linear SVM** and **Kernel SVM** with hyper-parameter search from python scikit-learn library.
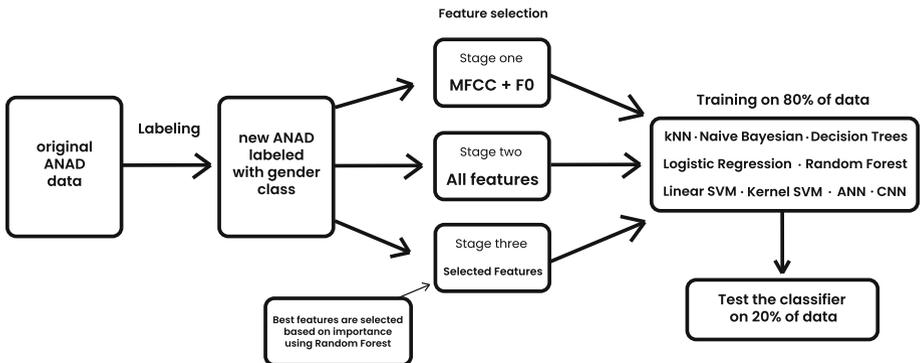
Fig. 1. Summary of proposed methodology

For deep learning approaches family, using Keras and Tensorflow, we proposed Artificial Neural Network (ANN) and Convolutional Neural Network (CNN) architectures based on many experiments. The dataset is divided into 80% of training and the remaining 20% for test. The Fig. 1 presents our methodology and the Table 2 shows a summary of the used classifiers for our problem.

6   Experiments Results

Training was done on a local PC with the following configuration: MacOS Catalina 10.15.4 16 GB of memory 2.3 GHz Quad-Core Intel Core i7 as CPU

Table 2. Proposed methods for building gender identification from arabic speechProposed methods for building gender identification from arabic speech

Method	Parameters
k-Nearest Neighbor	k = 3
Naive Bayesian	default scikit-learn parameters
Decision Trees C4.5	default scikit-learn parameters
Logistic Regression	default scikit-learn parameters
Random Forest	n_estimators = 800
	min_samples_split = 2
	min_samples_leaf = 1
	max_depth = 100
Linear SVM	kernel = linear
Kernel SVM	kernel = rbf
	C = 1
	gamma = 0.05
ANN architecture	Dense(units = 32, activation=relu) + Dropout(0.2)
	Dense(units = 16, activation = relu) + Dropout(0.1)
	Dense(units = 1, activation = sigmoid)
	Number of epochs = 70
	Batch size = 64
	optimizer = adam
	loss = binary_crossentropy
CNN architecture	Conv1D(filters = 16, filter_size = 2, activation = relu)
	Conv1D(filters = 16, filter_size = 2, activation = relu) + Dropout(0.1)
	MaxPooling1D(window_size = 2)
	Conv1D(filters = 8, filter_size = 2, activation = relu)
	Conv1D(filters = 16, filter_size = 2, activation = relu) + Dropout(0.1)
	Dense(units = 50, activation = relu)
	Dense(units = 1, activation = sigmoid)
	Number of epochs = 70
	Batch size = 64
	optimizer = adam
	loss = binary_crossentropy

and NVIDIA GeForce GT 750M 2 GB as GPU and after several experiments in the three stages for each one of algorithms, we obtained the following results:

6.1   Stage One

The result shows that CNN classifier presents the the worst test accuracy of 59.92%, while its training accuracy is 95.11%, which indicates that the proposed CNN architecture cannot accommodate perfectly the problem under consideration, while the highest accuracy 93.86% is achieved using Logistic Regression and Linear SVM with a training accuracy of 93.85% and 97.46%, respectively.

Table 3. Experiments results using proposed methodology for the first stage

Algorithm	Train Acc. [%]	Test Acc. [%]
k-Nearest Neighbor	92.04	84.83
Naive Bayesian	82.73	81.94
Decision Trees C4.5	100	80.86
Logistic Regression	93.85	**93.86**
Random Forest	100	93.14
Linear SVM	97.46	**93.86**
Kernel SVM	93.58	92.41
ANN	92.67	92.05
CNN	95.11	59.92
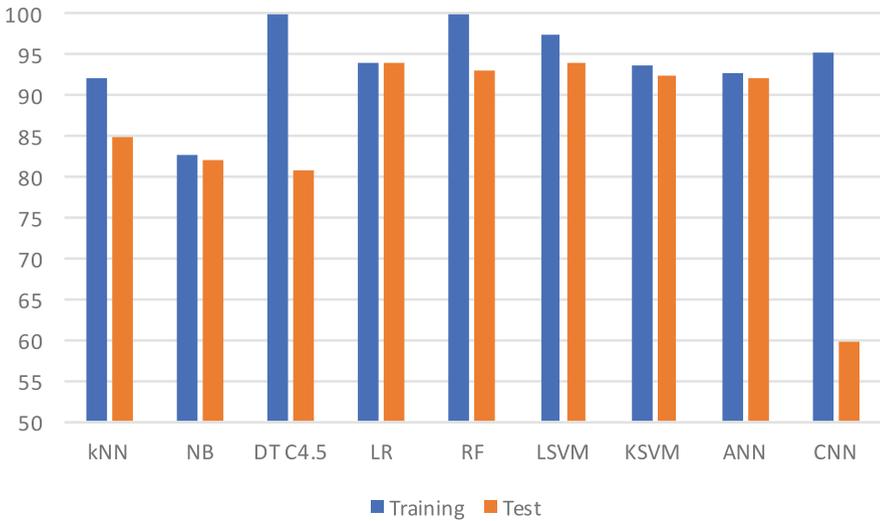
Fig. 2. Summary of achieved results in the first stage which show that Logistic Regression and Linear SVM did better than other methods

The Table 3 shows the achieved results using all methods when only MFCCs and F0 features are used as input (434 features in total).

6.2   Stage Two

In order to improve the results of the first stage, we tried to use all remaining features with their calculated statistics: LSP frequencies, f0 envelope, intensity, zero crossing rates, probability of voicing. All classification rates have been improved in the second stage. Naive bayesian classifier achieves the worst test accuracy of 86.28% with 84.62% training accuracy, while the best test classification rate of 95.30% is achieved using Logistic Regression where, out of a total of 277 test samples, 119 were correctly classified as female and 145 as male, and only 13 misclassifications. This is followed by Kernel SVM that achieved 94.94% test accuracy, corresponding to 115 correctly classified female, 148 correctly classified male and 14 misclassifications. Although there is a significant improvement

Table 4. Experiments results using proposed methodology for the second stage

Algorithm	Train Acc. [%]	Test Acc. [%]
k-Nearest Neighbor	94.21	91.69
Naive Bayesian	84.62	86.28
Decision Trees C4.5	100	88.08
Logistic Regression	97.46	**95.30**
Random Forest	100	94.58
Linear SVM	99.81	93.86
Kernel SVM	96.20	94.94
ANN	92.22	94.22
CNN	96.38	91.69
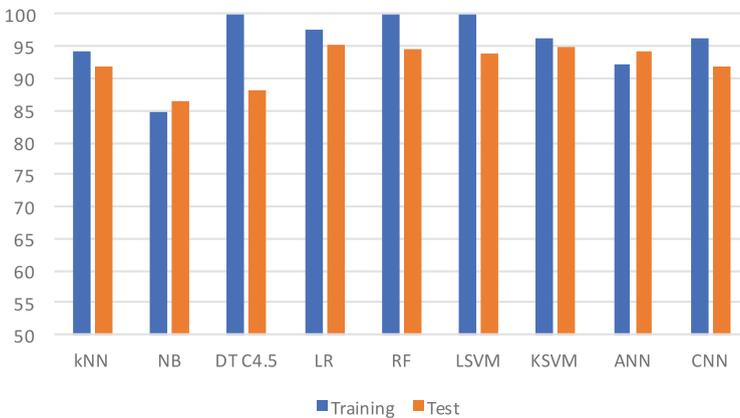
Fig. 3. Summary of obtained results in the second stage which show that Logistic Regression outperform the other methods followed by random forest

in the CNN result which means that the previous model is more complex than that of Stage one. However, the additional features added more informative patterns to the data. Table 4 shows the achieved results using all methods when all features are used (844 features in total).

6.3   Stage Three

A last experiment employs Random Forest method to extract the best and important features. Based on the same hyper-parameters found by grid search, we train again a random forest classifier. A set of 163 features has been found including MFCC, LSP frequencies, Zero Crossing Rates, F0, F0 envelope and Probability of voicing. The highest values of importance from the first to 43th

Table 5. Experiment results using proposed methodology for the third stage

Algorithm	Train Acc. [%]	Test Acc. [%]
k-Nearest Neighbor	95.11	94.22
Naive Bayesian	88.60	88.44
Decision Trees C4.5	100	90.61
Logistic Regression	93.94	93.86
Random Forest	100	95.66
Linear SVM	96.02	**96.02**
Kernel SVM	93.67	94.94
ANN	94.12	93.14
CNN	95.56	88.44
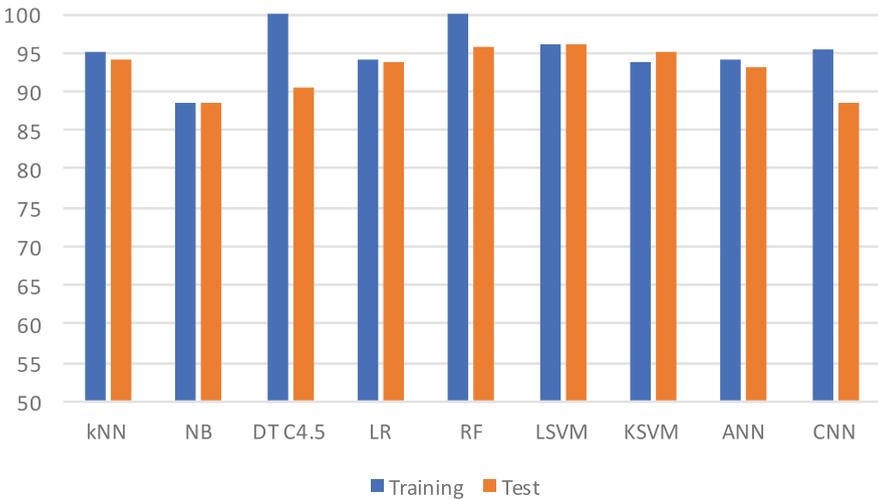
**Fig. 4.** Summary of obtained results in the third stage which show that Linear SVM outperform the other methods followed by random forest

feature were taken by MFCC features which show the importance of MFCCs and validate the results of stage one. We used the extracted features to re-train and improve the classification rate. We achieved the best test accuracy of 96.02% using Linear SVM classifier on 277 test samples when 122 are correctly classified as female and 144 as male with 11 misclassifications followed by Random Forest with test accuracy of 95.66% where 116 are correctly classified as female and 149 as male with 12 misclassifications. Table 5 presents the experiment results for the third stage.

The following Table 6 and Fig. 5 summarize all obtained experiments results.

Table 6. Summary of test accuracy for each method in all stages

| Algorithm	Stage one [%]	Stage two [%]	Stage three [%]
k-Nearest Neighbor	84.83	91.69	94.22
Naive Bayesian	81.94	86.28	88.44
Decision Trees C4.5	80.86	88.08	90.61
Logistic Regression	**93.86**	**95.30**	93.86
Random Forest	93.14	94.58	95.66
Linear SVM	**93.86**	93.86	**96.02**
Kernel SVM	92.41	94.94	94.94
ANN	92.05	94.22	93.14
CNN	59.92	91.69	88.44
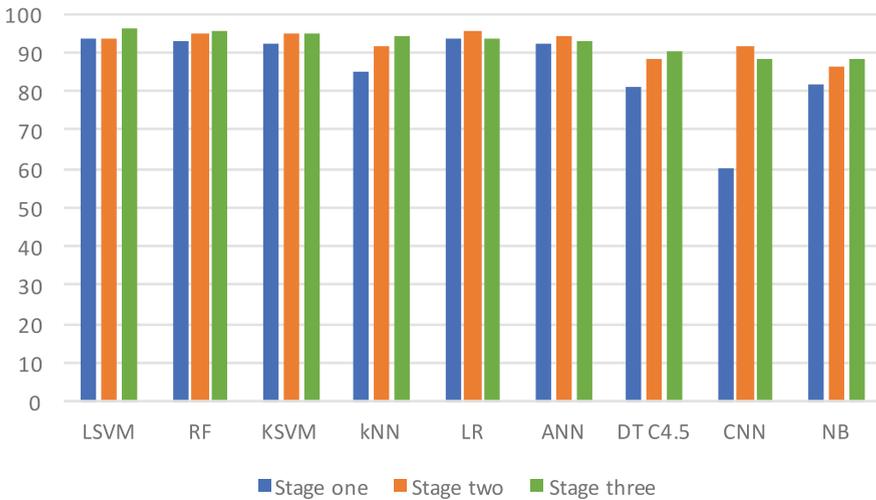
Fig. 5. Summary of test accuracy using all methods in the three stages which show the performance of Linear SVM after selecting important features

7 Conclusion and Future Work

There are a lot of researches that have used benchmarks in different languages such as English, French, German to build gender recognition models from natural speech recordings. In this paper, we used an arabic natural audio dataset with a comparison of different machine learning and deep learning approaches. In order to investigate the influence of the features on the overall approach, we devised a three stage strategy. The first stage uses only MFCCs and F0 as features. In the second stage, all features proposed in [1] were employed in each classifier. This includes acoustic intensity, which measures the power of the sound waves, Zero Crossing Rate that measures the sign signal change rate, among others. The third and last stage uses Random Forest for selecting the most important features in order to use only the discriminant ones for improving the classification rate. We have extracted 163 important features and we succeeded in constructing a model whose classification rate for the test set is 96.02% using Linear SVM which gave the best results for the first and last stages. We also comprehended the importance of speech features variety (pitch and spectral) for classification tasks or speech recognition to obtain good results. As future work, we will try to combine ANAD with another database which includes children's speech recordings and trying also to make a cross-lingual model trained on arabic data for non-arabic speech gender detection.

References

1. Klaylat, S., Osman, Z., Hamandi, L., et al.: Emotion recognition in Arabic speech. Analog Integr. Circ. Sig. Process **96**, 337–351 (2018). https://doi.org/10.1007/s10470-018-1142-4
2. Klaylat, S., Osman, Z., Hamandi, L., et al.: Enhancement of an Arabic speech emotion recognition system. Int. J. Appl. Eng. Res. **13**(5), 2380–2389 (2018). ISSN 0973–4562
3. Rami, S., Alkhawaldeh, D.G.R.: Gender recognition of human speech using one-dimensional conventional neural network. Sci. Program. (2019). https://doi.org/10.1155/2019/7213717. ISSN: 1058–9244
4. Livieris, I., Pintelas, E., Pintelas, P.: Gender recognition by voice using an improved self-labeled algorithm. Mach. Learn. Knowl. Extr. **1**, 492–503 (2019). https://doi.org/10.3390/make1010030
5. Kabil, S., Muckenhirn, H., Magimai-Doss, M.: On learning to identify genders from raw speech signal using CNNs, pp. 287–291 (2018). https://doi.org/10.21437/Interspeech.2018-1240
6. Doukhan, D., Carrive, J., Vallet, F., Larcher, A., Meignier, S.: An open-source speaker gender detection framework for monitoring gender equality (2018). https://doi.org/10.1109/ICASSP.2018.8461471
7. Doukhan, D., Carrive, J.: Investigating the use of semi-supervised convolutional neural network models for speech/music classification and segmentation (2017)

8. Yusnita, M.A., Hafiz, A.M., Fadzilah, M.N., Zulhanip, A.Z., Idris, M.: Automatic gender recognition using linear prediction coefficients and artificial neural network on speech signal. In: 2017 7th IEEE International Conference on Control System, Computing and Engineering (ICCSCE) (2017). https://doi.org/10.1109/iccsce.2017.8284437

9. Levitan, S., Mishra, T., Bangalore, S.: Automatic identification of gender from speech, pp. 84–88 (2016). https://doi.org/10.21437/SpeechProsody.2016-18

10. Meinedo, H., Trancoso, I.: Age and gender classification using late fusion of acoustic and prosodic features. In: Proceedings of Interspeech 2010, Makuhari, Japan, pp. 2818–2821 (2010)

11. Klaylat, S., Osman, Z., Hamandi, L., et al.: Enhancement of an Arabic speech emotion recognition system. Int. J. Appl. Eng. Res. **13**(5), 2380–2389 (2018). ISSN 0973–4562

12. Low, L.A., Maddage, N.C., Lech, M., Sheeber, L., Allen, N.: Influence of acoustic low-level descriptors in the detection of clinical depression in adolescents. In: 2010 IEEE International Conference on Acoustics, Speech and Signal Processing, Dallas, TX, pp. 5154–5157 (2010). https://doi.org/10.1109/ICASSP.2010.5495018

13. Kaushik, D., Jain, N., Majumdar, A.: Gender Voice Recognition through speech analysis with higher accuracy (2014). https://doi.org/10.13140/2.1.1331.5842

14. Ali, M.: Gender recognition system using speech signal. Int. J. Comput. Sci. Eng. Inf. Technol. **2**, 1–9 (2012). https://doi.org/10.5121/ijcseit.2012.2101

15. Letter symbols to be used in electrical technology – Part 3: Logarithmic and related quantities, and their units, IEC 60027-3 Ed. 3.0, International Electrotechnical Commission, July 19, 2002

16. Fahy, F.: Sound Intensity. CRC Press, London (2017). ISBN 978-1138474192. OCLC 1008875245

17. Gouyon, F., Pachet, F., Delerue, O.: On the use of zero-crossing rate for an application of classification of percussive sounds (2002)

18. Fundamental frequency, Pitch, F0. In: Li, S.Z., Jain, A. (eds.) Encyclopedia of Biometrics. Springer, Boston (2009)

19. Tan, L., Karnjanadecha, M.: Pitch detection algorithm: autocorrelation method and AMDF (2003)

20. Drugman, T., Huybrechts, G., Klimkov, V., Moinet, A.: Traditional machine learning for pitch detection. IEEE Sig. Process. Lett. **PP**(99), 1 (2018). https://doi.org/10.1109/LSP.2018.2874155

21. Sahidullah, M., Saha, G.: Design, analysis and experimental evaluation of block based transformation in MFCC computation for speaker recognition. Speech Commun. **54**(4), 543–565 (2012). https://doi.org/10.1016/j.specom.2011.11.004

22. Rehr, R., Krawczyk, M., Gerkmann, T.: A posteriori voiced/unvoiced probability estimation based on a sinusoidal model. In: 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Florence, pp. 6944–6948 (2014). https://doi.org/10.1109/ICASSP.2014.6854946.

23. Sahidullah, M., Chakroborty, S., Saha, G.: On the use of perceptual Line Spectral pairs Frequencies and higher-order residual moments for Speaker Identification. Int. J. Biometr. **2**, 358–378 (2010). https://doi.org/10.1504/IJBM.2010.035450

24. Sandri, M., Zuccolotto, P.: Variable selection using random forests (2006). https://doi.org/10.1007/3-540-35978-8_30

25. Alotaibi, Y., Meftah, A.: Review of distinctive phonetic features and the Arabic share in related modern research. Turk. J. Electr. Eng. Comput. Sci. **21**, 1426–1439 (2013). https://doi.org/10.3906/elk-1112-29

Face Recognition Based on Harris Detector and Convolutional Neural Networks

Assaad Oussama Zeghina[(✉)] , Oussama Zoubia , and Ali Behloul

LaSTIC Laboratory, Department of Computer Science, Batna-2 University, Batna, Algeria
assaad.o.zeghina@gmail.com, zoubia.oussama@hotmail.com,
a.behloul@univ-batna2.dz

Abstract. Facial recognition has always been a field of continuous development and research due to its usage in different areas such as security and robotics. It has gained even more popularity and interest by the researchers with the recent advancements in artificial intelligence and deep learning, which improved the robustness of facial recognition systems. In this paper, we focus on facial recognition using deep learning on small data sets with a limited number of individuals, for that we propose a local features based facial recognition approach that combines the robustness of feature extraction of CNN with the Harris corner detector. The experimental results of our proposed method surpassed the results of classical methods (LBP, Eigen Face, and Fisher Face) as well as recent works on Georgia Tech Face Database and AR Face Database and proved its efficiency and its robustness in different conditions including illumination variation, face pose variation, changes in facial expressions and face occlusions.

Keywords: Facial recognition · Deep learning · Convolutional neural networks · Harris corners detector

1 Introduction

Facial recognition is a biometric identification process that consists of identifying a person based on digital images of his face. It is widely used in various fields such as security and robotics. The research on facial recognition can be traced back to the 1960s, and it is still continuous, it gained more popularity recently due to the advancements in computer hardware and artificial intelligence. Despite the achievements made in this area, it still faces major challenges caused by the variations that the images of an individual's face could contain, such as the variation in light, the face pose variation, the variation in age, changes in facial expressions and face occlusions [1].

The main steps of a facial recognition system are the face detection, the extraction and classification of facial features. The face detection is the process of identifying the location of a face within an image. Object detection algorithms such as the Viola Jones algorithm [2] are used for face detection, and recently deep learning based methods have also been used like Faster R-CNN and YOLO [3, 4]. The extraction and the classification

of facial features step consists of extracting the important facial features, which are used to distinguish between the individuals. Three approaches for feature extraction can be used: global approaches which consider the entirety of the face images to extract global features, local approaches extract local face features, the last approach is the hybrid approach which is a combination of the two approaches [5].

Classical facial recognition methods use simple handcrafted features to describe the content of the image, machine learning algorithms are then employed for the classification. Recent years have seen a great development in deep learning which spread its uses to facial recognition. The convolutional neural networks (CNN) are the most used type of neural networks in facial recognition.

In this paper, a facial recognition method based on the detection of regions of interest and the convolutional neural network is proposed. The main focus of this method is to achieve high recognition rates on small data sets with limited number of subjects.

The paper is organized as follows: In Sect. 2, the state-of-the-art methods for facial recognition are explored, including classical methods, a brief overview of CNN and deep face recognition. In Sect. 3, our proposed deep learning based facial recognition method is presented. The description of the data sets as well as the results and discussions of the experiments are presented in Sect. 4. The last section is dedicated to the conclusion.

2   Related Work

Many researches were conducted to improve the robustness of the facial recognition methods. The classical methods were based on techniques such as edges and contours, Gabor filters [6] are an example of these methods, which have been successfully applied in many image processing tasks including face recognition. The Local Binary Pattern (LBP) [7] is another method that can be used for facial recognition, it is a powerful texture descriptor that is invariant against the change of illumination. Other variants of the LBP were proposed which achieved better recognition rates [8–10]. The Eigen face [11] and the Fisher face [12] are two other facial recognition methods which are based on dimensionality reduction algorithms, the Eigen face is based on PCA (principal component analysis) and the Fisher face is based on LDA (Linear Discriminant Analysis).

Recently, deep learning based methods particularly convolutional neural networks have gained much popularity in various fields, including facial recognition. A CNN architecture is composed of different types of layers: a convolution layer which is the core component of the CNN is used to extract the features from an image and return feature maps, it consists of a combination of convolution operations and activation functions. For each layer, the convolutions are calculated between the feature maps of the previous layer and a set of kernels whose weights are learned during the training, followed by an activation function applied on the resulting feature maps. The convolution layer is usually followed by a pooling layer. The pooling layer is used to reduce the dimensionality of the feature map and retain only the most important features. The input image is divided into a set of windows of the same size, each window is down-sampled by outputting its maximum or average value and discarding all the other values. The fully connected layers are a multilayer perceptron which takes a flattened vector from previous layers and outputs a class for the input image. The number of output nodes of the last fully

connected layer is the same as the number of classes [22]. In facial recognition, the convolution layers are used for the automatic features extraction from the face images, while the fully connected layers are used for the classification.

Deep face [13] was the first proposed CNN model for facial recognition, it was developed by Facebook AI research in 2014, mainly composed of nine layers, containing more than 120 million parameters. It achieved an accuracy of 97.35% on the LFW face data set when trained on the SFC (social face classification data set) which contains over 4.4 million images. FaceNet [14] is another CNN model that was developed in 2015, composed of 22 layers with more than 140 million parameters, trained on more than 200 million images, it achieved an impressive accuracy of 99.63% on the LFW data set. Another CNN model is the VGG-face [15] with 22 layers and more than 138 million parameters, it was trained on 2.6 million images, achieving an accuracy of 98.95% on the LFW data set. DeepID [16] is another popular CNN model for facial recognition reaching a high accuracy of 97.45% on the LFW data set when it was trained on 0.2 million images, this CNN model has more than 101 million parameters. These CNN models are complex and have high number of parameters, and they were trained on massive face data sets. A lot of work has been done on shallow CNN models with small number of parameters, which showed good results on fairly small data sets [17–19].

3   Proposed Method

Our method is composed of three main modules: the regions of interest extraction module, the convolutional neural network and finally the decision module (Fig. 1).

A points of interest detection algorithm [20] is used mainly for two reasons: firstly, because these algorithms are generally very fast and efficient. Secondly, the detected points of interest in a facial region in an image have a high probability of being detected in the same facial regions in other images of the same person (Fig. 2). For each image a maximum of 28 regions of size 32×32 pixels are extracted, the minimum distance between the centers of any two regions is set to 20 pixels.
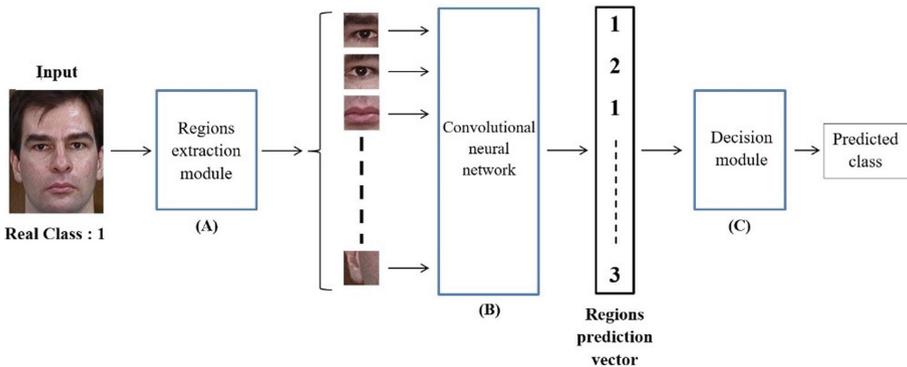

Fig. 1. The architecture of the proposed method.
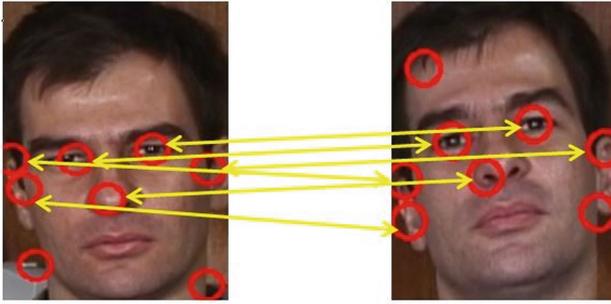
Fig. 2. Points of interest detected in a region are detected in the same regions in other images of the same person, 5 out of 8 points of interest detected in both images appear in similar regions.

The regions of interest are passed to CNN which returns a class for each region. The choice to use a CNN as a classifier is due to its efficiency in image classification tasks, particularly in facial recognition [13–19]. In case of small databases, a shallow neural network may achieve slightly higher recognition rates than a large neural network [32], for that reason we chose a shallow CNN model composed of 10 layers with 4 blocks of convolution and pooling layers followed by a fully connected layer of 512 nodes, and finally a softmax classifier. All blocs contain a batch normalization layer between the convolution and the pooling layers except for the first block. The batch normalization layers were used to prevent overfitting. We used filters of shape 3×3 for all the convolution layers, the first convolution layer employs 32 filters, where the second and the third contain 64 filters each, the last convolution layer contains 128 filters as shown in Fig. 3. The model was trained over 25 epochs using Adam optimizer and categorical cross entropy as a loss function. The number of layers and parameters as well as the number of epochs were chosen using a grid search technique.
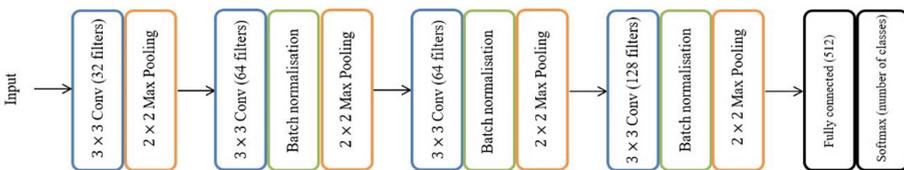
Fig. 3. The proposed CNN architecture.

The CNN outputs of each of the regions are assembled into a "regions prediction vector", and each element of this vector corresponds to the predicted class of a region by the CNN, the decision module takes this vector and returns the class with the most occurrences, which is the predicted class of the original input image.

4 Experiments and Results

In order to evaluate the results of our proposed method, we compared its recognition rate to those obtained with LBP, Eigen face, Fisher face, a CNN model similar to the one used in our method as well as the results of recent works on Georgia Tech Face Database and AR Face Database.

Fig. 4. Sample images of two subjects (male and female) from the Georgia Tech Face Database.

The first data set is Georgia Tech Face Database[1] which contains images of 50 people taken in sessions between 06/01/99 and 15/11/99 at different times at the Georgia Institute of Technology's Image and Signal Processing Center. Each individual in the database is represented by 15 color JPEG images, and the images of each subject are taken under different conditions, such as variation in exposure, variation in brightness, different facial expressions (as shown in the Fig. 4). The average size of the faces of these images is 150 × 150 pixels. We used a k-fold cross validation to divide this data set into training and testing where 80% of the images are used for training, and 20% are used for testing.

(A)

(B)   (C)
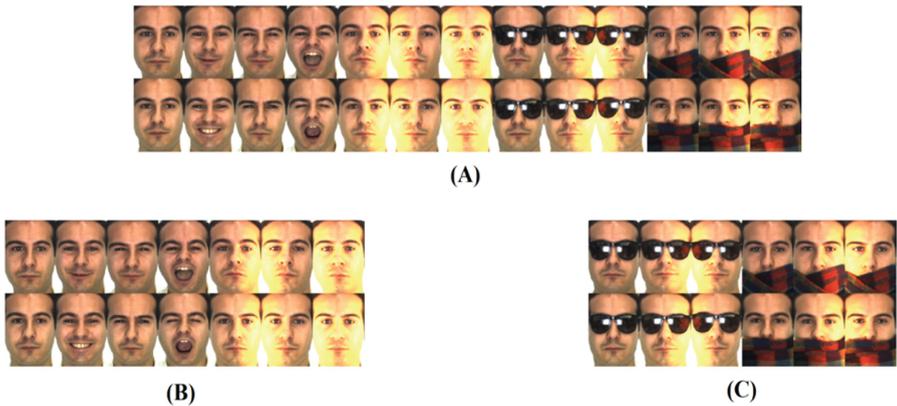
Fig. 5. (A) represents all the images of an individual (26 images), the images are divided into 2 sets, (B) which contains only the fully visible face images used for training, (C) which contains the face images with occlusions used for testing.

---

[1] Georgia Tech Face Database, https://www.anefian.com/research/face_reco.htm.

The second data set is AR Face Database [21] which contains more than 4.000 color images corresponding to 126 faces of people (70 men and 56 women). Images show frontal faces with different facial expressions, lighting conditions and occlusions (sunglasses and scarves). In this work, we selected 100 subjects (50 men and 50 women) where each subject has 26 images. This data set is used mainly to test the robustness of our method against occlusions. For the training subset we chose to include only the images of fully visible faces where each subject has 14 images (Fig. 5. B), the rest of the images, which are images of individuals with glasses and scarves are used for testing (Fig. 5. C).

Table 1. The average recognition rates of classical methods and our method on the Georgia Tech Face Database.

| Method	LBP	Eigen face	Fisher face	CNN	Our method
Fold 1	77.33%	76.66%	86.66%	95.33%	95.33%
Fold 2	72.66%	77.33%	84.66%	95.33%	94.66%
Fold 3	74.66%	80%	87.33%	94%	99.55%
Fold 4	74.66%	87.33%	86%	95.66%	98%
Fold 5	74.66%	84%	85.33%	95.33%	99.55%
Average	74.79%	81.06%	85.99%	95.13%	**97.41%**

Table 2. The average recognition rates of recent works and our method on the Georgia Tech Face Database.

Method	ILRVCS [23]	sgFKNN [24]	CNN [18]	CG-RF [25]	SAPFR [26]	Our method
Recognition rate	90%	79.57%	94.8%	95.1%	95.75%	**97.41%**

Table 1 shows the performance of different methods, as well as our method, we can notice that the LBP achieved the lowest recognition rate compared to other methods. Eigen Face performed slightly better than LBP, surpassed by the Fisher Face. CNN achieved a better recognition rate compared to previous methods, which proves its robustness against different variations in the face images. Compared to recent works on this data set, our method surpassed them as shown in Table 2.

Table 3. The recognition rates of classical methods and our method on the AR Face Database.

Method	LBP	Eigen face	Fisher face	CNN	Our method
Recognition rate	24.83%	23.66%	51%	16.41%	**94.72%**

Table 4. The average recognition rates of recent works and our method on the AR Face Database.

Method	OA-LGBPHS [27]	SSRC1 [28]	VGG-Face [29]	DLSL [30]	SR-NMR [31]	Our method
Recognition rate	83.54%	92.99%	59.09%	89.11%	86.35%	**94.72%**

The AR Face Database is the database on which our proposed method shows its potential and its robustness against occlusions. Table 3 and Table 4 present the recognition rates of classical methods and recent works on this data set, which used the same approach that we used to divide the data set into training and testing as well as the recognition rate of our method. We can observe that our method surpassed both the classical methods and the recent works, which confirms its effectiveness against face occlusions.
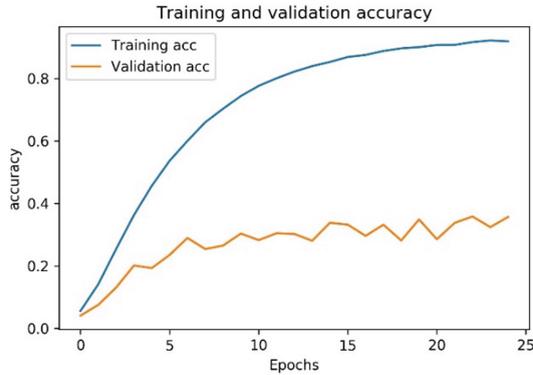
Fig. 6. History of training and testing accuracy of the CNN of our proposed method on the AR Face Database.

In Fig. 6, we notice that the testing accuracy of the CNN (the classification of the regions of interest) is very low (around 30%) but the face image recognition achieves high rates. The low accuracy of the CNN is caused by the misclassification of the occluded regions of the faces, since these regions have low similarity to the regions used for training, their classification is inconsistent which reduces their effect as noise.

5   Conclusion

The integration of deep learning into facial recognition has already proven to show an improvement in the recognition rates of facial recognition systems. In this paper, we propose a facial recognition method based on shallow convolutional neural networks and Harris corner detection algorithm, which showed great performance when tested on Georgia Tech Face Database and AR Face Database, and proved its robustness against pose variation, illumination variation, change in facial expressions and particularly face

occlusion. In addition, we obtained better results than state-of-the art methods. The proposed facial recognition approach can be useful in recognizing facial identity even with facial occlusion, and could be extended to explore larger databases.

References

1. Malikovich, K.M., Ugli, I.S.Z., O'ktamovna, D.L.: Problems in face recognition systems and their solving ways, In: International Conference on Information Science and Communications Technologies (ICISCT), Tashkent, pp. 1–4 (2017)
2. Wang, Y.Q.: An analysis of the Viola-Jones face detection algorithm. Image Process. On Line **4**, 128–148 (2014)
3. Jiang, H., Learned-Miller, E.: Face detection with the faster R-CNN. In: 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017), Washington, DC, pp. 650–657 (2017)
4. Yang, W., Jiachun, Z.: Real-time face detection based on YOLO. In: 1st IEEE International Conference on Knowledge Innovation and Invention (ICKII), Jeju, pp. 221–224 (2018)
5. Parisa Beham, M., Mohamed Mansoor Roomi, S.: A review of face recognition methods. Int. J. Pattern Recogn. Artif. Intell. **4**(27), 1356005 (2013)
6. Bhuiyan, A.A., Liu, C.H.: On face recognition using Gabor filters. World Acad. Sci. Eng. Technol. **28**, 51–56 (2007)
7. Ahonen, T., Hadid, A., Pietikainen, M.: Face description with local binary patterns: application to face recognition. IEEE Trans. Pattern Anal. Mach. Intell. **28**, 2037–2041 (2006)
8. Tan, X., Triggs, B.: Enhanced local texture feature sets for face recognition under difficult lighting conditions. IEEE Trans. Image Process. **19**(6), 1635–1650 (2010)
9. Hasanul Kabir, Md., Ahmed, F.: Face recognition with directional ternary pattern (DTP). In: International Conference on Graphic and Image Processing (ICGIP 2012) (2013)
10. Yang, W., Wang, Z., Zhang, B.: Face recognition using adaptive local ternary patterns method. Neurocomputing **213**, 183–190 (2016)
11. Slavković, M., Jevtić, D.: Face recognition using eigenface approach. Serb. J. Electr. Eng. **9**, 121–130 (2012)
12. Anggo, M., Arapu, L.: Face recognition using fisherface method. J. Phys. Conf. Ser. **1028**, 012119 (2018)
13. Taigman, Y., Yang, M., Ranzato, M., Wolf, L.: DeepFace: closing the gap to human-level performance in face verification. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2014)
14. Schroff, F., Kalenichenko, D., Philbin, J.: FaceNet: a unified embedding for face recognition and clustering. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015)
15. Parkhi, O.M., Vedaldi, A., Zisserman, A., et al.: Deep face recognition. In: BMVC, vol. 1, p. 6 (2015)
16. Sun, Y., Wang, X., Tang, X.: Deep learning face representation from predicting 10,000 classes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1891–1898 (2014)
17. Yan, Y., Li, C., Lu, Y., Zhou, F., Fan, Y., Liu, M.: Design and experiment of facial expression recognition method based on LBP and CNN. In 14th IEEE Conference on Industrial Electronics and Applications (ICIEA), Xi'an, China, pp. 602–607 (2019)
18. Coşkun, M., Uçar, A., Yildirim, Ö., Demir, Y.: Face recognition based on convolutional neural network. In: 2017 International Conference on Modern Electrical and Energy Systems (MEES), pp. 376–379 (2017)

19. Chen, J., Zhang, Z., Yao, L., Li, B., Chen, T.: Face recognition using depth images base convolutional neural network. In International Conference on Computer, Information and Telecommunication Systems (CITS), Beijing, China, pp. 1−4 (2019)
20. Shi, J., Tomasi, C: Good features to track. In: IEEE Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA (1994)
21. Martinez, A.M., Benavente, R.: The AR Face Database. CVC Technical Report #24, June 1998
22. Yamashita, R., Nishio, M., Do, R.K.G., et al.: Convolutional neural networks: an overview and application in radiology. Insights Imaging **9**, 611–629 (2018)
23. Huang, W., Wang, X., Zhu, Y., Zheng, G.: Improved LRC based on combined virtual training samples for face recognition. Int. J. Pattern Recogn. Artif. Intell. **30**, 1656006 (2016)
24. Kasemsumran, P., Auephanwiriyakul, S., Theera-Umpon, N.: Face recognition using string grammar fuzzy K-nearest neighbor. In: 8th International Conference on Knowledge and Smart Technology (KST), Chiangmai, pp. 55–59 (2016)
25. Chark, S.Y., Noor, N.Mohd.: Integrating complete Gabor filter to the random forest classification algorithm for face recognition. J. Eng. Sci. Technol. **14**, 859–874 (2019)
26. Li, Z.M., Li, W.J., Wang, J.: Self-adapting patch strategies for face recognition. Int. J. Pattern Recogn. Artif. Intell. **34**, 2056002 (2019)
27. Rui, M., Hadid Abdenour, Dugelay Jean-Luc: Efficient detection of occlusion prior to robust face recognition. Sci. World J. **2014**, 519158 (2014)
28. Ou, W., You, X., Tao, D., Zhang, P., Tang, Y., Zhu, Z.: Robust face recognition via occlusion dictionary learning. Pattern Recogn. **47**, 1559–1572 (2014)
29. Ghazi, M.M., Ekenel, H.K.: A comprehensive analysis of deep learning based representation for face recognition. In IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 102–109 (2016)
30. Liao, M., Gu, X.: Face recognition based on dictionary learning and subspace learning. Digit. Signal Process. **90**, 110–124 (2019)
31. Chen, Z., Wu, X.J., Kittler, J.: A sparse regularized nuclear norm based matrix regression for face recognition with contiguous occlusion. Pattern Recogn. Lett. **125**, 494–499 (2019)
32. Peng, M., Wang, C.Y., Chen, T., Liu, G.Y.: NIRFaceNet: A convolutional neural network for near-infrared face identification. Information **7**, 61 (2016)

Softcomputing and Optimization

Quality Preserved Color Image Compression Using Particle Swarm Optimization Algorithm

Djamel Eddine Touil[1(✉)] and Nadjiba Terki[2]

[1] Energy Systems Modeling Laboratory, University of Biskra, Biskra, Algeria
tde.touil@gmail.com
[2] LESIA Laboratory of Research, Electrical Engineering Department,
University of Biskra, Biskra, Algeria
t_nadjiba@yahoo.fr

Abstract. In this paper, we propose an efficient discrete wavelet transform-based compression method for color images. Generally, the strong correlation exists between the three planes R, G, and B of a color image, where the decrease of this correlation gives an improvement in the compression quality. The proposed method utilizes an efficient technique to reduce this correlation efficiently. In this regard, the main contribution is to design an optimized color space $S_1S_2S_3$ using the PSO algorithm to represent the RGB image in a space more appropriate for performing the compression. The idea is to maximize the energy of the image in the plane S_1 more than in S_2 and S_3. Moreover, we propose to optimize the thresholds appropriate for each plane of the converted image to partially reduce the number of the less important DWT coefficients that correspond to the lower quantity of energy. The obtained results facing those of state-of-the-art methods confirm that the proposed method shows clearly that the proposed method achieves high performances.

Keywords: Compression · Color space · PSO · DWT

1 Introduction

Recently, many image processing applications have appeared due to the fast improvement done in the computer vision field. In this regard, image compression is one of the most active applications. All image compression works intend to improve an effective technique that removes the useless amount of the original image without losing its pertinence. Image compression deals with redundancy, the number of bits needed to represent an image by removing redundant data. Decreasing the redundancy is the main aim of the image compression algorithms [1,2].

In literature, image compression techniques are categorized inside two main groups, whether the original image can be recovered with fill mathematic

precision from the compressed image. Lossless or Lossy compression approaches can be applied to the hyperspectral image.

– Lossless compression: This group is envisioned for conscious applications such as medical, archiving, and technical subjects. The objective is to reconstruct the compressed data to reproduce ideally the original image.
– Lossy compression: It is dedicated to approximate the original image, where the loss of information is negligible. It results in a significant compression ratio than the lossless technique. However, it can hurt the reconstruction of the original data.

Build important transforms for the Lossless image compression area is very useful and this by using various color space transforms such as DCT and DWT algorithms. Each input image is divided into sub-image blocks. Then each block is transformed using DCT or DWT. The DCT coefficients of each block is arranged hierarchically [2]. Wavelet transforms mean representing the image data into approximation and details sub-images using a particular fixed-function called mother wavelet function. DCT is low-level image compression, where it compresses the image of lower performance.

The rest of this paper is ordered as follows. Section 2 provides the related work. In Sect. 3, we introduce in detail our proposed method. The fourth section presents and discusses the experimental results and finally, Sect. 5 provides the conclusion and future work.

2 Related Work

Among the proposed compression methods, the work presented by Jagadeesh *et al.* [3] can be classified as a lossy and lossless compression method. It presents greater compression ratios compared with other lossless coding methods such as the Lempel-Ziv-Welch (LZW) coding method. It exists also the Huffman coding technique, which is widely adopted through lossless image compression methods. This work aims to reduce the amount of code used to compress a BMP image. This method presents acceptable results for grayscale images, while it does not for color images. Moreover, the work proposed by Kaur *et al.* [4] aims to improve the significance of the compressed image by using the Retinex algorithm, which guarantees the similarity between the real and the reconstructed images.

Otherwise, lossy compression methods have been widely proposed especially transform-based methods. The idea is to represent the image information in a few coefficients such as cosine transform (CT) coefficients and wavelet transform (WT) coefficients. Many transform-based methods have recently been developed. In [5], Ohm *et al.* proposed the use of discrete cosine transform (DCT) to concentrate the information in a short sequence of coefficients. This feature made the DCT-based methods computationally efficient. In addition, several compression standards, such as the JPEG recognized the DCT as a basic technique. In [6] the authors proposed a lossy method for image compression. The idea is to apply the DCT on the YCbCr image obtained from the original RGB image. The bisection method is used to determine the adequate threshold based on a prefixed user

peak signal to noise ratio (UPSNR) as a controlled quality criterion. Then, they used the difference of the indexes of the retained coefficients in coordination with DCT block adaptive scanning to encode efficiently the DCT coefficients. The difference of the indexes is stored in a lookup table called (dLUT). Hassan *et al.* [7] proposed a color image compression scheme in RGB and YCbCr color models. 1-D DCT was used to transform the image planes to frequency domain instead of 2-D DCT. An adaptive scalar quantization step is applied. The zigzag scan was used to rearrange the data from nonzero in increased order, where the adaptive shift coding was applied that performs Differential Pulse Coding Modulation on the DC coefficients for the entire image and Run Length Encoding for the AC coefficients for the entire image.

In [8], the authors based on the conventional DCT compression principle and propose a Dynamic Energy Analysis Discrete Cosine Transform (DEADCT) image compression algorithm. The idea is to introduce a new concept of "compression radius" and partition the image, then adjusts and quantifies the compression radius dynamically according to the energy spectrum of different pixel blocks, which allow preserving more detailed image features while ensuring the compression effect.

The wavelet transform is considered as a primary method used for image compression. It gives the multi-resolution capability by decomposing the signal in time and scale. It improves image quality and provides higher compression ratios. In 2014, Rathee *et al.* [9] introduced a DWT-based method for image compression. They adopt the Haar mother wavelet to decompose the image into a sequence of wavelet coefficients, where the useless wavelet coefficients are thresholded. The resulted coefficients are then quantized for adjusting them into an integer sequence. Finally, entropy encoding is used to reduce its amount of bits quantized coefficients. In [10] Ghadah *et al.* introduced a hybrid inter-differentiation color compression system of RGB base to remove the redundancies efficiently. They used the polynomial coding and the hierarchical scheme of one level DWT of lossily base. The results shown are promising in terms of the higher compression gain achieved with preserving image quality, due to utilizes both the spatial and spectral contents simply and effectively. Mody *et al.* [12] proposed a graystage mapping approach for image compression using Discrete Wavelet Transform (DWT) technique and optimization using evolutionary algorithms like artificial bee colony and particle swarm optimization. The optimization provides the good quality of the image to provide better results. The drawback of this work is that the authors did not mention the database used in this work. Another DWT-based work is proposed in [11], where the authors aim to control the quality of the image compression in an optimized color space rather than using the conventional ones. The genetic algorithm is used to find three representation planes, in which the energy is almost maximized in the first plane than the two others.

Motivating by the aim of developing an efficient method for image compression, and unlike using GA in [11], we present in this paper an efficient method based on the DWT and the particle swarm optimization algorithm (PSO) for RGB images compression since the PSO algorithm proves its efficiency in the work [12].
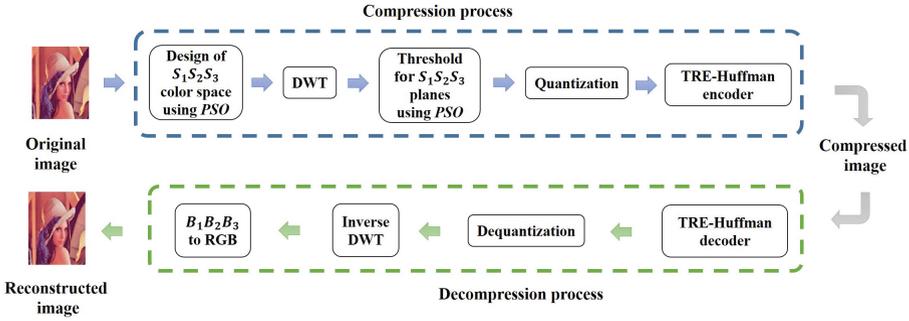
Fig. 1. The flowchart of the proposed method.

3 Proposed Method

We present in this section, the main steps of the proposed method.

– First, we design an optimized color space $S_1S_2S_3$ to give more efficiency in distributing the energy of the RGB image through the DWT, where the cost function is established to maximize the energy in the plane S_1 compared to S_2 and S_3.
– After converting the RGB image into space $S_1S_2S_3$, we produce optimized thresholds appropriate to each plane of the converted image. This step is maintained using the PSO algorithm, where the cost function is designed to minimize the bits per pixel without wasting the pertinence information after the image decompression step.
– We apply a lossless coding technique based on the TRE coding and the Huffman coding to improve the compression rate.

The method is decomposed into two main phases as shown in Fig. 1. The compression phase is detailed as follows.

3.1 Color Space Design

In a hand, this step is basically done due to the important redundancy in RGB color images, which can reduce the compression quality [13]. The purpose consists in decomposing the RGB image into three orthogonal planes $S_1S_2S_3$ in order to eliminate the correlation exhibited between the pixels of the RGB image. In this context, we adopt the PSO for energy separation between the RGB planes.

PSO Algorithm. Inspired from the social behavior of bird flocking, particle swarm optimization algorithm is the one most useful optimization algorithm due to its capability to solve non-linear, multimodal and high-dimensional optimization problems. Furthermore, the computation cost remains low.

The basic steps of PSO are summarized as a pseudo-code presented in Fig. 2 [14,15]. We implement the PSO algorithm's dynamic presented by Eqs. (4–5) to separate the high energy of the RGB image into the three planes S_1, S_2 and S_3, where the cost function f is established to maximize the energy in the first plane S_1 and the rest is distributed in the two planes S_2 and S_3 by optimizing nine parameters $a_{ij=1,2,3}$, i.e.,

$$f = max(E_1 - (E_2 + E_3)) \tag{1}$$

$$E_k = 100 \times \frac{\sum_{i=1}^{m} \sum_{j=1}^{n} (S_{kij})^2}{E_T}, \quad k = 1, 2, 3 \tag{2}$$

E_k represents the energy percentage for each kth plane. The percentage of the energy of each plane is defined by:

$$E_T = \sum_{i=1}^{m} \sum_{j=1}^{n} \left((S_{1ij})^2 + (S_{2ij})^2 + (S_{3ij})^2 \right) \tag{3}$$

$$\begin{aligned}
v_i(t+1) &= v_i(t) + Q_1 + Q_2, \quad i = 1, 2, ..., n \\
Q_1 &= c_1 \times rand \times (p_i^{best} - p_i(t)) \\
Q_2 &= c_2 \times rand \times (p_g^{best} - p_i(t))
\end{aligned} \tag{4}$$

$$p_i(t+1) = p_i(t) + v_i(t) \tag{5}$$

where $v_i(t+1)$ is the new velocity for the ith particle, n is the size of the population, c_1 and c_2 are the weights of the best position and best overall position respectively, $p_i(t)$ is the optimal position at instant t. p_g^{best} denotes the best position.

$$\begin{bmatrix} S_1 \\ S_2 \\ S_3 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{23} & a_{33} \end{bmatrix} \times \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{6}$$

where $a_{i,j=1,2,3}$ present the optimized solutions. The inverse transformation $S_1 S_2 S_3 - RGB$ is simply expressed by:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{23} & a_{33} \end{bmatrix}^{-1} \times \begin{bmatrix} S_1 \\ S_2 \\ S_3 \end{bmatrix} \tag{7}$$

The stop criterion chosen is defined as:

$$100 \times \frac{|PSNR - UPSNR|}{UPSNR} > \epsilon \tag{8}$$

where: ϵ is the convergence precision already predefined to be 1%, $UPSNR$ is the $PSNR$ set in advance by the user and $PSNR^*$ refers to the $PSNR$ that corresponds to the best thresholds. The PSO parameters are summarized in Table 1, where, the parameters are chosen empirically.

Begin

    For each particle in the swarm
        Initialize its position & velocity randomly
    end for
do
    For each particle in the swarm
        Evaluate the fitness function
        If the objective fitness value is better than the personal best objective
        fitness value (P_{best}) in history, current fitness value set as the new personal
        best (P_{best})
        end if
    end for

    From all particles or neighborhood, choose the particle with the
    best fitness value as the G_{best}

    for each particle in the swarm
        Update the particle velocity
        Update the particle position
    end for

    Until stopping criteria is satisfied

end begin

Fig. 2. Pseudo-code of the PSO algorithm.

Table 1. The parameters of the PSO algorithm.

Parameter	Value
c_1	1.45
c_2	1.2
n	20

3.2 Discrete Wavelet Transform (DWT)

In signal processing, 1-D DWT consists of a low (L) and high (H) pass filters splitting a line of pixels into two lines of half the size. For image processing, the DWT is employed separately on each image component to decorrelate the image into distinct level sizes, saving much of its spatial correlation, where it produces four subbands (LL, LH, HL, and HH), in which the LL subband is a lower resolution representation of the original image, and the missing details are filtered into the remaining subbands [16]. The subbands contain the horizontal (LH), vertical (HL), and diagonal (HH) edges on the scale size defined by the wavelet [11]. To achieve efficient compression, we apply the pyramidal digital wavelet transform until the fifth level. This choice was made based on several tests. We note that we used mother wave bior 4.4.

3.3   Thresholds Optimization

The DWT coefficients are classified into two groups. The nonzero coefficients NZ_i and the zero coefficients Z_i, which signify the coefficients less than or equal to the optimized threshold $T_{i=1,2,3}$. Each threshold T_i is optimized corresponding to each image plane S_i. These thresholds are obtained from the second utilization of the dynamic PSO algorithm shown in Fig. 2 for the aim to eliminate the ineffective DWT coefficients, in which they correspond to the PSO solutions. Thus, each particle searches for the three solutions T_1, T_2, T_3. We chose to maintain the image quality by minimizing the cost function given by the Eq. (11). The stop criterion chosen is defined as:

$$100 \times \frac{|bpp - Ubpp|}{Ubpp} > \epsilon \tag{9}$$

where: ϵ is the convergence precision already predefined to be 1%.

3.4   Quantization

Similar to [13], the NZ coefficients are quantized linearly with a resolution of Q bits according to the following equation:

$$QNZ = \left\lfloor \left(\frac{(NZ - NZ_{min})}{(NZ_{max} - NZ_{min})} \right) \times \left(2^Q - 2 \right) + 1 \right\rfloor \tag{10}$$

3.5   Lossless Encoder

We propose an efficient encoder which joins two main techniques. First, the compression process performs the coding of the quantized coefficients by the TRE coder. As its name suggests, the two-roles encoder (TRE) has two roles [17]. The thresholding effect of the DWT coefficients vector gives a significant number of sequences of zeros. Then, each sequence is replaced by a code of $(Q+1)$ bit. The minimum value is replaced by 1 (the null value is impossible). Second, the run-length-encoder (RLE) is used to compress the values greater than 2^Q. Then, the resulted coded vector is compressed using the Huffman encoder.

3.6   Performance Criteria

To evaluate the performance of the proposed method, the used criteria are defined as:

- **Bits per pixel:** This shows the relationship between the data amount of the compressed image and the original image.

$$bpp = \frac{\text{size of compressed image in bits}}{\text{number of pixels}} \tag{11}$$

- **Peak signal to noise ratio:** It measures of distortion used in a digital image. This measure is given as a ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation.

$$PSNR = 100 \times log_{10} \left(\frac{255^2 \times 3}{MSE(R) + MSE(G) + MSE(B)} \right) \qquad (12)$$

$$MSE = \frac{1}{N \times M} \times \sum_{i=1}^{N} \sum_{j=1}^{M} ((I(i,j) - I^*(i,j))^2 \qquad (13)$$

where I and I^* refer to the original and reconstructed image plane. N and M are the image dimensions. In the two phases of color space design and threshold optimization, the solutions obtained are always ensured within the limits according to the stopping criterion represented by Eq. 9.

Decompression is just the inverse process of compression as indicated in Fig. 1.

Table 2. Energy distribution in RGB and $S_1 S_2 S_3$ spaces.

Image	RGB			$S_1 S_2 S_3$		
	E_R	E_G	E_B	E_{S_1}	E_{S_2}	E_{S_3}
Aireplane	31.90	32.67	35.43	99.21	0.45	0.32
Peppers	49.11	38.08	12.81	85.57	5.54	8.87
Lena	58.37	21.09	20.53	90.33	1.39	8.26
Girl	32.36	32.93	34.71	99.37	0.14	0.47
Couple	46.76	28.89	24.35	92.66	2.28	5.03
House	33.35	30.90	35.74	96.34	1.08	2.57
Zelda	48.26	28.70	22.98	90.58	2.64	6.76
Average	42.88	30.47	26.65	**93.44**	**1.93**	**4.61**

4   Results and Performance Comparison

In order to evaluate and test the efficiency of the proposed method, two experiments are performed. In the first one, seven commonly used test color images in the literature are employed including Airplane, Peppers, Lena of size 512×512 for each one and Girl, Couple, and House of size 256×256 for each one [18]. In the second experiment, the image databases, Kodak lossless true-color image [19] is adopted.

Regarding the first experiment, the results reported in Table 2; show the efficiency in the performance of our approach. The energy distribution obtained according to the designed color space $S_1 S_2 S_3$ is particularly efficient compared

(a)

(b)
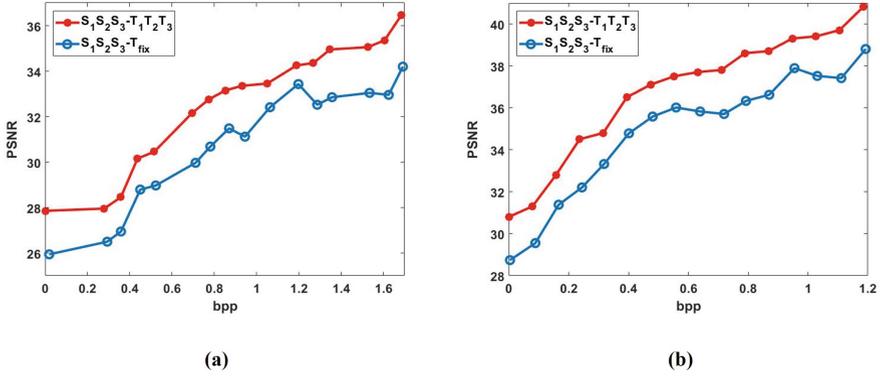
Fig. 3. Obtained performances in the $S_1S_2S_3$ color space for the optimized threshold $T_1T_2T_3$ and a fixed one obtained using bisection method for Lena and Girl color images: (a) Lena, (b) Girl

Table 3. Performances in the RGB and $S_1S_2S_3$ over the test images for quantizer width values of 7, 8 and 9.

Images	RGB						$S_1S_2S_3$					
	7 bits		8 bits		9 bits		7 bits		8 bits		9 bits	
	PSNR	bpp	PSNR	bpp	PSNR	bpp	PSNR	bpp	PSNR	bpp	PSNR	bpp
Aireplane	30.99	0.74	31.30	0.81	31.60	0.93	30.86	0.48	31.54	0.53	33.62	0.83
Peppers	31.00	0.75	30.96	0.88	31.06	0.86	31.19	0.92	30.91	0.74	30.96	0.90
Lena	32.94	0.99	32.94	1.09	33.09	1.12	32.55	0.73	33.03	0.77	32.75	0.82
Girl	35.84	0.81	35.57	0.87	35.97	1.01	36.73	0.64	36.21	0.51	35.81	0.76
Couple	33.66	1.41	32.96	1.44	33.96	1.63	33.03	0.7634	32.68	0.72	32.47	0.94
House	32.36	1.05	32.26	1.16	32.26	1.27	32.06	0.7434	32.07	0.71	32.75	0.96
Zelda	31.76	1.12	31.66	1.20	31.66	1.37	31.65	0.7834	32.10	0.80	32.25	0.98
Average	32.65	0.98	32.52	1.06	32.80	1.17	32.58	0.72	32.65	0.68	32.95	0.88

to the energy distributions in RGB; almost all of the energy is concentrated (93.44%) in the plane S_1.

Figure 3 demonstrates the effectiveness of the proposed optimized thresholds $T_1T_2T_3$ compared to the use of a fixed threshold obtained by the bissection method for removing the useless DWT coefficients. We have used the PSNR-bpp values obtained for the two cases of thresholding by PSO and thresholding by the dichotomy method for different PSNRs, these values are obtained by applying the conversion $RGB - S_1S_2S_3$ on the two images Lena and Girl. Therefore these results confirm that the bpp and the PSNR obtained by the $T_1T_2T_3$ thresholds are better than those of the threshold obtained from the bisection method. The experiments are extended to determine the PSNR according to the number of bits in pixel (bpp) using the conversion in $S_1S_2S_3$ for different values of Q. The results obtained are summarized in Tables 3. According to these results, it

(a) Aireplane

PSNR = 31.54
bpp = 0.53

(b) Peppers

PSNR = 30.91
bpp = 0.74

(c) Lena

PSNR = 33.03
bpp = 0.77

(d) Girl

PSNR = 36.21
bpp = 0.51

(e) Couple
PSNR = 32.68
bpp = 0.72

(f) House
PSNR = 32.07
bpp = 0.71

(g) Zelda
PSNR = 32.10
bpp = 0.80
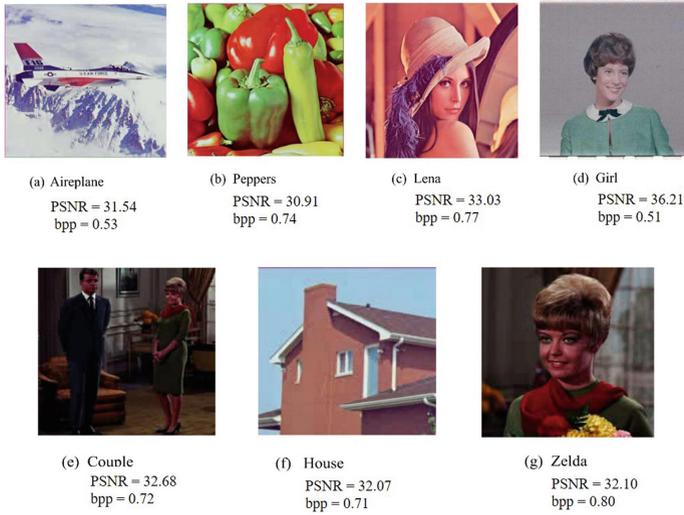
Fig. 4. The reconstructed images.

is obvious to observe the efficiency of the color space $S_1S_2S_3$ in terms of high performance. This performance obtained allowed us to perform compression as efficiently as possible without losing the quality of compressed images as shown in Fig. 4, which illustrates quantitative results for the quantifier width $Q = 8$.

Table 4 provides a clear comparison of the proposed method with the recent and state-of-the-art methods including, DCT-LUT [6], GA-DWT [11], CDABS [13] for the different image quality metrics used in these works. It is clear that the proposed method outperforms other methods.

To give more credibility to our proposed method, we summarize in Table 5 another results comparison with two other recent methods. It is worthy to note that our proposed method gives the best results compared with those methods in terms of high values of PSNR and low values of bpp.

In the second experiment, we extend the performance evaluation of the proposed method to SSIM and FSIM quality metrics [20]. Furthermore, the set of the test images is enlarged to the Kodak lossless true color image suite. Table 6 illustrates the performance of the proposed method for the different image quality metrics used over the Kodak image database. It is clear to remark, for the bitrates less than 0.3 bpp, our method gives acceptable PSNR values but more important in terms of SSIM and FSIM metrics. For the other values of bpp, the values are significantly high in terms of PSNR, SSIM, and FSIM metrics. These values allow us to see the efficiency of our method.

Table 4. Comparison of results with the state-of-the-art methods.

Image	Proposed method		DCT-LUT [6]		GA-DWT [11]		CDABS [13]	
	PSNR	bpp	PSNR	bpp	PSNR	bpp	PSNR	bpp
Aireplane	31.54	0.53	31.16	0.49	30.38	0.59	30.36	1.04
Peppers	30.91	0.74	31.20	0.83	30.05	0.80	30.15	1.50
Lena	33.03	0.77	32.76	0.66	31.97	0.81	31.93	1.17
Girl	36.21	0.51	35.90	0.41	35.00	0.45	35.13	0.60
Couple	32.68	0.72	32.87	0.89	32.28	0.92	32.44	1.00
House	32.07	0.71	32.10	0.83	31.72	0.82	31.79	1.20
Zelda	32.10	0.80	31.98	0.76	31.33	0.87	31.31	1.12
Average	**32.65**	**0.68**	32.54	0.70	32.57	0.70	32.56	0.89

Table 5. Comparison of results with the recent methods.

Image	Proposed method		AHCID [10]		DVT-DPCM-ASC [7]	
	PSNR	bpp	PSNR	bpp	PSNR	bpp
Peppers	30.91	0.74	–	–	29.29	2.13
Lena	33.03	0.77	32.59	1.09	32.22	1.875
Girl	36.21	0.51	36.33	1.19	–	–
House	32.07	0.71	33.67	0.89	–	–

Table 6. Performances of the proposed method over the Kodak image database.

bpp	PSNR	SSIM	FSIM
0.10	25.91	0.8145	0.8834
0.20	29.99	0.8280	0.8569
0.30	31.87	0.9115	0.8768
0.40	33.02	0.9000	0.9315
0.50	34.05	0.9439	0.9040
0.60	34.98	0.8954	0.9718
0.70	35.11	0.9481	0.9457
0.80	35.98	0.9244	0.8966
0.90	35.07	0.9041	0.9093

5 Conclusion

In this paper, we have proposed an image compression technique based on the DWT and the PSO algorithm. The efficiency of this technique is interpreted due to three main points. First, the proposed color conversion is more conducive to compression. The convenience of this color space is derived from the use of PSO to maximize the energy in the first color plane. Second, we produce

optimized thresholds suitable for each plane of the image previously converted basing on the PSO algorithm. Third, the application of an efficient encoder using the TRE-Huffman encoder improved the compression rate. The comparison results presented clearly show that our approach is entirely effective compared to the reference compression methods. Our future work will focus in a hand on improving the optimization of thresholding and color design steps by proposing using a recent bio-inspired algorithm and on the other hand improving the encoding step by proposing an efficient encoder.

References

1. Surabhi, N., Unnithan, S.N.: Image compression techniques: a review. Int. J. Eng. Devel. Res. **5**(1), 585–589 (2017)
2. Joshua, T.P., Arrivukannamma, M., Sathiaseelan, J.: Comparison of DCT and DWT image compression. Int. J. Comput. Sci. Mob. Comput. **5**(4), 62–67 (2016)
3. Jagadeesh, B., Ankitha, R.: An approach for image compression using adaptive Huffman coding. Int. J. Eng. Technol. **2**(12), 3216–3224 (2013)
4. Kaur, D., Kaur, K.: Huffman based LZW lossless image compression using retinex algorithm. Int. J. Adv. Res. Comput. Commun. Eng. **2**(8), 3145–3151 (2013)
5. Ohm, J., Sullivan, H., Schwarz, T.K.T., Wiegand, T.: Comparison of the coding efficiency of video coding standardsincluding high efficiency video coding (HEVC). IEEE Trans. Circuits Syst. Video Technol. **22**(12), 1669–1684 (2012)
6. Messaoudi, A., Srairi, K.: Colour image compression algorithm based on the DCT transform using difference lookup table. Electron. Lett. **52**(20), 1685–1686 (2016)
7. Hassan, E.K., George, L.E., Mohammed, L.E.: Color image compression based on DCT, differential pulse coding modulation, and adaptive shift coding. J. Theoret. Appl. Inf. Technol. **96**(11), 3160–3171 (2018)
8. Zhao, C., Tong, C.: Research on DCT image compression algorithm based on dynamic energy analysis. In: Proceedings of the International Conference on Artificial Intelligence and Advanced Manufacturing 2019, pp. 1–5 (2019). https://doi.org/10.1145/3358331.3358391
9. Rathee, M., Vij, A., Scholar, T.: Image compression using discrete haar wavelet transforms. Int. J. Eng. Innov. Technol. (IJEIT) **3**(12), 47–51 (2014)
10. Al-Khafaji, G., Al-Kazaz, H.B.: Adaptive color image compression of hybrid coding and inter-differentiation based techniques. Int. J. Comput. Sci. Mob. Comput. **8**(11), 65–70 (2019)
11. Boucetta, A., Melkemi, K. E.: DWT based-approach for color image compression using genetic algorithm. In: International Conference on Image and Signal Processing 2012. LNCS, vol. 7340, pp. 476–484 Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-31254-0_54
12. Mody, D., Prajapati, P., Thaker, P., Shah, N.: Image compression using DWT and optimization using evolutionary algorithm. In: Proceedings of the 3rd International Conference on Advances in Science & Technology (ICAST) (2020). https://doi.org/10.2139/ssrn.3568590
13. Douak, F., Benzid, R., Benoudjit, N.: Color image compression algorithm based on the DCT transform combined to an adaptive block scanning. AEU Int. J. Electron. Commun. **65**(1), 16–26 (2011)

14. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of ICNN 1995-International Conference on Neural Networks 1995, pp. 1942–1948. IEEE (1995). https://doi.org/10.1109/ICNN.1995.488968
15. Brownlee, J.: Clever Algorithms: Nature-Inspired Programming Recipes. 1st edn. LuLu.com (2011)
16. Javaid, R., Besar, R., Abas, F.S.: Performance evaluation of percent root mean square difference for ECG signals compression. Sig. Process. Int. J. (SPIJ) **2**(2), 1–9 (2008)
17. Benzid, R., Marir, F., Bouguechal, N.-E.: Electrocardiogram compression method based on the adaptive wavelet coefficients quantization combined to a modified two-role encoder. IEEE Sig. Process. Lett. **14**(6), 373–376 (2007)
18. USC-SIPI image database Homepage. http://sipi.usc.edu/database. Accessed 2 April 2020
19. Kodak lossless true color image suite Homepage. http://www.r0k.us/graphics/kodak. Accessed 14 July 2020
20. Sara, U., Akter, M., Uddin, M.S.: Image quality assessment through FSIM, SSIM, MSE and PSNR—a comparative study. J. Comput. Commun. **7**(3), 8–18 (2019)

A Simple Yet Effective Convolutional Neural Network Model to Classify Facial Expressions

Meriem Sari[1]([✉]) [iD], Abdelouahab Moussaoui[1], and Abdenour Hadid[2]

[1] Department of Computer Science, University of Ferhat Abbas Setif1, Setif, Algeria
{meriem.sari,abdelouahab.moussaoui}@univ-setif.dz
[2] Center for Machine Learning and Signal Analysis, University of Oulu, Oulu, Finland
hadid.abdenour@oulu.fi

Abstract. Facial Expression Recognition (FER) is a form of nonverbal communication; it translates the internal and emotional state of a human being by changing one or several facial muscles. Automated classification of facial expressions has known a great progress over the last decade; we observed the appearance of new methods based on Deep Learning (DL) instead of traditional classification methods. In this paper we propose an improved method based on Convolutional Neural Networks (CNN) that responds to the problem of classification of the six basic emotions(anger, disgust, fear, happy, sad and surprise) plus the neutral case. We validated our model on three public databases and we achieved better results than the state-of-the-art: CK+ 88,23%, JAFFE 86.24%, KDEF 82.38%. Our accuracies out perform results from recently proposed traditional methods as well as DL based methods.

Keywords: Facial expression recognition · Emotion recognition · Machine learning · Deep learning · Convolutional neural network

1 Introduction

Facial Expression is one of the most important aspects that can translate the emotional state of human beings since it has a high level of directness. It is defined as a set of facial muscles movements in response to a persons̀ internal and mental condition. Facial expression recognition (FER) have shown to play a key role in the emotion recognition process and has drawn a significant attention expressed by the great number of researchs works that have been conducted and the improved and interesting results that they achieved.

FER had known a large number of applications thanks to its efficiency such as: Human Machine Interaction (HCI) (interactive gaming, virtual reality, robotics etc.), the medical domain (autism [10], mental disorder [24], pain assessment [18] etc.), also used in surveillance [21] and health care systems.

By definition, automatic facial expression recognition is the process of automatically recognizing and classifying emotions using purely machines. The general classic process is conducted through different phases; it starts with face acquisition where faces are detected from still images or video sequences which are subsequently fed up to the detection system, then feature extraction and representation techniques are applied to the latter face images in an attempt to extract distinctive information represented by a feature vector. In fact, these features could be geometricbased (shape and location of facial components) [7,9,15,22,25,29] or appearancebased (texture and skin changes while exercising an expression) [1,11,13,14,28]. The final phase in which facial expression classification takes place, at this stage, most discriminative features are selected to eventually achieve a better classification of emotions using different classification techniques (SVM [30], Bayes Classifier [3], Fuzzy Techniques [2], Neural Networks [12] etc.). There are plenty of emotions in the wild that human beings can perform but in research the majority of researchers take in consideration the six basic emotions introduced by Paul Ekman in 1971 [4] which are: anger, fear, disgust, happiness, sadness, surprise and the neutral emotion that is used to differentiate between them all; these emotions became universal among human beings because they all perform them in the same way despite the difference in their race, gender or age.

In this research we are interested in classifying emotions using machine learning techniques and more precisely Deep Learning techniques in response to overcoming the limitations of classic methods (accuracy and efficiency) [26,27]. Deep Neural Networks (DNN) has the same structure as regular NN the only difference is that they have more hidden layers instead of just one. One of the very successful architectures in the field of classification is Convolutional Neural Networks (CNN) firstly introduced in 1997 [16]; they are based on the alternative use of convolutional layers responsible of generating image maps and subsampling layers (pooling layers) to reduce the maps size and a fully connected layer connected directly to the output layer. Its advantage is that it offers the possibility of using raw images as input instead of using handcoded features; this gives the ability to the model to extract more abstract features that could be very helpful.

The aim of this research is to develop a facial expression recognition system that classifies images into the seven basic emotions, this system is based on a simple yet effective CNN architecture combined with few steps of image preprocessing. We used images collected from various public databases namely: CK+, JAFFE and KDEF which are widely used in this field. We explored the accuracy of the model over many validations depending on the used dataset also we discussed the overall accuracy of the model over each dataset compared to recent related works and we presented the recognition rate of each emotion alone using confusion matrix and precision.

The rest of this paper is organized as next: Sect. 2 presents recently related works in this field, Sect. 3 describes our system and the different steps to implement it; Sect. 4 discusses the different results obtained in the validation step and at the end a conclusion.

2   Related Works

The research presented in [23] developed a system that is composed of four modules: input module, preprocessing, recognition and output module. It was tested on two databases JAFFE an CK+ and it was compared to a k nearest neighbour (KNN) algorithm, it achieved respectively 76,7442% and 80,303%. The system is constructed by implementing face detection using Haar like features in OpenCV and histogram equalization. The system architecture is composed of four layers; (two convolutional and two sub sampling) then a Softmax classifier is used for multiclassification. The authors favoured Haar like features in order to capture the useful portion of facial expression and removing most of the background information then applied Histogram Equalization in order to make gray scale values and contrast more uniform in all images, at the end they concluded that CNN model is more effective than classification based on KNN.

The work presented in [5] represents a traditional method for classification of facial expressions based on the use of histogram of oriented gradient (HOG) combined with support vector machine (SVM), authors applied face detection and cropping as preprocessing steps then added feature extraction function based on HOG to preserve local information of each facial image; the extracted HOGs are then concatenated into histogram bin to form one feature vecto fed to an SVM classifier. They tested their model on JAFFE and KDEF databases and achieved accuracies of 76.19% and 80.95% respectively, it can be observed that the emotion Fear scores the lowest recognition rate while Surprise scores the highest rate.

Another paper presented recently in [6] addresses the problem of facial expression classification using convolutional neural networks; authors applied face detection, image normalization and contrast enhancement preprocessing on the input images which are collected from JAFFE and KDEF databases, their architecture consists of five convolutional layers, One max pooling layer, two average pooling layers and three fully connected layers. The model was trained on a combined dataset where 90% of data is used for training while only 10% is used for test, bach size set to 100 and number of epochs to 100; their model achieved 70% on 100 epochs and took 200 epochs to achieve 72% and required 300 epochs to achieve a maximum accuracy of 78%.

A method presented by [19] studied the effect of face segmentation on the accuracy; first of all faces are detected using ViolaJones Haar cascaders then face region is cropped and resized, the image is then cropped to the specific region that is being tested (full face, half face, eyes, single eye, mouth and half of the mouth). They proposed a convolutional neural network that contains a single convolutional layer with kernal size of 5 * 5, a maxpooling layer and a fully connected layer. The model was tested on three public databases CK+, JAFFE and KDEF and achieved 87.32%, 76.55% and 89.4% respectively.

3   Facial Expression Recognition System

In this paper we implemented a system that automatically recognizes seven emotions: anger, disgust, fear, happy, neutral, sad and surprise using convolutional neural networks combined with some pre-processing steps. We tested our system on datasets collected from three largely used public databases namely: CK+ [17], JAFFE [20] and KDEF [8]; some pre-processing steps were applied in order to reduce the complexity of the system and increase the accuracy.

The system operates in two phases: training and test; for the training the input data is small sized grayscale normalized images that represents detected and cropped faces, faces were detected using OpenCV Cascade Classifier, these images are detached into seven classes representing seven emotions based on their labels. For the testing phase we apply the same pre-processing steps.

We drive attention that we validate our system by splitting each dataset 80% for training and 20% for validation. For the overall evaluation of our model we use the metric Accuracy and for the recognition rate of each emotion we use the confusion matrix to obtain the precision of each class.

3.1   Convolutional Neural Network

Our model comprises two blocks of convolutional layers and sub-sampling layers, it contains two convolutional layers with filter size of 32, 64 respectively, and a kernel size of $3*3$, each layer is followed by a max pooling layer of $2*2$ kernel size and a dropout operation in which 20% of nodes are eliminated in order to reduce overfitting; the output of these two blocks is fed to the first fully connected layer with 128 neurones and then to the second fully connected layer with 64 neurones and at last to the output layer containing seven neurones representing the seven emotions classes. All of the convolutional layers and the fully connected layers use Rectified Linear Unit (ReLu) function as activation function while the output layer uses Softmax activation function since we have multiple classes. The general architecture of the system can be viewed in Fig. 1.

3.2   Data Collection

Datasets used to examine our model are collected from three commonly used and public databases: the Extended Cohn Kanade (CK+) database, the Japanese Female Facial Expression (JAFFE) database and the Karolina Directed Emotional Face (KDEF) database.
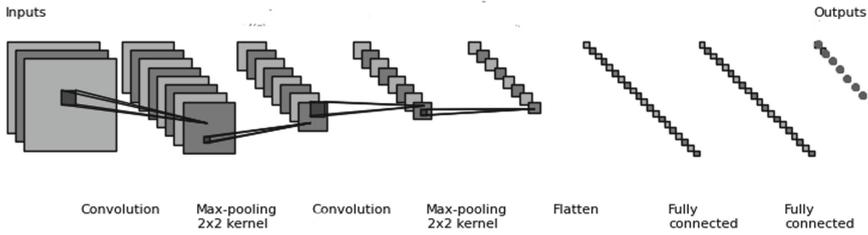
Fig. 1. Proposed CNN architecture

The CK+ database contains a series of expressions performed by males and females, each sequence begins and ends with a neutral expression; for our dataset seven expressions were selected as cited previously, and for each sequence of expression we chose the three most expressive frames except for the emotion neutral we chose only one frame. The complete number of images extracted for our dataset is 981 images distributed on seven classes. Figure 2 shows some examples of the CK+ dataset.
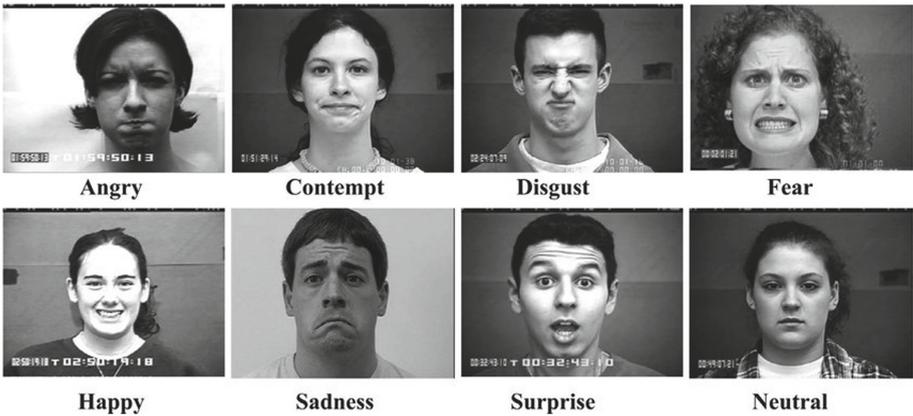
Fig. 2. Examples from CK+ database

The JAFFE database contains images divided on seven classes performed by 10 female Japanese subjects expressing emotions on different shots; the complete number of images is 213. Figure 3 shows some examples of the JAFFE database.

Fig. 3. Examples from JAFFE database

The KDEF database contains a total of 4900 images performed by 70 individuals mixed between males and females over many angles; only 490 images were adopted in our dataset (the ones with frontal view faces) divided equally on seven classes. Figure 4 shows some examples of the KDEF dataset.

Fig. 4. Examples from KDEF database

3.3   Image Pre-Processing

Based on related works and in order to decrease the complexity of the network and increase the accuracy we applied some simple pre-processing steps to our datasets.

Grayscale Conversion:
All images were subjects to grayscale conversion from RGB to GRAY in order to reduce the depth of the CNN filters from three channels to just one, that helps optimizing the performance of the model in terms of time and calculations.

Face Detection and Cropping:

In order to eliminate the meaningless information from images, we applied face detection on all images using OpenCv Cascade Classifier; after detecting the face we crop the image so that it contains only facial region. The final step after that is to resize the image to 48 * 48 pixels. Figure 5 gives an example of this operation.
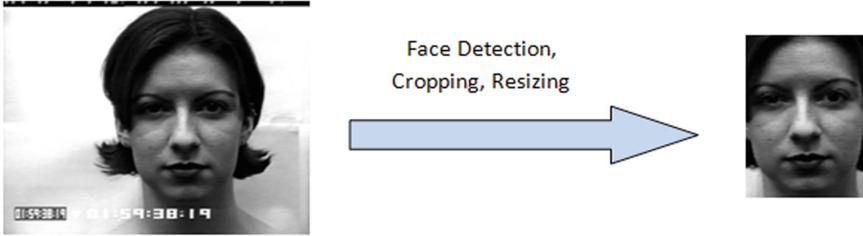

Fig. 5. Face detection and cropped using OpenCV Cascade Classifier

Intensity Normalization:

This process aims to change the range of pixel intensity values, its purpose is to convert the input image into a range of pixel values that are more uniform, in our case we normalized all of our images' pixels values from the range of 0–255 to the range of 0–1 and this is done using Image Data Generator Class in Keras API during run time.

4   Experiments and Discussions

Our system was implemented using Python programming language, Keras API and TensorFlow running in backend, each validation process uses 50 epochs, datasets was randomly divided between training and validation (80/20) using the Validation_split function in Image Data Generator class from Keras API.

We evaluated our system on three different datasets collected from public databases, we used the same model and the same parameters: batch size is set to 16, number of epochs is set to 50 and the metric used is Accuracy in addition to the confusion matrix and classification reports, each dataset is validated alone; results are discussed below.

For the CK+ dataset, 981 images were used from which 788 for training and 193 for testing; the model achieved a total accuracy of 95.89% on training and 88.23% on test, compared to results from [23] our model gives better accuracy eventhough we used the same architecture, our model also out performs results from [19]. We also observed that the recognition rate of the emotions Disgust, Happy and Neutral achieved 100% while Fear achieved 97%, Surprise 92%, Anger 82% and Sad 81%. Results are better shown in Fig. 6 and Table 1.
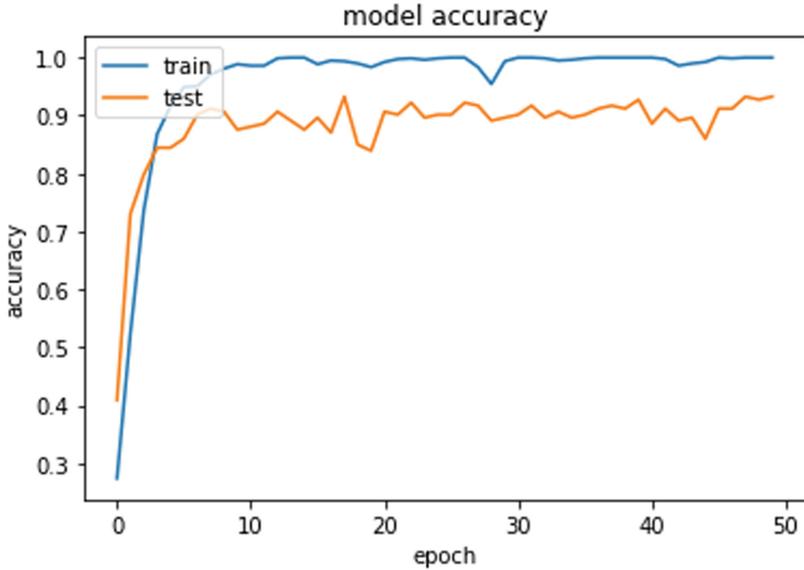
Fig. 6. Training and validation accuracy curves of the model using CK+ dataset

Table 1. Confusion matrix of the CK+ dataset

	Anger	Disgust	Fear	Happy	Neutral	Sad	Surprise
Anger	23	0	1	0	0	3	0
Disgust	0	10	0	0	0	0	0
Fear	0	0	35	0	0	0	0
Happy	5	0	0	9	0	0	1
Neutral	0	0	0	0	41	0	0
Sad	0	0	0	0	0	13	3
Surprise	0	0	0	0	0	0	49

For the JAFFE dataset, 213 images were used from which 172 for training and 41 for testing; the model out performs results from [5,19,23] by achieving a total accuracy of 92.17% on training and 86.24% on test. While [5] used a traditional method that combines HOG with SVM, and [19] presented a shallow CNN architecture combined with facial segmentation, [23] used the exact same architecture as ours with different parameters, our proposed method gave the best result therefor we can conclude that deep learning can be efficiently adopted to the problem of classification and that network parameters play a key role in the improvement of the model accuracy. We can observe that the recognition

rate of the emotions Fear, Happy, Neutral, Sad and Surprise achieved 100% while Anger and Disgust achieved only 86%. Results are better shown in Fig. 7 and Table 2.
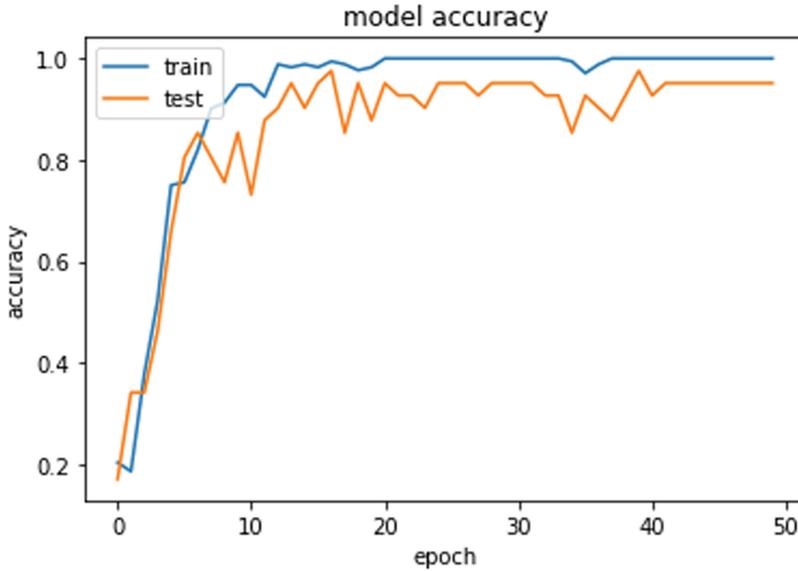

Fig. 7. Training and validation accuracy curves of the model using JAFFE dataset

For the KDEF dataset, 490 images were used from which 392 for training and 98 for testing; the model achieved a total accuracy of 90.39% on training and 82.38% on test. Our model gives better results than traditional method presented [5] while shallow architecture presented in [19] gives slightly better results. We can observe that the recognition rate of the emotions Anger, Disgust and Happy achieved 100%, Fear and Surprise achieved 92% while Neutral 91% and the lowest accuracy was the emotion Sad 54%. Results are better shown in Fig. 8 and Table 3.

Table 2. Confusion matrix of the JAFFE dataset

|	Anger	Disgust	Fear	Happy	Neutral	Sad	Surprise
Anger	6	0	1	0	0	0	0
Disgust	0	6	0	0	0	0	0
Fear	0	1	5	0	0	0	0
Happy	0	0	0	6	0	0	0
Neutral	0	0	0	0	6	0	0
Sad	0	0	0	0	0	6	0
Surprise	1	0	0	0	0	0	4
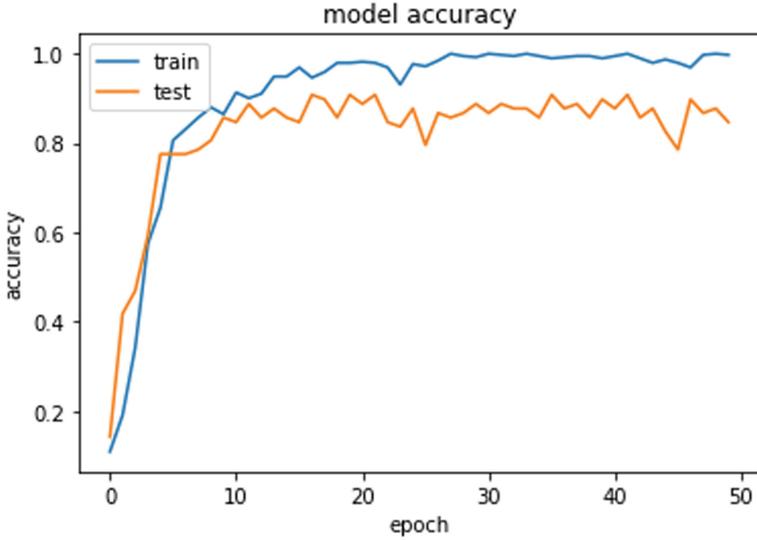
Fig. 8. Training and validation accuracy curves of the model using KDEF dataset

Table 3. Confusion matrix of the KDEF dataset

	Anger	Disgust	Fear	Happy	Neutral	Sad	Surprise
Anger	9	0	0	0	1	4	0
Disgust	0	13	0	0	0	1	0
Fear	0	0	12	0	0	1	1
Happy	0	0	0	13	0	1	0
Neutral	0	0	0	0	10	4	0
Sad	0	0	0	0	0	14	0
Surprise	0	0	1	0	0	1	12

We performed another validation inspired from [6] in which we combined JAFFE database with KDEF database, we obtained a total dataset of 703 from which 564 are dedicated for training and 139 for testing. We achieved a higher accuracy compared to the state-of-the-art [6]: 93.38% in training and 78.93% in validation. Although their architecture is deeper than ours since it contains five convolutional layers our model achieved higher accuracy in just 50 epochs while their model took 100 epochs to obtain just 70% and 200 epochs to obtain 72% and 300 epochs to obtain 78%. The recognition rate of each emotion is as follow: 100% for emotion Disgust, 88% for Surprise, 86% for Happy, 84% for Anger and Fear, 73% for Neutral and the lowest rate 59% for Sad. Detailed results are represented in Fig. 9 and Table 4.
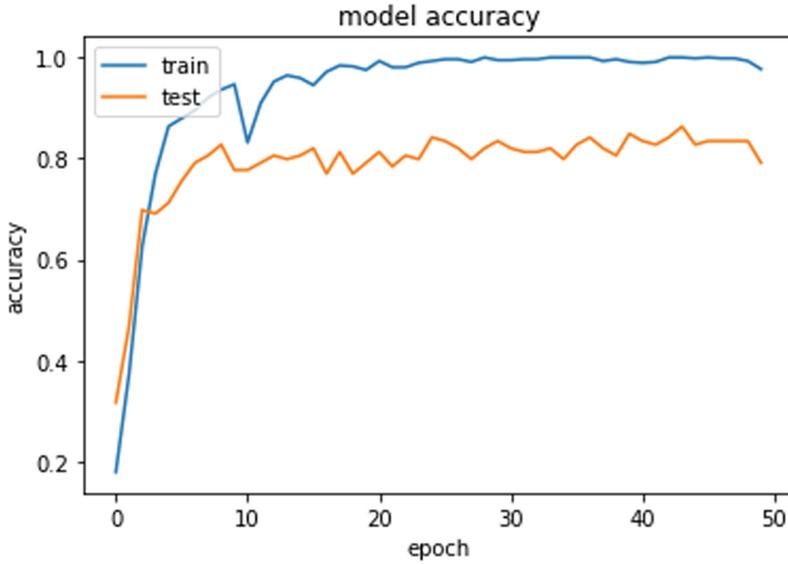
Fig. 9. Training and validation accuracy curves of the model using combined datasets JAFFE and KDEF

Table 5 resumes the recognition rates of the seven emotions namely: Anger, Disgust, Fear, Happy, Neutral, Sad and Surprise obtained from the validation of our model on the three databases CK+, JAFFE and KDEF in addition to a combined dataset between JAFFE and KDEF. We can observe that the feeling of happiness and disgust have 100% recognition rate in most of the validations; the validation on the JAFFE dataset achieved 100% recognition rate on five emotions (fear, happy, neutral, sad, surprise) and 86% on anger and disgust since these two emotions are often confusing. When using a combined dataset the only emotion that achieves 100% rate is Disgust. For most of the validations

Table 4. Confusion matrix of the combined datasets JAFFE and KDEF

	Anger	Disgust	Fear	Happy	Neutral	Sad	Surprise
Anger	16	0	0	0	0	4	0
Disgust	1	12	1	1	3	2	0
Fear	0	0	16	1	0	1	2
Happy	0	0	1	18	1	0	0
Neutral	0	0	0	0	16	4	0
Sad	1	0	1	0	2	17	0
Surprise	1	0	1	1	0	1	15

the emotion Sad shows the lowest recognition rate despite the percentage this is due to confusing it with neutral and fear emotions.

Table 5. Precision of each emotion

	Anger	Disgust	Fear	Happy	Neutral	Sad	Surprise
CK+	82%	100%	97%	100%	100%	81%	92%
JAFFE	86%	86%	100%	100%	100%	100%	100%
KDEF	100%	100%	92%	100%	91%	54%	92%
JAFFE + KDEF	84%	100%	84%	86%	73%	59%	88%

Table 6. Comparison between proposed model and state-of-art methods

Database	Method	Accuracy
CK+	[23]	80.303%
	[19]	87.32%
	Proposed Method	**88.23**%
JAFFE	[5]	76.19%
	[19]	76.55%
	[23]	76.742%
	Proposed Method	**86.24**%
KDEF	[5]	80.95%
	Proposed Method	**82.38%**
	[19]	89.4%
JAFFE + KDEF	[6]	78%
	Proposed Method	**78.93**%

Table 6 resumes the accuracies achieved in this work compared to the state of the art accuracies. We can observe that our method out performs the results of the state-of-art works in most of the cases therefore our system is considered as an effective and reliable system that responds to the problem of facial expression classification. Also we can observe that deep learning based models are more efficient than traditional models [5] because of the limitless number of features offered by raw input images instead of feature vectors. Another observation is that good classification doesn't require necessarily very deep networks [6] because the deeper the network is the more time consuming and increasing calculation complexity it will have. We also observed that sometimes a shallow architecture [19,23] is enough depending on the size of the data because usually CNNs are devoted for large databases so when we have small datasets like the ones used in our system and commonly used in the field of emotion classification using facial expression there is no need to implement deep architectures.

5   Conclusion and Future Work

In this paper we presented an effective method for classifying the six basic emotions: Anger, Disgust, Fear, Happy, Sad and Surprise plus the neutral case; this method is based on Deep Learning and uses Convolutional Neural Networks architecture, we evaluated our model on three publically available databases namely: CK+, JAFFE and KDEF and we obtained better results than recently presented works, in addition we demonstrated that deep learning based methods can give better results in classification than traditional methods and that good accuracy does not necessarily require deeper networks, sometimes a shallow one can do the job depending on the size of the dataset. Our aim in the future is to explore larger databases since deep learning requires a large number of data in order to give better results especially to improve the recognition rate of emotion Sad since it gives the lowest rate in our model.

References

1. Ahonen, T., Hadid, A., Pietikainen, M.: Face description with local binary patterns: application to face recognition. IEEE Trans. Pattern Anal. Mach. Intell. **12**, 2037–2041 (2006)
2. Chakraborty, A., Konar, A., Chakraborty, U.K., Chatterjee, A.: Emotion recognition from facial expressions and its control using fuzzy logic. IEEE Trans. Syst. Man Cybern. Part A Syst. Hum. **39**(4), 726–743 (2009)
3. Cohen, I., Sebe, N., Gozman, F., Cirelo, M.C., Huang, T.S.: Learning Bayesian network classifiers for facial expression recognition both labeled and unlabeled data. In: Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, p. I. IEEE (2003)
4. Ekman, P., Friesen, W.V.: Constants across cultures in the face and emotion. J. Pers. Soc. Psychol. **17**(2), 124 (1971)
5. Eng, S., Ali, H., Cheah, A., Chong, Y.: Facial expression recognition in JAFFE and KDEF datasets using histogram of oriented gradients and support vector machine. In: IOP Conference Series: Materials Science and Engineering, vol. 705, p. 012031. IOP Publishing (2019)
6. Ghaffar, F.: Facial emotions recognition using convolutional neural net. arXiv preprint arXiv:2001.01456 (2020)
7. Ghimire, D., Lee, J.: Geometric feature-based facial expression recognition in image sequences using multi-class adaboost and support vector machines. Sensors **13**(6), 7714–7734 (2013)
8. Goeleven, E., De Raedt, R., Leyman, L., Verschuere, B.: The Karolinska directed emotional faces: a validation study. Cogn. Emot. **22**(6), 1094–1118 (2008)
9. Happy, S., Routray, A.: Automatic facial expression recognition using features of salient facial patches. IEEE Trans. Affect. Comput. **6**(1), 1–12 (2014)
10. Harms, M.B., Martin, A., Wallace, G.L.: Facial emotion recognition in autism spectrum disorders: a review of behavioral and neuroimaging studies. Neuropsychol. Rev. **20**(3), 290–322 (2010)

11. Jafri, R., Arabnia, H.R.: A survey of face recognition techniques. J. Inf. Process. Syst. **5**(2), 41–68 (2009)
12. Kobayashi, H., Hara, F.: Recognition of six basic facial expression and their strength by neural network. In: 1992 Proceedings of the IEEE International Workshop on Robot and Human Communication, pp. 381–386. IEEE (1992)
13. Koelstra, S., Pantic, M., Patras, I.: A dynamic texture-based approach to recognition of facial actions and their temporal models. IEEE Trans. Pattern Anal. Mach. Intell. **32**(11), 1940–1954 (2010)
14. Kristensen, R.L., Tan, Z.H., Ma, Z., Guo, J.: Binary pattern flavored feature extractors for facial expression recognition: an overview. In: 2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), pp. 1131–1137. IEEE (2015)
15. Kumari, J., Rajesh, R., Pooja, K.: Facial expression recognition: a survey. Procedia Comput. Sci. **58**, 486–491 (2015)
16. Lawrence, S., Giles, C.L., Tsoi, A.C., Back, A.D.: Face recognition: a convolutional neural-network approach. IEEE Trans. Neural Netw. **8**(1), 98–113 (1997)
17. Lucey, P., Cohn, J.F., Kanade, T., Saragih, J., Ambadar, Z., Matthews, I.: The extended Cohn-Kanade dataset (CK+): a complete dataset for action unit and emotion-specified expression. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops, pp. 94–101. IEEE (2010)
18. Lucey, P., et al.: Automatically detecting pain in video through facial action units. IEEE Trans. Syst. Man Cybern. Part B **41**(3), 664–674 (2010)
19. Melaugh, R., Siddique, N., Coleman, S., Yogarajah, P.: Facial expression recognition on partial facial sections. In: 2019 11th International Symposium on Image and Signal Processing and Analysis (ISPA), pp. 193–197. IEEE (2019)
20. Michael, J., Lyons, M.K., Gyoba, J.: Japanese female facial expressions (JAFFE). Database of digital images (1997)
21. Sajjad, M., Nasir, M., Ullah, F.U.M., Muhammad, K., Sangaiah, A.K., Baik, S.W.: Raspberry pi assisted facial expression recognition framework for smart security in law-enforcement services. Inf. Sci. **479**, 416–431 (2019)
22. Sandbach, G., Zafeiriou, S., Pantic, M., Yin, L.: Static and dynamic 3d facial expression recognition: a comprehensive survey. Image Vis. Comput. **30**(10), 683–697 (2012)
23. Shan, K., Guo, J., You, W., Lu, D., Bie, R.: Automatic facial expression recognition based on a deep convolutional-neural-network structure. In: 2017 IEEE 15th International Conference on Software Engineering Research, Management and Applications (SERA), pp. 123–128. IEEE (2017)
24. Sprengelmeyer, R.: Facial expression recognition in people with medicated and unmedicated Parkinson's disease. Neuropsychologia **41**(8), 1047–1057 (2003)
25. Tian, Y., Kanade, T., Cohn, J.F.: Facial expression recognition. In: Handbook of Face Recognition, pp. 487–519. Springer (2011)
26. Valstar, M.F., et al.: Fera 2015-second facial expression recognition and analysis challenge. In: 2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG), vol. 6, pp. 1–8. IEEE (2015)
27. Valstar, M.F., Jiang, B., Mehu, M., Pantic, M., Scherer, K.: The first facial expression recognition and analysis challenge. In: Face and Gesture 2011, pp. 921–926. IEEE (2011)
28. Yang, J., Zhang, D., Frangi, A.F., Yang, J.: Two-dimensional PCA a new approach to appearance-based face representation and recognition. IEEE Trans. Pattern Anal. Mach. Intell. **26**(1), 131–137 (2004)

29. Zhang, L., Tjondronegoro, D.: Facial expression recognition using facial movement features. IEEE Trans. Affect. Comput. **2**(4), 219–229 (2011)
30. Zhang, Y.D.: Facial emotion recognition based on biorthogonal wavelet entropy, fuzzy support vector machine, and stratified cross validation. IEEE Access **4**, 8375–8385 (2016)

Materialized View Selection Using Discrete Quantum Based Differential Evolution Algorithm

Raouf Mayata[(✉)] and Abdelmadjid Boukra

Faculty of Electronics and Computer Science Laboratory LSI, USTHB BP 32,
16111 El Alia, Bab-Ezzouar, Algiers, Algeria
`mayataraouf@gmail.com`, `aboukra@usthb.dz`

Abstract. A Data warehouse is a structure that stores big amount of data. This data is exploited in the best possible ways in order to improve the efficiency of decision-making. The huge volume of data makes answering queries complex and time-consuming. Therefore, materialized views are used in order to reduce the query processing time. Since materializing all views is not possible, due to space and maintenance constraints, materialized view selection became one of the crucial decisions in designing a data warehouse for optimal efficiency. In this paper, we propose a discrete - quantum based - version of Differential Evolution DE algorithm named QDE algorithm to solve the materialized view selection (MVS) problem with space constraint. This algorithm is a merging of the original DE with Quantum Evolutionary (QEA) algorithm. The experimental results show the efficiency of the proposed algorithm compared to well known algorithms used to solve MVS problem with space constraint such as HRUA and GEA.

Keywords: Data warehouse · Materialized view selection · Differential Evolution · Metaheuristic · Quantum Evolutionary

1 Introduction

Data warehouses is a repository of data integrated from multiple sources and delivered to decision makers supporting their complex OLAP queries and helping them to make proper strategic decisions. Due to their huge volume, data warehouses' response time to complex OLAP queries is considered as a problem. To shorten that response time, several optimization methods exist in different phases of building a data warehouse. We mention *Indexing* for the physical phase, and both *fragmentation* and *views materialization* for the logical phase. Unlike regular views, a materialized view store the result of queries physically in a table. The queries executed on these tables are significantly faster than those executed on the whole raw data. However, every materialized view will occupy additional storage space, and need to be maintained (updated) from the raw

data periodically. In a data warehouse context, one always has to deal with limited resources. Therefore, all possible views can not be materialized due storage space and maintenance constraints. Materialized View Selection (MVS) is the problem of choosing among the views' universe, the set of ones that minimizes the Total Processing Cost (TPC) when materialized with respect to space or maintenance constraints. In this paper, authors will focus on the MVS problem with storage space constraint.

The MVS problem is proven to be NP-Hard. Therefore, several approaches using heuristics and meta-heuristics were proposed so far in literature.

2   Related Work

When talking about MVS problem, one has to mention the first paper dealing with it, which is [5]. At that time, they considered only the storage space constrained MVS problem. Authors proposed a greedy algorithm named HRU, this algorithm selects views to materialize in every iteration in the following way: for every view not yet chosen for materialization, the algorithm calculates its total benefit when materialized, and the view having the maximum benefit is then materialized. This step is repeated until there is not enough space to materialize further views. An extension was proposed in [1] to the selection of both views and indices. [3] also proposed heuristics based on greedy algorithm, the difference between these heuristics and HRU is that they select the view with maximum benefit per unit space. [6] compared between different greedy based algorithms for large problem instances (>10 dimensions), after proposing a search space reduction algorithm based on the observation that some hierarchically related views may have the same size. There have been several evolutionary based approaches proposed for the storage space constrained MVS problem such as [9] in which the authors proposed a Genetic Greedy Algorithm; they used the greedy concept in the repairing function. The MVS problem with maintenance constraint was first considered in [2], after noticing that storage space causes fewer problems as a constraint. They propose an inverted tree and A-star algorithms. [14] applied a hybrid evolutionary algorithm consisting of two levels of algorithm processing. The higher-level algorithm searches for good global processing plan from local processing plans based on queries. The lower level algorithm selects the best set of materialized views with the minimal total cost for a particular global processing plan. [8] proposed a genetic algorithm with a penalty function. [13] proposed an extension of [8] by replacing the penalty function with stochastic ranking procedure. In all of the aforementioned works the authors used one of three frameworks to represent all possible views. The first one is *lattice* framework introduced in [5]. We also find the Multiple View Processing Plan(MVPP) and AND-OR graph.

3   Background

3.1   Lattice Framework

V. Harinarayan et al. [5] Introduced an oriented acyclic graph G = (V, E) called *lattice* to represent the dependencies between all possible views of a star schema, where each view is characterized by a group-by clause. The Vertices V(G) of this graph represents the set of views or queries, while the Edges E(G) represents the relation between those queries. For a pair of vertices *(x,y)* we say that x depends on y $(x \leq y)$ if the query x can be answered using the result of y. An edge (v_i, v_j) between two views v_i and v_j exists only if v_i is the "immediate proper ancestor" of v_j, which means that $(v_j \leq v_i)$ and $\nexists v_k$ $(v_j \leq v_k \wedge v_k \leq v_i)$ for every three distinct v_i, v_j and v_k. Fig. 1 is the lattice representation of the views dependencies of the star schema given in the same figure.
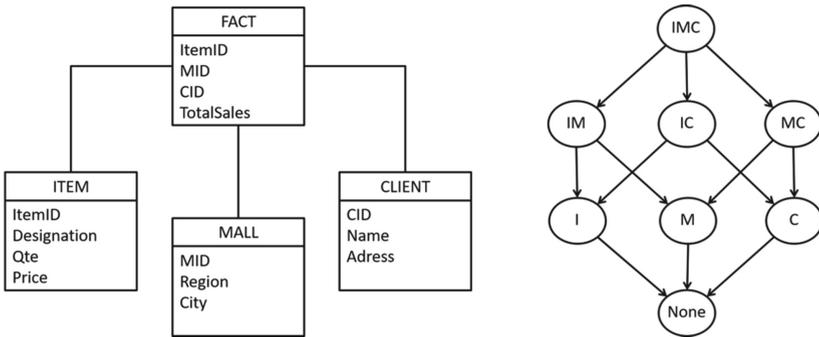

Fig. 1. Example of a lattice framework representing dependencies between views of a star schema

3.2   Space-Constrained MVS Problem

The vertices and edges of the lattice are weighted in the following way:

– Every vertex $v \in V(G)$ has three weights :
  • r_v: the initial data scan cost.
  • f_v: query frequency.
  • $|v|$: the storage space needed to materialize the view v itself.
– Every edge $(v, u) \in E(G)$ has one weight:
  • $w_q(v, u)$: processing cost of query u using v.

An additional vertex v_+ is added to the graph, this vertex represents the raw data from which every query can be calculated. For every pair of queries (u, v), we call $q(u, v)$ the query processing cost of u using v. $q(u, v)$ is the sum of the query processing costs of every edge in the shortest path between u and v, added to the initial data scan cost of $v(r_v)$. If no path links u to v, the vertex v_+ is used. Let M be the set of views to be materialized, $q(v, M)$ be the minimum query processing cost of v in presence of M. The space-constrained materialized view selection (MVS) problem can be formulated as follows :

Select a set of views M to materialize which minimizes $T(G, M)$, where:

$$T(G, M) = \sum_{v \in V(G)} f_v.q(v, M) \tag{1}$$

Under the constraint $H(G, M) \leq S$, where:

$$H(G, M) = \sum_{v \in M} |v| \tag{2}$$

S is the storage threshold not to be exceeded by the views of M.

3.3   Quantum Inspired Evolutionary Algorithm Overview

The smallest unit of information in quantum computing is called q-bit, and each q-bit is defined by two complex numbers α and β verifying: $|\alpha|^2 + |\beta|^2 = 1$. $|\alpha|^2$ represents the probability of the q-bit to be in the state '0', as for $|\beta|^2$, it represents the probability of the q-bit to be in the state '1'. In QEA, each individual q is represented by a quantum vector of length m (m being the length of solutions). Each column is represented by a single q-bit, which means: $|\alpha_i|^2 + |\beta_i|^2 = 1, i = 1, 2, 3, ...m$.

q-bit representation of solutions has the advantage of being able to represent a linear superposition of states, i.e: a single quantum vector can refer to multiple solution states with different probabilities [4], which gives the QEA a better population diversity characteristic. Like any independent evolutionary algorithm, QEA has his own operators. We mention *Measurement, Quantum interference* and *Mutation. Measurement* is the name of the operation allowing QEA to generate a solution vector using a quantum vector. As for *Quantum interference*, it intensifies the research around the best self/global solution. It consists in moving the state of each q-bit in the direction of the corresponding bit's value of the best solution in progress, and this is accomplished by making a rotation whose angle is a function of the amplitudes α_i and β_i. Like the evolutionary mutation, quantum *Mutation* is used to inspect some of the current solution's neighbors with some defined probability.

3.4   Differential Evolution

The differential Evolution (DE) algorithm was first introduced not as a separate evolutionary algorithm but as a genetic algorithm variation [12]. It is considered as a unique EA because it is not biologically motivated. DE is a population-based algorithm that is designed to optimize functions in an n-dimensional continuous domain [12]. Each individual in the population is an n-dimensional vector that represents a candidate solution to the problem. The basic idea of DE algorithm is described as follows:

Like every EA algorithm we first generate randomly a set of individuals $X = x_1, x_2, x_3, ..., x_n$. Then we proceed in the following way:

1. *Mutant vector generation:*
   For each individual x_i, a mutant vector v_i is generated using three different random individuals x_{r1}, x_{r2} and x_{r3} other than x_i, following the formula (3):

   $$V_i = x_{r1} + F(x_{r2} - x_{r3}) \tag{3}$$

   Where x_{r1}, x_{r2} and x_{r3} are three distinct vectors, and F is a real number used to scale the difference between x_{r2} and x_{r3}.

2. *Trial vector generation:*
   The mutant vector v_i is then combined (crossed over) with the individual x_i itself to generate a trial vector u_i. Crossover is done as follows (formula (4)):

   $$u_{ij} = \begin{cases} v_{ij}, & if(rand < c) \ or \ (j = Jr) \\ x_{ij}, & otherwise \end{cases} \tag{4}$$

   For every j in $[1, n]$, where n is the problem size and is also the size of u_i, v_i and x_i; u_{ij} is the jth component of u_i, v_{ij} is the jth component of v_i, x_{ij} is the jth component of the individual x_i, $rand$ is a random number taken from the uniform distribution in $[0, 1]$; c is the constant crossover rate (in $[0, 1]$); and Jr is a random integer taken from the uniform distribution in $[1, n]$.

3. *Best fit choice:*
   After the trial vector ui is created, it is compared to the individual xi, and the fittest vector in each pair (x_i, u_i) is kept to the next generation of DE algorithm. The least fit is discarded.

   $$x_{i+1} = \begin{cases} u_i, & if \ (f(u_i) < f(x_i)) \\ x_i, & otherwise \end{cases} \tag{5}$$

4   QDE Algorithm

In this paper we propose a new hybrid algorithm (QDE) using both QEA and DE algorithms. In QDE, the population is a set of quantum vectors. Each quantum vector Q_i is considered as a vector. After measurement, a set of solutions X_i is generated from every Q_i vector. Then, *mutant vectors generation* and *trial vectors generation* are applied on every vector Q_i like in the original DE algorithm. The *best fit choice* is also applied on both the original Q_i population and the new trial vectors based on the fitness of their corresponding solutions after applying measurement. After that, the *Quantum interference* operator of QEA is used on the new Q_i generation to improve it by rotating towards the global best solution. The above steps are repeated until some stop criteria are satisfied.

The main idea of the QDE is outlined in the Fig. 2 and its pseudo code is given in Algorithm 1.
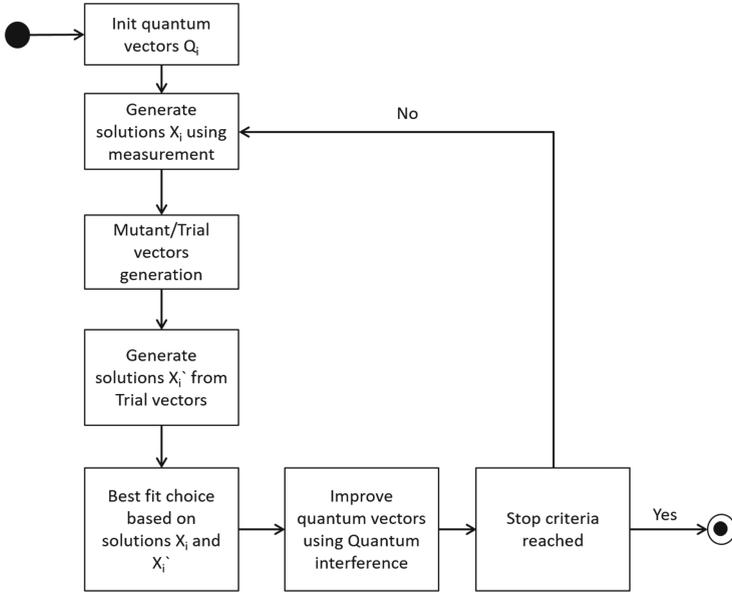
Fig. 2. Main steps of QDE Algorithm

4.1 Solution Representation

In this work, the authors represent every solution by a binary vector $Xi = (x_1, x_2, x_3, ..., x_N)$, where N is the total number of views. A case x_i takes the value 1 if the view v_i is materialized, 0 otherwise. As shown in Fig. 3, the binary vector X is the representation of the solution given by the lattice L (the lattice structure of a star model with 3 dimensions), in which the materialized views are painted in grey. In this example the views v_2 and v_3 are materialized, which corresponds to the value 1 in the second and third position of the vector X. Whereas, the rest of the views are not materialized, hence the value 0 of the remaining positions.

Algorithm 1: QDE Pseudo Code

N**:** population size;

n**:** number of views;

f**:** stepsize parameter $\in [0.4, 0.9]$;

c**:** crossover rate $\in [0.1, 1]$;

t**:** the maintenance constraint threshold $\in [0.1, 1]$;

C_{max}**:** the max space-cost when all vertices are materialized;

ra**:** rotation angle $\in [0.01\pi, 0.1\pi]$;

begin

> Initialize a population of quantums Q_i where $i \in [1, N]$;
>
> **foreach** Q_i *($i \in [1, N]$)* **do**
>
> > $X_i =$ measurement(Q_i);
> >
> > **if** *($H(X_i) > C_{max} * t$)* **then** Repair X_i;
>
> **end**
>
> **repeat**
>
> > **foreach** Q_i *($i \in [1, N]$)* **do**
> >
> > > $r_1 =$ random int $\in [1, N] : r_1 \neq i$;
> > >
> > > $r_2 =$ random int $\in [1, N] : r_2 \notin \{i, r_1\}$;
> > >
> > > $r_3 =$ random int $\in [1, N] : r_3 \notin \{i, r_1, r_2\}$;
> > >
> > > $v_i = Q_{r1} + f(Q_{r2} - Q_{r3})$;   /* v_i : mutant vector */
> > >
> > > $J_r =$ random integer $\in [1, n]$;
> > >
> > > **foreach** Q_i *($i \in [1, N]$)* **do**
> > >
> > > > /* u_i : trial vector   */
> > > >
> > > > **if** *($rand < c$)or($j = J_r$)* **then** $u_{ij} = v_{ij}$;
> > > >
> > > > **else** $u_{ij} = Q_{ij}$;
> > >
> > > **end**
> >
> > **end**
> >
> > **foreach** $i \in [1, N]$ **do**
> >
> > > $X_i' =$ measurement(u_i);
> > >
> > > **if** $H(X_i') > C_{max} * t$ **then** Repair X_i';
> > >
> > > **if** $T(X_i') < T(X_i)$ **then** $Q_i = u_i$;
> > >
> > > apply *Quantum interference* on Q_i;
> >
> > **end**
> >
> > **if** *no update on global best for 100 iterations* **then** Apply reset operator;
>
> **until** *stop criteria*;

end

4.2   Q-Bit Representation

Since $|\alpha|^2 + |\beta|^2 = 1$, it basically represents the equation of a unit circle and each point on its perimeter can be represented by a single variable θ with the Cartesian co-ordinates given by cos_θ and sin_θ where θ is defined in $[0, 2\pi]$. Therefore, instead of using α and β to represent a q-bit like QEA, we use the variable θ,
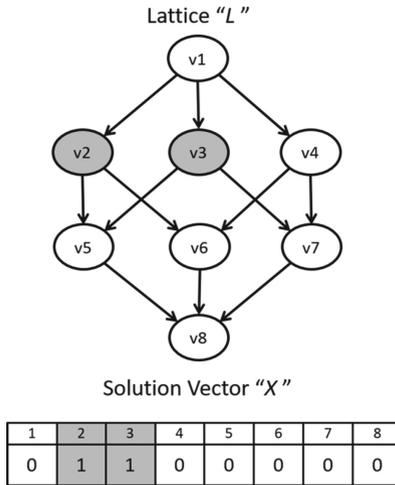
Fig. 3. Example of solution representation

from which the values α and β can be inferred. This, not only will allow us to gain space, but also will make the use of quantum interference operator easier, because the rotation angle will be added directly to the quantum itself.

4.3   Quantum Interference Application

As mentioned in the above section, quantum interference is applied by adding the rotation angle γ to the quantum directly. In order to apply this operator, we use the same look-up table used in [7] described in Table 1.

Table 1. Look up table from [7]

α	β	Best solution bit value	Angle
> 0	> 0	1	$+\gamma$
> 0	> 0	0	$-\gamma$
> 0	< 0	1	$-\gamma$
> 0	< 0	0	$+\gamma$
< 0	> 0	1	$-\gamma$
< 0	> 0	0	$+\gamma$
< 0	< 0	1	$+\gamma$
< 0	< 0	0	$-\gamma$

The choice of the rotation angle's value γ is very important. A badly chosen value can lead to premature or very late convergence or divergence.

4.4 Discrete Vs Continuous Search Space

Both QEA and DE algorithms are dedicated to solve continuous problems. In those algorithms, solution vectors move around the search space which is within a continuous real domain. However, MVS problem deals with a binary discrete search space. Some adaptation of those algorithms to solve binary discrete problems is needed. Several works like [10, 11] use transfer functions. Those functions consider the continuous search space as a probability of being at the values 0 or 1. In this work the authors follow a similar approach. Instead of having solution vectors, the Vector population is a set of quantum vectors. Those vectors generate solutions using the measurement operator mentioned earlier. So, instead of manipulating the actual position of each solution, the probability of the position being at 0 or 1 is manipulated. This way one can use any continuous search space dedicated algorithm to solve a problem having a binary discrete search space.

4.5 Dealing with Unfeasible Solutions

Since the MVS is a constrained combinatorial optimization problem, several solutions during the algorithm's application will not respect the constraint forming what is called unfeasible solutions. One of the most commonly used methods to deal with this problem is to implement a penalty function penalizing those unfeasible solutions, and ensuring that even if they have good fitness, they will not be highly evaluated among other solutions. In our work, we do not use any penalty function. Instead of that, we use a repairing function. We keep dematerializing the smallest view in every iteration, until the constraint is not violated any more. Doing that will ensure that after measurement, there will be no unfeasible solutions.

4.6 The "Reset" Operator

As in most swarm algorithms, every single particle seeks the best solution. With lack of diversity and after some iterations, the particles stagnate at the same position (around the best solution). Consequently, the population prematurely converges to local optima. To deal with this issue, the authors have introduced the reset operator (known as reseeding or extinction) to ensure the escape from the local optima case. If the best global solution does not change for a certain number of iterations, *reset operator* is applied by replacing all the quantum vectors with new ones. By doing that, and keeping the best global solution found so far, one can ensure that those vectors will seek the best solution from a different starting points, covering new areas in the search space, and ensuring diversity for the whole population. In QDE the number of iteration needed to apply the reset operator is set to 100 iterations.

5 Experimental Results

In this section we will prove the effectiveness of the proposed QDE algorithm against the well-known HRU algorithm of [5]. Also, we test it against the Genetic

Algorithm which is one of the most commonly used evolutionary algorithms used to solve MVS problem both with space and maintenance constraints. The tests were conducted on an Intel based i-5 2.4 GHz PC having 8 GB RAM. Due to several tests, the parameters were set as shown in Table 2.

Table 2. Parameters used for QDE/GEA implementation

Parameters	QDE values	GEA values
Population size	100	100
Nb of generations	1000	1000
Query frequencies	following Zipf distribution	
Scale factor	0.5	-
Cross Over rate	0.3	0.5
Rotation angle γ	0.02π	-
Mutation Pb	-	0.001

We chose to compare QDE to HRU for 5 and 6 dimensions' lattices. Table 3 shows the solution's fitness (TPC) given by HRU, compared to the average of 30 independent runs of QDE for 1000 iterations. For 5-dimension lattice, QDE outperforms HRU for almost all space thresholds. They provide same TPC for the space threshold 0.7, 0.2 and 0.1. this is due to small search area for this lattice. As for 6-dimension lattice, QDE provide better solutions for all space thresholds.

Table 3. QDE vs HRU for 5-dimension and 6-dimension lattices

	5-Dimension		6-Dimension	
t	HRU	QDE	HRU	QDE
0.9	4943892	**4939772**	5091526	**5090352**
0.8	4967327	**4967200**	5114865	**5113931**
0.7	**5000283**	**5000283**	5156049	**5143236**
0.6	5052126	**5051514**	5186824	**5180342**
0.5	5106501	**5099592**	5245149	**5231102**
0.4	5228176	**5190686**	5311518	**5296674**
0.3	5337020	**5299322**	5429357	**5385871**
0.2	**5479742**	**5479742**	5554820	**5552887**
0.1	**5864309**	**5864309**	5890921	**5866430**

Table 4, Table 5 and Table 6 contain the results of 30 independent runs of QDE and GEA on a 5, 6 and 7-dimension lattices respectively. We kept track of the mean, min and max TPC found during these runs.

Table 4. QDE vs GEA for 5-dimension lattice

t	0.1			0.2		
	mean TPC	min TPC	max TPC	mean TPC	min TPC	max TPC
QDE	**5864309**	**5864309**	**5864309**	**5479742**	**5479742**	**5479742**
GEA	5895203	**5864309**	6178534	5499864	**5479742**	5580018
t	0.3			0.4		
	mean TPC	min TPC	max TPC	mean TPC	min TPC	max TPC
QDE	**5299322**	**5299322**	**5299322**	**5190686**	**5190686**	**5190686**
GEA	5312721	**5299322**	5354626	5203998	**5190686**	5236239
t	0.5			0.6		
	mean TPC	min TPC	max TPC	mean TPC	min TPC	max TPC
QDE	**5099592**	**5099592**	**5099592**	**5051514**	**5051328**	**5052126**
GEA	5106157	**5099592**	5127608	5077674	5064453	5092808
t	0.7			0.8		
	mean TPC	min TPC	max TPC	mean TPC	min TPC	max TPC
QDE	**5000283**	**5000283**	**5000283**	**4967200**	**4967200**	**4967200**
GEA	5064873	**5000283**	5014292	4969717	**4967200**	4975976

By observing Table 4, one can notice that QDE outperforms GEA, for all thresholds. What seems to be the best solution found by GEA (min TPC) is almost always the mean TPC of QDE. i.e QDE found that solution for every single run since mean = min = max for all thresholds except 0.6 as shown in Table 4.

As for 6-dimesion lattice, Table 5 shows that QDE finds better solutions since its mean, min and max TPC are better than GEA, except for thresholds 0.1 and 0.3 where GEA finds a similar min TPC.

Finally for 7-dimesion lattice (Table 6), the supremacy of QDE over GEA becomes very clear since QDE outperforms GEA for all the statistical indicators (min,max and mean) TPC and that is for all the thresholds with no exception. This proves the scalability of QDE's solutions quality.

Table 5. QDE vs GEA for 6-dimension lattice

t	0.1			0.2		
	mean TPC	min TPC	max TPC	mean TPC	min TPC	max TPC
QDE	**5866430**	**5866430**	**5866430**	**5552887**	**5552444**	**5554820**
GEA	5903295	**5866430**	6005452	5584828	5555631	5662251
t	**0.3**			**0.4**		
	mean TPC	min TPC	max TPC	mean TPC	min TPC	max TPC
QDE	**5385871**	**5385457**	**5388109**	**5296674**	**5295998**	**5301854**
GEA	5415317	**5385457**	5463672	5309944	5301737	5327368
textbit	**0.5**			**0.6**		
	mean TPC	min TPC	max TPC	mean TPC	min TPC	max TPC
QDE	**5231102**	**5229696**	**5234680**	**5180342**	**5179914**	**5181307**
GEA	5241880	5233912	5251542	5186114	5181298	5194346
t	**0.7**			**0.8**		
	mean TPC	min TPC	max TPC	mean TPC	min TPC	max TPC
QDE	**5143236**	**5142962**	**5144289**	**5113931**	**5113650**	**5116112**
GEA	5146883	5143262	5152941	5115813	5113974	5118915

Table 6. QDE vs GEA for 7-dimension lattice

t	0.1			0.2		
	mean TPC	min TPC	max TPC	mean TPC	min TPC	max TPC
QDE	**5866197**	**5864225**	**5878000**	**5589373**	**5585840**	**5598541**
GEA	5920038	5873451	6009651	5619766	5591987	5692837
t	**0.3**			**0.4**		
	mean TPC	min TPC	max TPC	mean TPC	min TPC	max TPC
QDE	**5454278**	**5451819**	**5459491**	**5366446**	**5364286**	**5369022**
GEA	5474363	5460181	5505542	5380534	5369855	5397451
t	**0.5**			**0.6**		
	mean TPC	min TPC	max TPC	mean TPC	min TPC	max TPC
QDE	**5303783**	**5301315**	**5306194**	**5259941**	**5258691**	**5261597**
GEA	5310349	5303681	5319140	5265795	5261428	5274166
t	**0.7**			**0.8**		
	mean TPC	min TPC	max TPC	mean TPC	min TPC	max TPC
QDE	**5226450**	**5225031**	**5228281**	**5200795**	**5200095**	**5201447**
GEA	5229828	5226855	5233312	5202571	5200888	5205480

6   Conclusion

In this paper, we adapted a binary version of Differential Evolution (DE) Algorithm to solve MVS problem with space constraint. We used the lattice structure of [5] to represent hierarchy between different views. Also, adaptation of the DE algorithm to the discrete binary case was done through merging DE algorithm with the QEA algorithm instead of the use of transformation functions such as Sigmoid or V-shaped. The Experimental results show the efficiency of the proposed QDE algorithm in comparison with HRU and GEA algorithms. That is shown through the TPC values of QDE, HRU and GEA which is significantly better in QDE. As perspective, we want to investigate the use QEA alongside with other evolutionary algorithms. That is, to study further the efficiency of QEA when use instead of transformation functions.

References

1. Gupta, H., Harinarayan, V., Rajaraman, A., Ullman, J.D.: Index selection for OLAP. In: 13th International Conference on Data Engineering, 1997. Proceedings. pp. 208–219 (1997). https://doi.org/10.1109/ICDE.1997.581755
2. Gupta, H., Mumick, I.S.: Selection of views to materialize under a maintenance cost constraint. In: Proceedings of the 7th International Conference on Database Theory, vol. 13, pp. 453–470 (1999).https://doi.org/10.1007/3-540-49257-7-28, http://dl.acm.org/citation.cfm?id=645503.656261
3. Gupta, H., Mumick, I.S.: Selection of views to materialize in a data warehouse. IEEE Trans. Knowl. Data Eng. **17**(1), 24–43 (2005). https://doi.org/10.1109/TKDE.2005.16
4. Han, K.H., Kim, J.H.: Quantum-inspired evolutionary algorithm for a class of combinatorial optimization. IEEE Trans. Evol. Comput. **6**(6), 580–593 (2002). https://doi.org/10.1109/TEVC.2002.804320
5. Harinarayan, V., Rajaraman, A., Ullman, J.D.: Implementing data cubes efficiently. ACM SIGMOD Record **25**(2), 205–216 (1996). https://doi.org/10.1145/235968.233333. http://portal.acm.org/citation.cfm?doid=235968.233333
6. Łatuszko, M., Pytlak, R.: Methods for solving the mean query execution time minimization problem. Eur. J. Oper. Res. **246**(2), 582–596 (2015). https://doi.org/10.1016/j.ejor.2015.04.041
7. Layeb, A.: A novel quantum inspired cuckoo search for knapsack problems. Int. J. Bio-Inspl. Comput. **3**(5), 297 (2011)
8. Lee, M., Hammer, J.: speeding up materialized view selection in data warehouses using a randomized algorithm. Int. J. Coop. Inf. Syst. **10**(03), 327–353 (2001). https://doi.org/10.1142/S0218843001000370
9. Lin, W.Y., Kuo, I.C.: A Genetic Selection Algorithm for OLAP Data Cubes.Knowledge and Information Systems **6**(1), 83–102 (2004). https://doi.org/10.1007/s10115-003-0093-x, http://www.springerlink.com/openurl.asp?genre=article&id=doi:10.1007/s10115-003-0093-x
10. Mirjalili, S., Mirjalili, S.M., Yang, X.S.: Binary bat algorithm. Neural Comput. Appl. **25**(3–4), 663–681 (2014). https://doi.org/10.1007/s00521-013-1525-5
11. Rashedi, E., Nezamabadi-Pour, H., Saryazdi, S.: BGSA: binary gravitational search algorithm (2010). https://doi.org/10.1007/s11047-009-9175-3

12. Simon, D.: Evolutionary Optimization Algorithms. Wiley (2013). https://books.google.dz/books?id=gwUwIEPqk30C
13. Yu, J.X., Yao, X., Choi, C.H., Gou, G.: Materialized view selection as constrained evolutionary optimization. IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. **33**(4), 458–467 (2003). https://doi.org/10.1109/TSMCC.2003.818494
14. Zhang, C., Yao, X., Yang, J.: An evolutionary approach to materialized views selection in a data warehouse environment. IEEE Trans. Syst. Man Cybern. Part C: Appl. Rev. **31**(3), 282–294 (2001). https://doi.org/10.1109/5326.971656

Context-Aware Based Evolutionary Collaborative Filtering Algorithm

Ibtissem Gasmi[1(✉)] , Fouzia Anguel[1,2] , Hassina Seridi-Bouchelaghem[3],
and Nabiha Azizi[3]

[1] Department of Computer Science, Chadli Bendjedid University, El Tarf, Algeria
`gasmibtissem@gmail.com`
[2] LISCO Laboratory, Badji Mokhtar University, Annaba, Algeria
[3] LabGED Laboratory, Badji Mokhtar University, Annaba, Algeria

Abstract. Recommender systems are tools that provide personalized suggestions of items for users. They must be able to adapt constantly to user preferences and behavior in order to generate relevant recommendations. However, initial works in this field do not consider the context in which the users are. In recent years, a new recommendation technique, called Context-Aware Recommender System (CARS), has emerged. This approach integrates contextual information about users and/or items in the recommendation process to satisfy even more users' needs. Therefore, accurate prediction depends upon the degree to which a recommendation method has incorporated the relevant contextual data. To address this issue, we propose to combine user based collaborative filtering with the Genetic Algorithm based meta-heuristic in order to provide better predictions for users. The proposed model uses a weighting function which incorporates the contextual factors that influence the users' decisions. It is based on the Genetic Algorithm based meta-heuristic to estimate, for each contextual parameter, a degree of importance that would reduce the mean absolute error and increase the F-measure. Experimental results from Movielens dataset validate that our proposed algorithm improves recommendations accuracy.

Keywords: Recommender System · Collaborative filtering · Context · Genetic Algorithm · Adaptation

1 Introduction

There has been an explosion, in recent years, of the number of digital items (i.e. document, product, music, book, etc.) available on the web. This increases the overload of information and provides multiple choices for users [1]. Therefore, searching and finding target items are hard and time consuming tasks for costumers [2]. A recommender system offers a feasible solution to this problem. It aims to provide information or items that are likely of interest to users. It has been successfully employed in e-commerce and academic applications [3].

Last few years, the importance of contextual information has motivated many researchers to incorporate them in their systems. However, the notion of context is complex and not easily integrated into recommendation process. Hence, many attempts have been made in order to define the context in various domains [4]. Abowd et al. highlights that "*Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves*" [3].

Contextual information can be integrated in various stages of the recommendation process. Adomavicius and Tuzhilin distinguish three main context-aware techniques [4]: prefiltering, postfiltering and contextual modeling. In *contextual prefiltering*, only items which are appropriate to a given context are considered for recommendations. That is to say, contextual factors are used to select relevant data which will be used to generate recommendations by any traditional algorithm. However, the *Contextual postfiltering* approach ignores contextual information when generating recommendations. It applies the filter after generating the ranked list of all candidate items. *Contextual modeling* approach learns the predictive model by using a multidimensional recommendation function. Thus, context is not used as filters in the recommendation process. It is incorporated directly into the model.

Furthermore, different context factors can influence user's preferences such as latent contexts [5], location information [6], temporal information [2], social information [4] etc. Time factor has a significant impact on people's preferences and enables tracking the evolution of user needs [2]. For example, there are things we preferred in the past which are totally outdated today. We may have dreamed of having a Walkman and now we have moved on to other choices. Other contextual data like occupation, age and gender, etc., can also affect the preferences of users. Typically, the recommendation of books is highly dependent on the profession and the passions of users. On the other hand, the degree of influence of contextual information on user preferences differs from one contextual parameter to another. Therefore, contextual information has some influence on user preferences, which can be expressed with a degree of importance.

Identifying the context factors that influence evaluation process is important for understanding user preferences. It has been shown that the context plays an important role in providing relevant suggestions for users [3, 7]. However, integrating contextual information represents a real challenge in recommender systems. Specifically, data sparsity becomes more severe when user preferences are filtered with contextual factors [8]. In fact, using too many contextual parameters increases the data sparsity and few context factors in recommendations fail to bring the contextual impact [1]. We thus attempt to address this issue by incorporating directly context into the prediction process instead of removing items that are irrelevant in a given context.

Gasmi et al. [9] developed a context-aware recommendation method which considers time effect in collaborative filtering by introducing a weighting function that takes into account the changes in user interests over time. The proposed function keeps user's recent, long and periodic interests, and attenuates user's old and short desires. The purpose of this study is to examine the impact of context on recommendations quality by extending the weighting function proposed in [9] with more contextual data. Thus,

this paper presents a context-aware method that combines the user based collaborative filtering with the Genetic Algorithm based meta-heuristic. The extended function is computed for each rating used in the prediction process. It is a simple combination of contextual factors (user's age, profession and gender; item genre and evaluation time) with their degree of importance to consider the similarity between user's contexts. The GA-based meta-heuristic is used to assign suitable weight to each context feature. Indeed, this meta-heuristic is a simple and flexible method that considers large research space. It has been widely applied by many researchers to resolve complex problems [10–12].

In this paper, we try to answer the following questions:

- Does the prediction method uses in this paper is suitable for the proposed model?
- Does the proposed weighting function improves the prediction quality?
- Among the used contextual factors, which ones are discriminating and their combination really improve the results?

The rest of the paper is organized as follows. Section 2 introduces some related works. Section 3 describes in details the proposed model. Section 4 shows the different steps of the proposed algorithm. Section 5 presents experimental works. It provides dataset details, results of different experiments and discussion. The final section provides a conclusion and future perspectives.

2 Related Work

With the advent of the World Wide Web and big data, recommendation systems (RSs) are becoming more and more popular [1]. The purpose of these systems is to propose different services to different users. Despite their success in many applications, RS techniques suffer a number of problems that remain to be resolved. Usually users express opinions on a few items. This leads to sparse user × item matrices. Thus, making accurate recommendations for users with extremely few preferences remains a great challenge in RSs. In order to handle the data sparsity problem, many techniques have been adopted in the literature [13, 14]. In the last few years, several studies have been carried out to improve the quality of recommendation systems by incorporating contextual information [5, 7, 15]. Particularly, many attempts have been made to exploit the time context for users modeling and recommendation process [2, 7, 15]. Zhang et al. [15] developed a hidden semi-Markov model (HSMM) framework to tackle users' drifting interests. This model captures the heterogeneity of users in terms of interest duration by allowing them to stay in different (latent) states for various time periods.

Furthermore, a number of researches on recommender systems have exploited machine learning methods such as Genetic Algorithm (GA) [10, 16] and Particle Swarm Optimization (PSO) [17]. GA is a randomized search and optimization technique that has mainly been used in two aspects of recommender system: clustering [10] and hybrid methods [18]. Salehi et al. [16] considered, in their contribution, weights of implicit or latent attributes of materials for learners as chromosomes. Then, they optimized the weights according to historical ratings. Karatzoglou et al. presented a genetic collaborative filtering model that is based on a generalization of matrix factorization to address

contextual recommendation problems [19]. Other authors developed a similarity metric that computes the similarity values between users using the genetic algorithm [11, 12].

Some previous works rely only on explicit ratings. However, in real life, the way in which two people or items are said to be similar is not based solely on evaluations, but also on other factors such as contextual data. Moreover, some above context-based approaches have focused on using the unitary context because multi-dimensional contexts will aggravate the sparsity problem, which means that the user preference matrix would become extremely sparse. Consequently, in most multidimensional situations, there are not enough or even no preferences. To alleviate the sparsity problem for CARS, this paper combines user based collaborative filtering with the GA-based meta-heuristic in order to provide better predictions for users. The proposed model uses a weighting function which incorporates the contextual data that influence the users' decisions. The details of our proposition are more explained in the next section.

3   Methodology

Context-Aware Recommender System focuses on the improvement of accuracy and a user's satisfaction by incorporating context features while making recommendations [4]. However, there are several challenges open in these systems. Data sparsity is one such concern which becomes more severe when user preferences are diluted with contextual conditions. Addressing this issue, we introduce a weighting function which directly integrates context into the collaborative filtering algorithm. The proposed function considers the similarity between the active user and his neighbor in term of gender, age group and occupational category. It also favorites the users' most recent evaluations and the items genres that are often consulted and well evaluated. Furthermore, the proposed model uses the Genetic Algorithm based meta-heuristic in order to assign the optimal weight for each contextual information. The values and weights of context features are then combined to compute the weighting function which will be used in the calculation of predictions.

3.1   GA-Based Meta-heuristic to Learn Optimal Weight Vector

A genetic algorithm is a search method based on the concepts of natural evolution. A population of solutions is manipulated through a number of generations, according to the principals of natural selection [11]. In our study, we assume that ratings, which are most similar in context, are more relevant in making predictions. Thus, we use a GA-based meta-heuristic to learn the optimal weight for each context feature in order to control the contribution of contextual factors in the recommendation process.

3.2   Genetic Representation

GA uses a collection of candidate solutions called population where each one represents a chromosome. In our model, chromosomes are vectors of real values in the range of [0, 1]. The sum of the weights in a chromosome should be equal to 1. Further, the number of genes in each chromosome is equal to the number of contextual features (user's age,

user's gender, user's occupation, movie's genre, time evaluation). Thus, a chromosome can be represented by a vector $(w_a, w_s, w_o, w_g, w_t)$. Where w_a is the weight of the user's age, w_s denotes the weight associated to the user's gender, w_o represents the weight of user's occupation. w_g is the weight corresponding to the movie genre and w_t designates the weight of the evaluation time. In fact, the initial population is filled with randomly generated vectors.

3.3 Fitness Functions

The fitness function is used to quantify the optimality of the weight vector. Two fitness functions are used in this paper: Mean absolute error (MAE) and F-measure. The fitness value of *k-th* chromosome is computed using the Mean absolute error by Eq. (1) and F-measure by Eq. (2):

$$fitness1 = \frac{\sum_{(u,i) \in Q} \left| P_{u,i} - P'_{ui} \right|}{|Q|} \tag{1}$$

|Q| is the cardinality of a training set. $P'_{u,i}$ and $P_{u,i}$ are real rating and predicted rating, respectively.

$$fitness2 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{2}$$

F-measure is the harmonic mean of Precision and Recall. Precision is a percentage of recommended items that are relevant. It can be expressed by Eq. (3).

$$Precision = \frac{Number\ of\ correct\ recommendation}{Total\ number\ of\ recommendations} \tag{3}$$

The Recall is a percentage of the relevant item that is recommended. The formula for Recall is given in Eq. (4).

$$Recall = \frac{Number\ of\ correct\ recommendation}{Total\ number\ of\ relevant\ recommendations} \tag{4}$$

The predicted ratings are calculated according to the following steps:

Neighborhood Formation
The most important step in collaborative filtering is the measure of similarities between users. Firstly, the proposed algorithm collects those users who shared common items with the active user. Then, the similarity weight between the active user and his neighborhood is computed by Pearson correlation coefficient given in Eq. (5):

$$corr_{u,v} = \frac{\sum_X (R_{u,i} - \bar{R}_u)(R_{v,i} - \bar{R}_v)}{\sqrt{\sum_X (R_{u,i} - \bar{R}_u)^2 \sum_X (R_{v,i} - \bar{R}_v)^2}} \tag{5}$$

$R_{u,i}$ is the rating of user u on item i. X denotes users u and v. the set of common items of $\bar{R}_u$ is the average ratings of user u.

Prediction

After the similarity weights are computed between user u and his neighborhood, a prediction value of the user u to unrated item i can be computed using the Eq. (6):

$$P_{u,i} = \bar{R}_u + \frac{\sum_{j=1}^{K} \left(R_{j,i} \times corr_{u,j} \times f(u, i, j) \right)}{\sum_{j=1}^{K} \left(corr_{u,j} \times f(u, i, j) \right)} \tag{6}$$

$corr_{u,j}$ is the similarity weight between users u and j. $R_{u,j}$ $R_{j,i}$ is the rating of user j for item i. K is the number of neighborhoods of user u. The proposed weighting function $f(u, i, j)$ represents the context of ratings. Its values are in the range of [0, 1]. It is defined according to Eq. (7):

$$f(u, i, j) = 1/2 \times \left(\frac{\left(e^{-t \times w_t} + \gamma \right) \times w_c \times \beta}{2} + \delta \right) \tag{7}$$

$e^{-t \times w_t}$ is defined in the range of [0, 1]. It is a monotonic decreasing coefficient with an exponential form in order to express the changes of user's interests over time by making users' most recent evaluations more significant than the older ones. The parameter t can be expressed by Eq. (8) [9]:

$$t = \frac{t_0 - t_d}{\Delta t} \tag{8}$$

t_0 represents the time of the last evaluation given by user u. t_d is the time of the last evaluation given by user u to an item having the same genre as item i. Δt corresponds to the average duration between each two successive ratings of user u.

Parameters β, γ and δ are defined in the range of [0, 1]. They are computed using Eqs. (9), (10) [9] and Eq. (11), respectively:

$$\beta = X/Y \tag{9}$$

$$\gamma = W/G \tag{10}$$

X is the sum of the ratings given by user u to items having the same genre as item i. Y represents the sum of the ratings of user u. G is the number of items rated by user u and having the same genre as item i. Z is the number of items rated by user u.

$$\delta = \frac{w_o \times foccupation(u, j) + w_a \times fage(u, j) + w_s \times fgender(u, j)}{w_o + w_a + w_s} \tag{11}$$

Item genres, age groups and occupational categories are illustrated in the Table 1. (w_a, w_s, w_o, w_g, w_t) represents the weight vector computed using GA-based meta-heuristic.

foccupation(u, j), *fage(u, j)* and *fgender(u, j)* are distance functions defined by Eq. (12), Eq. (13) and Eq. (14), respectively.

$$foccupation(u, j) = \begin{cases} 1 & \text{if users } u \text{ and } j \text{ belong to the same occupational category} \\ 0 & \text{otherwise} \end{cases}$$

$$(12)$$

$$fage(u, j) = \begin{cases} 1 & \text{if users } u \text{ and } j \text{ belong to the same age group} \\ 0.75 & \text{if there is only one age group between users } u \text{ and } j \\ 0.5 & \text{if there are two age groups between users } u \text{ and } j \\ 0.25 & \text{if there are three age groups between users } u \text{ and } j \\ 0 & \text{otherwise} \end{cases} \quad (13)$$

$$fgender(u, j) = \begin{cases} 1 & \text{if users } u \text{ and } j \text{ have the same gender} \\ 0 & \text{otherwise} \end{cases}$$

$$(14)$$

For example, if user *u* is twenty-seven years old and user *j* is thirty years, *fage(u, j)* = 1 because the both users belong to the same group (25–34). However, if user *j* is fifty years old, *fage(u, j)* = 0.25 because user *u* belong to the group (25–34) and user *j* belong to the group (50–55). Thus, there are three age groups between users *u* and *j*. Consequently, δ equal to 1, means that users *u* and *j* have the same gender, the same age group and the same occupational category.

Table 1. The context information

Age group	Occupational category	Genre
Under 18	executive/administrator/librarian	Action
18–24	artist/entertainment	Adventure
25–34	doctor/healthcare/scientist	Animation
35–44	engineer/technician/programmer	Children's
45–49	educator/student	Comedy
50–55	marketing/salesman	Crime
	lawyer/writer	Documentary
	retired/homemaker/none	Drama
	other	Fantasy
		Film-Noir
		Horror
		Musical
		Mystery
		Romance
		Sci-Fi
		Thriller
		War
		Western

3.4 Genetic Operators

As usual, our GA-based meta-heuristic applies the common operators: selection, crossover and mutation. Parents are selected for crossover using a *roulette wheel selection* process. In the roulette wheel, individuals are selected with a probability equal to the proportion that their calculated fitness contributes to the sum of the fitness of all members of the population. Then, *two-point crossover* is used to combine two parents with a crossover probability. Thus, two children will be produced with characteristics of both parents. After applying crossover, chromosomes will be mutated with a mutation probability, by altering one or more gene values. A random position is chosen and replaced by a random number between 0 and 1. Finally, an elitist strategy is applied. So, chromosomes with the highest fitness are always passed on, without mutation, to the next generation.

3.5 Termination Condition

The stop criterion of our GA-based meta-heuristic is to reach a maximum number of iterations. We only keep the two best chromosomes from each generation to obtain the next one (elitist selection). Then, we complete the size of the new population by selecting the remaining individuals in a random manner. At the end of the algorithm, we get the optimal weight vector which will be used to calculate predictions.

4   System Description

The GA-based meta-heuristic is performed in the learning phase. The different stages are as follows:
Input: Movielens dataset
Output: optimal weight vector
Operations process:

1. Create the initial population
2. Evaluate each individual in the initial population, according to Eq. (1) or Eq. (2)
3. Select individuals
4. Apply crossover and mutation operators
5. Evaluate new individuals according to Eq. (1) or Eq. (2)
6. Apply elitist strategy
7. Replace the parent population with the new generation
8. Go to step 3 until the number of iterations is not reached

Operation steps of the proposed algorithm are performed during the testing phase after obtaining the optimal weight vector. They are described as follows:
Input: Movielens dataset, optimal weight vector
Output: Predicted *user* $\times$ *item* matrix

Operations process:

1. For each user of the test base:

   – Compute the neighbour of target user according to Eq. (5).
   – For each neighbour, compute the weighting function according to Eq. (7)
   – Compute predicted rating according to Eq. (6).

9. Return predicted *user* × *item* matrix

5   Results and Discussion

To demonstrate the effectiveness of our method, we have conducted several experiments on the most popular Movielens[1] dataset. All experiments have been performed by randomly splitting dataset into two subsets: training set (80%) and testing set (20%). We have performed two types of experiments: The first one concerns the training of the GA-based meta-heuristic. It consists in carrying out numerous tests to select the best configuration of its parameters. The second category concerns the evaluation of the overall quality of the proposed model after learning it.

5.1   Dataset

Movielens is widely used to evaluate recommender systems. This dataset is provided by the University of American Minnesota GroupLens project group. The ML100K dataset consists of 943 users, 1682 movies and 100000 ratings. The information of users includes: UserID, Gender, Age and Occupation and those of the items are MovieID, Title and Genres. All ratings are contained in the file ratings.dat. Each line of this file represents one rating of one movie by one user, and has the following format: UserID::MovieID::Rating::Timestamp. The lines within this file are ordered first by UserID, then, within the user, by MovieID. Ratings are made on a 5 star scale. Timestamps represents seconds since midnight Coordinated Universal Time (UTC) of January 1, 1970. The sparsity of the ML100K dataset is 93.7%.

5.2   Evaluation Metrics

The objective of this research is to provide efficient recommendations using the item ratings and contextual information. We use Mean Absolute Error and F-measure to evaluate the performance of the proposed model and to compare algorithms which are used in this work.

---

[1] http://www.grouplens.org/.

5.3   Experiment 1: Impact of the Population Size

This experiment consists in evaluating the impact of different population sizes on the performance of the proposed algorithm. We set the mutation rate to 0.01 and the crossover rate to 0.5. We have chosen these non-high values to explore clearly the influence of population size. Figure 1 shows the evolution of the mean absolute error over iterations for different population size. It can be seen that the error decrease very slowly over iterations when the population size equal to 5. However, for a size greater than 30, the MAE follows a decreasing trend in the beginning, and it remains stable after a number of iterations. Overall, population size is one of the most important features of the genetic algorithm. A large population would increase the diversity, but requires a high runtime. On the other hand, small population can induce the risk of premature convergence due to loss of diversity.
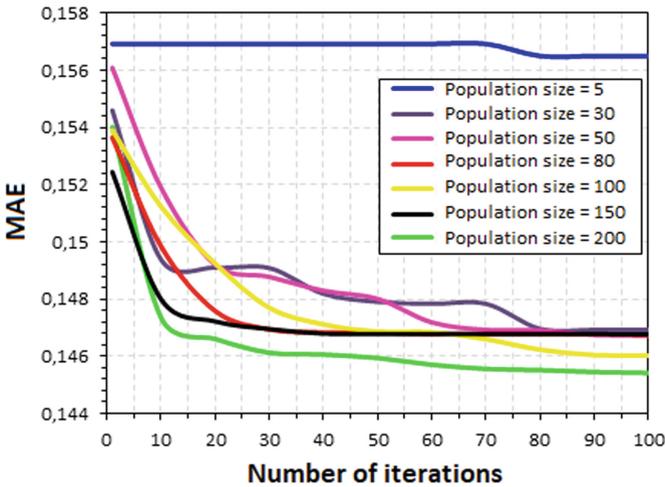
Fig. 1. Impact of the population size on the Mean Absolute Error

5.4   Experiment 2: Impact of the Crossover Probability

For this experiment, the proposed algorithm was performed with a population size of 30, a mutation rate of 0.01 and a crossover rate varying between 0.2 and 0.95. Figure 2 illustrates the evolution of the Mean Absolute Error over iterations for different crossover rates. As is observed, the error declines during the iterations and its best value is obtained when the crossover rate is 0.95. In brief, high crossover rate improves the results by accelerating the convergence of the algorithm.
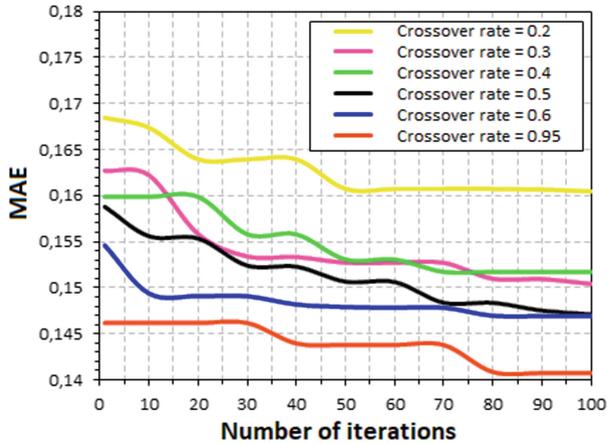
Fig. 2. Impact of the crossover rate on the Mean Absolute Error

5.5 Experiment 3: Impact of the Mutation Probability

Here, we set the population size to 30 and the crossover rate to 0.5. Figure 3 shows the evolution of the mean absolute error over iterations for different mutation rates. We can see that MAE decreases gradually during early iterations. It then levels out in the following generations. It is also shown that, from the 60th iteration, we have absolutely the same error value when a mutation rate $\in$ [0.2, 0.4]. Clearly, a low probability of mutation promotes diversity in the population. However, high mutation rate leads the
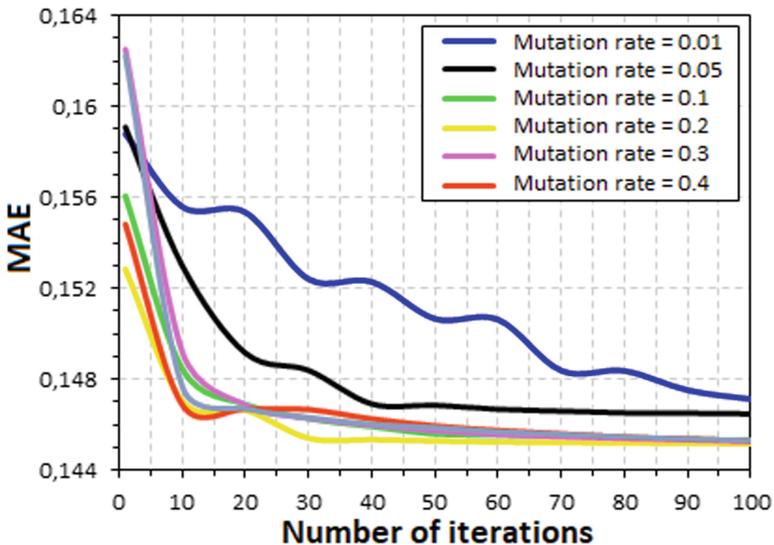
Fig. 3. Impact of the mutation rate on the Mean Absolute Error

search to be random. This does not arise in this experiment because of the strategy of elitism which allows keeping the best individuals in the population.

The Table 2 summarizes the best settings obtained after adjusting the parameters of the GA-based meta-heuristic used in this work.

Table 2. GA-based meta-heuristic parameters

Setting	Value
Population size	80
Number of iterations	100
Crossover probability	0.8
Mutation probability	0.2
Selection strategy	Roulette Wheel
Crossover strategy	Two-point
Mutation strategy	Single point

5.6   Experiment 3: Evaluation of the Proposed Weighting Function

We have conducted several experiments to compare the performances of the proposed system with some baseline models. We measure the accuracy of our model using the MAE and F-measure. Various similarity thresholds were also executed in order to understand and evaluate the corresponding behavior of the recommender models. We use User-CF to represent the user based collaborative filtering algorithm, CF1 to denote the model proposed by Gasmi et al. [9], MAE-CF to represent the proposed algorithm using the Mean Absolute Error as fitness function and F-CF to denote the proposed algorithm using the F-measure as fitness function. Figure 4 illustrates the evolution of the Mean Absolute Error through the similarity threshold. The most obvious trend in the graph is that MAE decrease with similarity threshold (T) for CF1, MAE-CF and F-CF. We notice a decline in error that is practically the same for MAE-CF and F-CF: MAE slumps nearly by 19% when $T \in [0, 0.6]$ and over 0.6 it decreases linearly about 34%. In the case of CF1, the error is relatively high, it starts at 0.8 and remains stable until $T = 0.1$. However, when $T \in [0.1, 0.6]$, the percentage decrease is 34.54% and it reaches a value of 53.57% for $T > 0.6$. It is also clear that the error value hit rock bottom when $T = 0.9$. Furthermore, all values of the proposed model are lower than traditional User-CF. To sum up, the proposed function is less sensitive to the similarity threshold compared to that used by CF1. This implies the decrease of MAE even when using a neighborhood containing too many users with bringing some no correlated information.
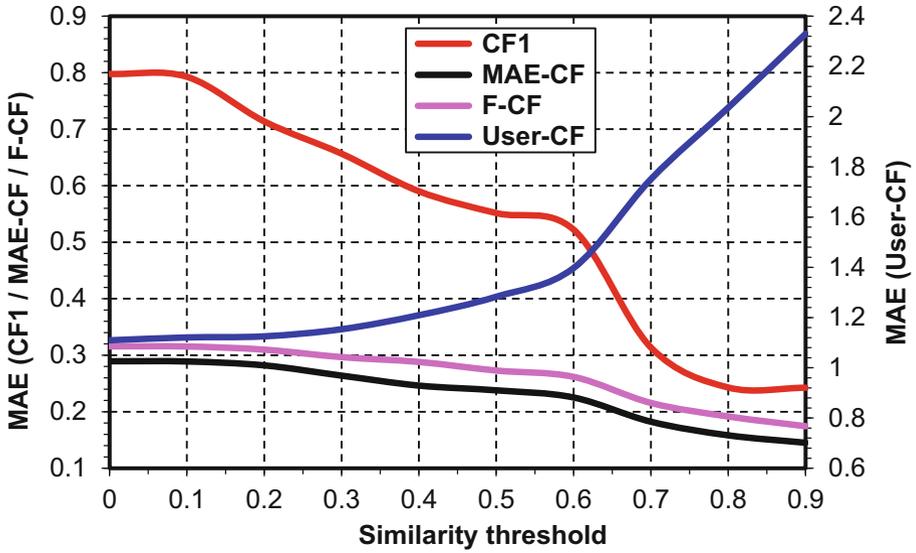
Fig. 4. MAE with different similarity threshold

On the other hand, Fig. 5 outlines the evolution of F-measure through the similarity threshold. At the first glance, we notice the rise and fall of F-measure for CF1, MAE-CF and F-CF. It touches the highest point at T = 0.7 for F-CF and at T = 0.8 for MAE-CF. Furthermore, all values of the proposed model are better than traditional User-CF. To sum up, the combination of low error rate and high F-measure are indeed desirable when we choose the similarity threshold because it improves the utility and usefulness of the system.

Finally, Table 3 provides the value of the optimal weight vector for F-CF and MAE-CF. Notice that the most important contextual factors to decrease MAE are time and gender while age, time, and genre is the best set in the case of F-measure. Interestingly, the relevant context can strongly vary in the same domain application. It will be very interesting to dynamically select between different contextual data in order to detect the specific context situation and adapt it to each user.

We can conclude from different tests conducted in this work that correct predictions generated by our model are significantly better than that suggested by Gasmi et al. [9]. In fact, it is crucial to take into consideration several context features with their degree of importance in order to obtain satisfactory results. However, the choice of contextual factors must be made according to the application domain. In addition, it will be very useful to apply a switch mechanism to select relevant contextual data in the same application. Moreover, the value of the similarity threshold must be chosen carefully, and its tuning strongly depends on the performance measurement.
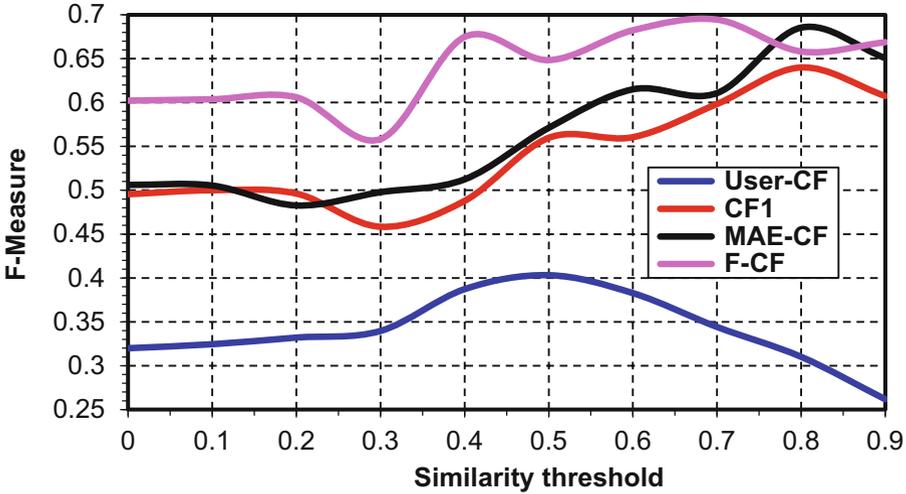
Fig. 5. F-measure with different similarity threshold

Table 3. Optimal weight vector

	MAE-CF	F-CF
w_s	0.5003	0.0082
w_a	0.00000005	0.2891
w_o	0.00004	0.0339
w_t	0.4995	0.2115
w_g	0.000000003	0.4570

6   Conclusion

This study proposes a context-aware collaborative filtering method based on an evolutionary algorithm to directly incorporate contextual information in the recommendation process. Our model does not increase the sparsity problem because the number of ratings is no longer restricted to a given context. Nevertheless, we introduce a weighting function that captures the context of each rating, which will be used to compute predictions. The proposed weighting function takes into account the similarities in terms of gender, age group and occupational category between the active consumer and his neighborhood. It also favors the most recent evaluations of users and the genres of items that are frequently consulted and well rated. In addition, the proposed model employs the GA-based meta-heuristic to assign the optimal weights for contextual factors. The values and weights of the contextual characteristics are then combined to calculate predictions.

Experiments on Movielens data set, shows the effectiveness of the proposed method to improve predictions. In a future work, we intend to extend the proposed model by considering other contextual information and proposing a dynamic context selection mechanism.

References

1. Dixit, V.S., Jain, P.: Recommendations with sparsity based weighted context framework. In: Gervasi, O., Murgante, B., Misra, S., Stankova, E., Torre, C.M., Rocha, A.M.A.C., Taniar, D., Apduhan, B.O., Tarantino, E., Ryu, Y. (eds.) Computational Science and Its Applications – ICCSA 2018, pp. 289–305. Springer, Cham (2018)
2. Campos, P.G., Díez, F., Cantador, I.: Time-aware recommender systems: a comprehensive survey and analysis of existing evaluation protocols. User Model. User-Adapt. Interact. **24**, 67–119 (2013)
3. Abowd, G.D., Dey, A.K., Brown, P.J., Davies, N., Smith, M., Steggles, P.: Towards a better understanding of context and context-awareness. In: Gellersen, H.-W. (ed.) Handheld and Ubiquitous Computing, pp. 304–307. Springer, Heidelberg (1999)
4. Adomavicius, G., Mobasher, B., Ricci, F., Tuzhilin, A.: Context-aware recommender systems. AI Mag. **32**, 67–80 (2011)
5. Rao, V., RosniK, KV., Padmanabhan, V.: Divide and transfer: understanding latent factors for recommendation tasks. In: RecSysKTL, Italy, pp. 1–8 (2017)
6. Levandoski, J.J., Sarwat, M., Eldawy, A., Mokbel, M.F.: LARS: a location-aware recommender system. In: 2012 IEEE 28th International Conference on Data Engineering, pp. 450–461 (2012)
7. Xu, G., Tang, Z., Ma, C., Liu, Y., Daneshmand, M.: A collaborative filtering recommendation algorithm based on user confidence and time context. J. Electr. Comput. Eng. **2019**, 1–12 (2019)
8. Zheng, Y.: A revisit to the identification of contexts in recommender systems. In: International Conference on Intelligent User Interfaces, pp. 109–115 (2015)
9. Gasmi, I., Seridi-Bouchelaghem, H., Hocine, L., Abdelkarim, B.: Collaborative filtering recommendation based on dynamic changes of user interest. Intell. Decis. Technol. **9**, 271–281 (2015)
10. Kim, K., Ahn, H.: A recommender system using GA K-means clustering in an online shopping market. Expert Syst. Appl. **34**, 1200–1209 (2008)
11. Bobadilla, J., Ortega, F., Hernando, A., Alcalá, J.: Improving collaborative filtering recommender system results and performance using genetic algorithms. Knowl. Based Syst. **24**, 1310–1316 (2011)
12. Alhijawi, B., Kilani, Y.: Using genetic algorithms for measuring the similarity values between users in collaborative filtering recommender systems. In: 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS), Okayama, Japan, pp. 1–6 (2016)
13. Idrissi, N., Zellou, A.: A systematic literature review of sparsity issues in recommender systems. Soc. Netw. Anal. Min. **10**(1), 1–23 (2020). https://doi.org/10.1007/s13278-020-0626-2
14. Jain, A.F., Vishwakarma, S.K., Jain, P.: An efficient collaborative recommender system for removing sparsity problem. In: Fong, S., Dey, N., Joshi, A. (eds.) ICT Analysis and Applications, pp. 131–141. Springer, Singapore (2020)
15. Zhang, H., Ni, W., Li, X., Yang, Y.: Modeling the heterogeneous duration of user interest in time-dependent recommendation: a hidden semi-Markov approach. IEEE Trans. Syst. Man Cybern. Syst. **48**, 177–194 (2018)

16. Salehi, M., Pourzaferani, M., Razavi, S.A.: Hybrid attribute-based recommender system for learning material using genetic algorithm and a multidimensional information model. Egypt. Inform. J. **14**, 67–78 (2013)
17. Peška, L., Tashu, T.M., Horváth, T.: Swarm intelligence techniques in recommender systems - a review of recent research. Swarm Evol. Comput. **48**, 201–219 (2019)
18. Al-Shamri, M.Y.H., Bharadwaj, K.K.: Fuzzy-genetic approach to recommender systems based on a novel hybrid user model. Expert Syst. Appl. Int. J. **35**, 1386–1399 (2008)
19. Karatzoglou, A., Amatriain, X., Baltrunas, L., Oliver, N.: Multiverse recommendation: n-dimensional tensor factorization for context-aware collaborative filtering. In: Proceedings of the Fourth ACM Conference on Recommender Systems - RecSys 2010, Barcelona, Spain, pp. 79–86. ACM Press (2010)

A Rule Based Human Skin Detection Method in CMYK Color Space

Abdelkrim Sahnoune$^{(\boxtimes)}$, Djamila Dahmani , and Saliha Aouat

Artificial Intelligence Laboratory (LRIA), Computer Science Department, University of Science and Technology Houari Boumediene (USTHB), Algiers, Algeria
{asahnoune,ddahmani,saouat}@usthb.dz

Abstract. Skin detection is a very important task in computer vision, since we can find it in many applications such as face detection and recognition, face tracking, gesture analysis, content-based image retrieval systems and human machine interaction systems. In this paper we present a novel rule-based skin detection method in the Cyan Magenta Yellow Key (CMYK) color space. This space is a subtractive color space used in color printing, and poorly explored in image processing and still less in skin detection tasks. Our method uses thresholds which are based on the relation between CMYK color components in order to recognize skin pixels, two thresholding models were proposed and we have considered the most performing one. The proposed method has been tested on two public skin image databases and has achieved very satisfactory qualitative and quantitative results against other widely used rule-based methods.

Keywords: Skin detection · Rule-based method · CMYK color space

1 Introduction

Skin detection can be defined as pixels classification into skin pixels and non-skin pixels. This task is very important since we can find it in many computer vision applications such as face detection and recognition, video surveillance, gesture analyse and human machine interaction systems to name but a few.

Due to its simplicity and effectiveness, its non-sensitivity to: rotation, scaling and partial occlusions. Color is the most used attribute in skin detection methods. However choosing this attribute presents some drawbacks like:

– Sensibility to luminance variation and camera characteristics.
– Color variation depending on ethnicity and individual characteristics.
– Complex backgrounds which means the existence within the input (image, video) of some objects with skin-like color such as wood and sand.

These drawbacks lead to erroneous detection of skin segment. In particular, complex backgrounds increase the false positive rate *i.e.* non-skin regions

detected as skin regions. Hence, other attributes can be associated with color such as texture, form and motion.

Skin detection methods can be classified into machine-learning methods and rule-based methods. We can also find hybrid methods which combine the two classes mentioned before. Principal skin detection works are presented in [7,13] and [16]. The major difference between machine-learning-based and rule-based methods lies in the fact that unlike the first class, the second class does not require a training data set. Machine-learning methods provide better results than rule based methods, but as mentioned above they require a training data set and that can be considered as a drawback, learning and classification time can also be considered as a disadvantage especially in real time applications.

Since we are aiming to have a fast, simple and effective solution that does not depend on any dataset and can be used in real-time applications we opted for a rule-based method. Therefore, we propose in this paper an explicit method in a color space rarely explored in computer vision namely the Cyan Magenta Yellow Key CMYK color space in order to accomplish the skin detection. This method is based on thresholds resulting from fitting the CMYK space. For this purpose we have developed two models and considered the most performing one.

The experimentations we have done showed that our method allows us to have good results both in terms of detection rate and execution time. And so it is suitable for skin detection related applications particularly interaction applications.

The remaining part of the paper will be organized as follows, Sect. 2 describes briefly the related works, in Sect. 3 we present our method, experimental results are reported in Sect. 4 and a conclusion is given in the Sect. 5.

2   Related Works

As mentioned in the Introduction color is the most used attribute in skin detection, hence it is very important to start with selecting an adequate color space. Different color spaces were used to this aim and we note that RGB, HSV and YCbCr are the most found in the literature. Excluding RGB, the other spaces are issued from linear or nonlinear transformations of the first one. And contrary to RGB, in the latter spaces chrominance and luminance are separated which improves the distinction between skin and non-skin colors. There are two major classes of skin detection methods which are the rule-based and machine-learning based classes.

In the Rule-based methods (also known as explicit methods or thresholding methods) we try to generate rules related the color distribution in the selected color space, since skin colors occupy a little place in the color space and tend to cluster in one region. Rule-based methods use color only as attribute and go from a simple one component thresholding to a more or less complex rules. A simple thresholding in different spaces is used in [8] and [19] for the RGB space, [1] and [12] for the HSV space and [2] and [4] for the YCbCr space. In [3] a constructive induction algorithm is used to construct a three component decision rules from

normalized RGB components through simple arithmetic operations and in [6] a heuristic rules in the RGB color space are used to detect skin regions. In [18] they estimated correlation rules between YCb and YCr sub spaces to perform the skin detection relying on the calcul of some distances and thresholds. Combining more than one space is also common in skin detection methods. [10] used both YCbCg and YCgCr spaces which are variation of YCbCr while [14] used an RGB + Cr (from YCbCr) thresholding. [19] combined RGB,HSV and YCbCr to achieve the skin detection. We note also that many works proposed a dynamic method in RGB where the thresholds change according to the image [15].

Machine-learning methods differ from the first ones by the use of training data sets of both skin and non-skin elements in order to generate models capable of completing the skin detection task. This class can be divided into two categories: Non-parametric methods, these methods estimate the distributions of skin colors by probability calculations made on color histograms constructed from the training data. the most popular and best-performing method based on histograms is proposed in [5]. In which they create probability lookup tables from RGB histogram built from skin and non-skin data sets and use Bayes rules to calculate the probabilities of being a skin or non-skin entity for a given pixel. The classification is achieved by thresholding the probabilities. In this same category we also have the artificial neural networks in which we generally associate texture information to color information [17]. Parametric methods, they use probabilistic Gaussian models to represent the distribution of skin colors in color spaces and thus predict the class of each pixel. We can have models using one Gaussian which are called Single Gaussian Models SGM and other models called Gaussian mixture models GMM that use more than one Gaussian. These methods are widely used since they provide good results with significantly less training data [9,15].

Finally we have hybrid methods that combining both rule-based and machine-learning methods such as the work of [11].

3   The Proposed Method

Since our method uses color as a unique attribute, our first step was to choose the right color space. So we opted for CMYK (Cyan, Magenta, Yellow, Key -black-) a color mode used to represent colors in the RGB space. The idea behind the use of this mode of color is of artistic and industrial inspiration. In effect, contrary to the default space used in digital images RGB which considers that the main colors are red, green and blue. In art and printing the primary colors are Cyan, Magenta and Yellow; as well as black, whose addition allows for darker colors depending on its value. This can be seen as a separation between the luminance: K and the chrominance: CMY like the YCbCr and HSV spaces, which justifies our choice to adopt this color space.

The first use of CMYK in industry dates back to 1906 when the Eagle Printing Ink Company first used four colors (CMYK) for printing, discovering that these four colors can be combined to produce an almost unlimited number of tones [22]. Since then, this system has been adopted in all printers.

A CMYK image is visually identical to an RGB image. The difference lies, as mentioned above, in the use of different primary colors. So, to represent a given color we will have a different distribution for RGB and CMYK. Figure 1 shows this different distribution for the same colour.

Fig. 1. CMYK and RGB color code for the same color.

Figure 2 represents the CMYK color wheel which gives a global view of the different colors with their CMYK distribution.
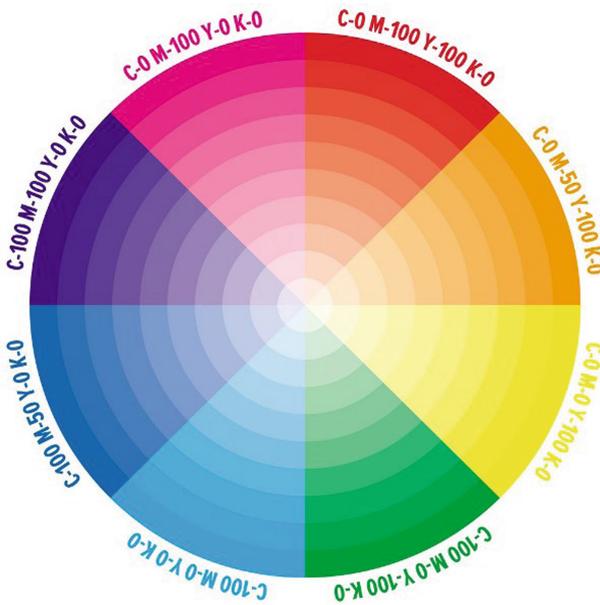

Fig. 2. CMYK color wheel.

By observing the Color Wheel, we can notice that the non-skin colors have a value of C different from zero. So just by applying this simple thresholding we will eliminate a large part of the non-skin objects this step also justifies the choice of this mode. To confirm this we also tried to do the same with the default space RGB by considering the condition $B = 0$ as Cyan is a variant of blue and the results were largely unsatisfactory.

Our method starts by converting the input image from the RGB space to the CMYK space. The conversion is done according to the following formulas [23]:

$$R' = \frac{R}{255} \tag{1}$$

$$G' = \frac{G}{255} \tag{2}$$

$$B' = \frac{B}{255} \tag{3}$$

$$K = 1 - max(R', G', B') \tag{4}$$

$$C = \frac{1 - R' - K}{1 - K} \tag{5}$$

$$M = \frac{1 - G' - K}{1 - K} \tag{6}$$

$$Y = \frac{1 - B' - K}{1 - K} \tag{7}$$

Thus, we will have values between 0 and 1 corresponding to percentages. To obtain value in [0.255] interval, we will just multiply them by 255. After the conversion we apply a threshold for each pixel in order to perform the skin classification. This threshold includes, as mentioned before, the condition $C = 0$ to eliminate non skin-like colors. We must also take into account objects with colors similar to skin colors, *i.e.* pixels with $C = 0$ but which are not skin pixels. To solve this problem we have proposed two thresholding models for the three remaining components M Y and K.

In the first one noted $M1$, we were inspired by the basic threshold methods. Mainly the most used one, the RGB model [19]. So model $M1$ was conceived after intense experimentation that followed an analysis of the CMYK color wheel Fig. 2 as well as the analysis of many skin images. The aim of the latter was to find a relationship between the MYK components that allows distinguishing skin pixels from non-skin pixels. This work allowed us to come out with a condition discriminating skin and non-skin regions and which we confirmed by experimentation. At the end of the experimentation the optimal quantitative values were obtained by applying the following threshold:

$$C = 0 \; and \; M > 19 \; and \; M - Y < 33 \tag{8}$$

For this first model the experimentation showed that the value of K does not influence the results, which led us to ignore this component.

In the second model noted $M2$ we wanted to rely on statistical measures in order to have more reliability regarding our method. Hence, we collected 60 skin images with different light conditions and with complex backgrounds and whose skin masks are available. Which gives us about $3,35 \times 10^6$ skin tone pixels, in order to represent the distribution of skin colours in the CMYK space. However, since the CMYK space is 4-Dimensional, we considered the 2D subspaces of CMYK to be able to perform the analysis. Figure 3 shows the 2D projections of CMYK subspaces, skin colors distribution in each subspace is represented with a blue cluster.

(a) C-M subspace

(b) C-Y subspace

(c) C-K subspace

(d) M-K subspace

(e) Y-K subspace
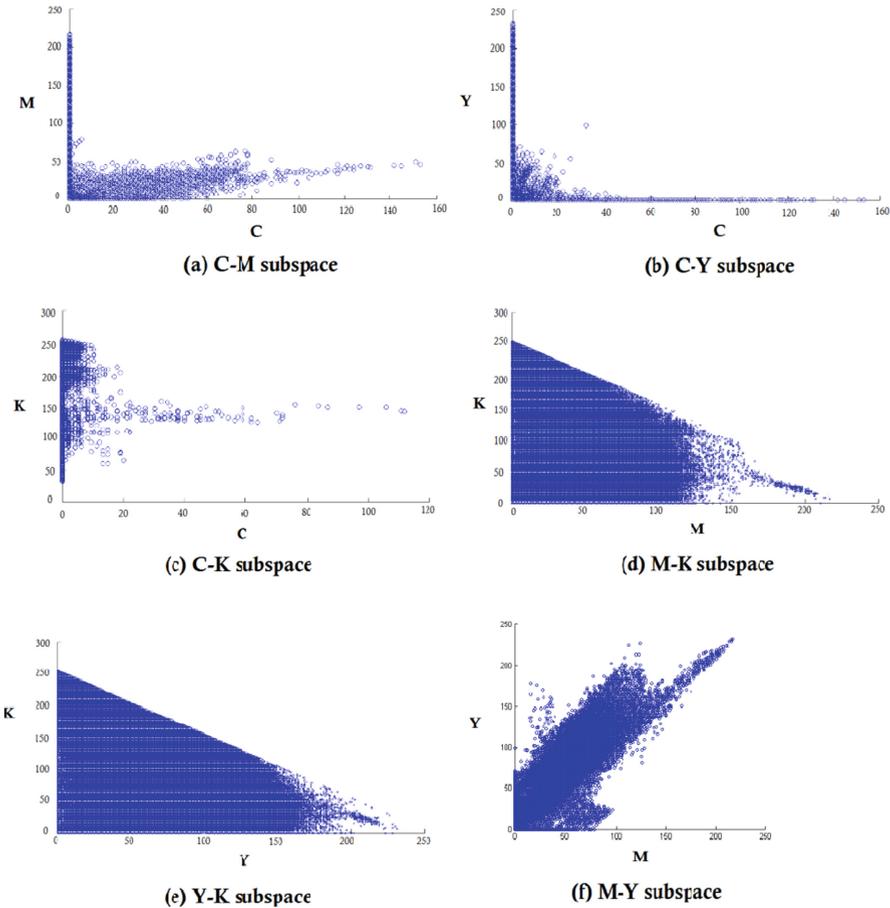
(f) M-Y subspace

Fig. 3. Skin colors distribution in CMYK subspaces: (a) C-M, (b) C-Y, (c) C-K, (d) M-K, (e) Y-K, (f) M-Y. Blocs in blue represent skin colors clusters.
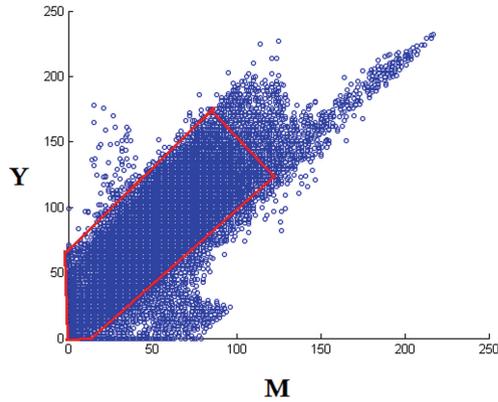
Fig. 4. M-Y subspace: skin color cluster represented in blue, the dense region highlighted with red rectangle.

By observing the subspaces in which the C component is involved we can see that the majority of skin pixels have a $C = 0$. Indeed, on the $3,35 \times 10^6$ skin pixels the number of pixels having $C \neq 0$ does not exceed 3×10^4 and this confirms what we mentioned above. From Fig. 3 we can see on the skin colors cluster in the MY subspace, especially the dense region highlighted in Fig. 4 that the difference between the components M and Y is practically constant and can be thus bounded by a threshold that we have determined graphically as being $0 < Y - M < 80$. We can also notice from this dense region that the value of Y is always greater or equal to the value of M. From Fig. 4 as well, we can notice that the value of M for skin pixels are limited and belongs to the interval $[0, 120]$, this was also been taken into account for the thresholding. Finally, the relationship between the skin colors and the K was not drawn from the graphs. It was deduced by experimentation. We note that we have started experimenting with thresholding directly issued from the graphs *i.e*:

$$C = 0 \ and \ M > 0 \ and \ M < 120 \ and \ Y - M < 80 \ and \ Y \geq M \qquad (9)$$

These values were modified after experimentation according to the results obtained. Thus, after experimentation a pixel is classified as a skin pixel if it meets the following condition:

$$C = 0 \ and \ M > 30 \ and \ M < 97 \ and \ K < 160 \ and \ Y - M < 65 \ and \ Y \geq M \ (10)$$

Algorithm 1 summarizes the steps of the proposed method.

Algorithm 1: Our skin detection algorithms

Result: segmented image (skin and non-skin regions)
read the input image **i**;
convert **i** from RGB to CMYK;
for *each pixel **p** in **i*** **do**
    if *the CMYK components satisfy the threshold** **then**
        p = skin pixel;
    else
        p = non-skin pixel;
    end
end
* (8) for model $M1$ and (10) for model $M2$

---

4   Experimental Results

To prove its effectiveness, our method was tested on different datasets and compared to other rule-based skin detection methods. First, we carried out a comparison between our two models $M1$ and $M2$. These two models were also compared to the method proposed in [19] which is the most common threshold skin detection method. The latter was used as a reference to validate our results. Skin detection results are presented as binary images where white segments represent skin regions and black segments represent non-skin regions. Figure 5 shows some qualitative results for the three methods and Fig. 6 illustrates the region covered by the threshold of each model ($M1$ and $M2$) in the MY subspace.
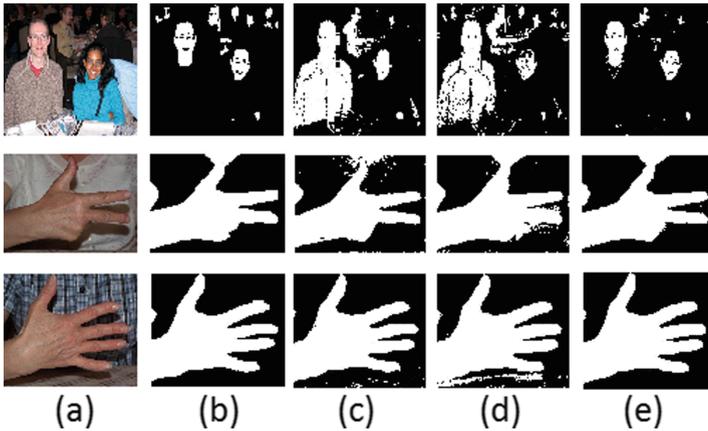
Fig. 5. Skin detections results: (a) input images; (b) skin mask; (c) RGB thresholding - S. Kolkur et al.; (j) proposed method M1; (k) proposed method M2.
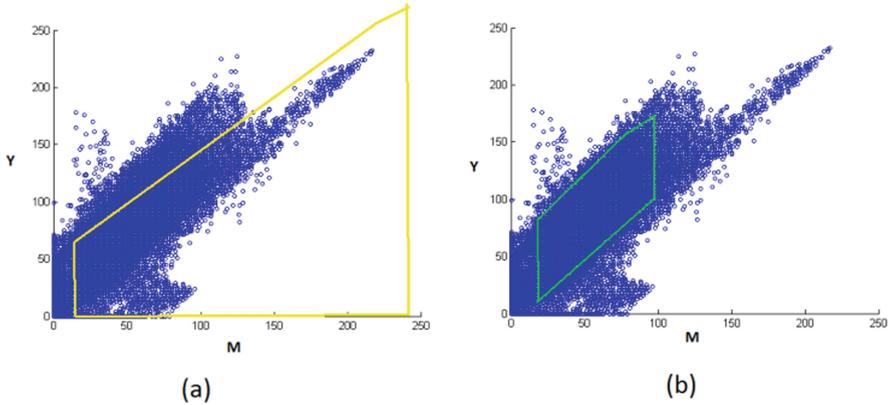
Fig. 6. M-Y subspace: region covered by thresholds (a) M1 model; (b) M2 model.

From Fig. 5 we can notice a difference in the results of $M1$ and $M2$. In model $M2$ the distinction between skin and non-skin regions is nearly perfect, contrary to model $M1$ where in addition to skin area parts of clothes were also detected incorrectly as skin. $M1$ threshold covers a large part of the MY subspace (see Fig. 6(a)) and encompasses the skin-colors clusters and the other colors. On the other hand, $M2$ threshold (see Fig. 6(b)) covers only the colors belonging to the skin-colors cluster. This explains the inequality in the two models results and proves that $M2$ is more performing. Figure 5 shows also that the results obtained by model $M2$ were better than those obtained by the method proposed in [19], which validates our choice of the color space and the threshold model.

Afterwards, our method was compared to other explicit methods and we have selected the most widely used in the literature: the methods proposed in Sobotka and pitas [1], Chai and Ngan [2], Gomez and Morales [3], Hsu et al. [4] Kovac et al. [6], Brancati et al. [18] and the RGB threshold used in Kolkurl et al. [19]. This choice is made because we wanted to test it against methods of the same class *i.e.* rule-based methods that use color as single attribute.

The different methods were tested on two data sets that are publicly available: The database for hand gesture recognition HGR1 [20], which contains 899 hand images, taken in different light conditions and containing images with skin-like objects. And the Human skin detection dataset (Pratheepan data set) [21], which contains 78 skin images containing several images with complex background. The evaluation was done on a simple PC, equipped with an $IntelCorei3 - 3217U$ CPU at $1.8\,$GHz and with $4\,$GB RAM using MatLab R2013a. For a 640×480 resolution image the execution time of our method is $245\,$ms, which is very satisfying considering the configuration of the machine used.

Some qualitative results of the different methods are illustrated in Fig. 7 for the two datasets and in Fig. 8 for real images.

Fig. 7. Skin detection results in the two used datasets: HGR1 and Pratheepan; (a) input image; (b) skin mask; (c) Sobottka and Pitas; (d) Chai and Ngan; (e) Hsu et al.; (f) gomez et morales; (g) kovac et al.; (h) brancatti et al.; (i) RGB thresholding - S. Kolkur et al.; (j) proposed method M1; (k) proposed method M2.

Figure 7 and Fig. 8 shows that our method (model $M2$) qualitatively surpasses the other methods. In fact, in the other methods we have either strict rules that by willing to eliminate all non-skin pixels do not entirely detect the skin regions. Or tolerant rules that incorrectly classify non-skin regions while trying to ensure perfect skin regions detection. In our method we tried to have a fair compromise between the two. We can also notice that the other methods are irregular, as they do not provide good results for all images.

Fig. 8. Skin detection results on real images; (a) input image; (b) Sobottka and Pitas; (c) Chai and Ngan; (d) Hsu et al.; (e) gomez et morales; (f) kovac et al.; (g) brancatti et al.; (h) RGB thresholding - S. Kolkur et al.; (i) proposed method M1; (j) proposed method M2.

To make a quantitative evaluation of the proposed method and thus test its effectiveness we used the following measurements: Precision, Recall, Specificity, F-measure and D_{prs}. These measures are calculated as follows:

$$Precision = \frac{TP}{TP + FP} \tag{11}$$

$$Recall = \frac{TP}{TP + FN} \tag{12}$$

$$Specificity = \frac{TN}{TN + FP} \tag{13}$$

$$F - measure = \frac{2 \times Precision \times Recall}{Recall + Precision} \tag{14}$$

$$D_{prs} = \sqrt{(1 - Precision)^2 + (1 - Recall)^2 + (1 - Specificity)^2} \tag{15}$$

Where TP is the number of pixels correctly classified as skin pixels, TN the number of pixels correctly classified as non-skin pixels, FP the number of pixels wrongly classified as skin pixels, and as FN the number of pixels wrongly classified as non-skin pixels. Among the five measurements considered, we note that the F-measurement takes into account the Precision and the Recall. The D_{prs}, which represents the Euclidean distance between the resulting image and

the Skin mask, takes into account: Precision, Recall and Specificity which represents a general measurement of the effectiveness of the method. Hence, these two criteria are the most important and representative of the quality of skin detection methods. We note that the values of these five measurements are to be maximized, except for the D_{prs}, which is to be minimized. The values of the measurements for the different tested methods are presented in Table 1. As far as our method is concerned, we notice that the values of the comparison measurements differ according to the thresholds considered. Our objective was to find a compromise allowing to have the highest possible TPR and TNR and at the same time the lowest possible FPR and FNR which allows us to have the best measurements values. So in Table 1 we present the results obtained by using the thresholds mentioned in the algorithm, with the two models $M1$ and $M2$.

Table 1 confirms quantitatively that model $M2$ is better than model $M1$. Indeed, although model $M1$ provides very good results on the HGR1 database with even the best F-measure of all methods. The results of model $M1$ were not as good when tested on the Pratheepan database. So we have chosen model $M2$ which was satisfying on both dataset for the comparison with the other methods.

Also from Table 1 we notice that our method (model $M2$) outperforms the other methods with which it was compared. Indeed, in terms of the two most important measures: F-measure and D_{prs} our method achieves the best result whether on HGR1 or Pratheepan datasets. Concerning the other measurements; our method obtains the best Precision in the two bases HGR1 and Pratheepan. Our method obtains the second best Recall on Pratheepan and the fourth best *Recall* on HGR1 with a difference of only 0.05 from the first. Finally, the Specificity of our method is the best on HGR1 and fourth on the Pratheepan dataset, with a considerable difference from the first. Thus contrary to the other methods which have inconsistent results on the two data sets, our method is always the best or at least among the best. And it is only for the last case that our method is a little far from the best ones. We note that we obtained during the experimentation of the Pratheepan database a Specificity value 0.1 higher than the one presented in Table 1 but with a drop in Precision and F-measure. This led us to keep the threshold presented in this paper, but also because we didn't want to take different thresholds for the two data bases.

Finally, from Fig. 7, Fig. 8 and Table 1 we can observe that model $M1$ provide very good results in HGR1 data set, but less good results in Pratheepan data set. Contrary to $M2$ model which is effective on both bases. This shows that the CMYK color space is the most suitable color space for rule-based skin detection methods provided that we choose the right threshold.

Table 1. Qualitative comparison between the selected rule-based methods and the proposed method

Measurement	Method	HGR1	Pratheepan
Precision	Sobotka and pitas	0.6477	0.4186
	Chai and Ngan	0.7731	0.3852
	Hsu et al.	0.7322	0.5215
	Gomez and Morales	0.8664	0.4121
	Kovac et al.	0.8335	0.5849
	Brancatti et al.	0.8714	0.5513
	Kolkurl et al. (RGB threshold)	0.8373	0.6060
	Proposed method M1	0.8807	0.5061
	Proposed method M2	**0.9103**	**0.6596**
Recall	Sobotka and pitas	0.8577	0.7808
	Chai and Ngan	**0.9599**	**0.9494**
	Hsu et al.	0.9229	0.9009
	Gomez and Morales	0.8149	0.7032
	Kovac et al.	0.8561	0.6258
	Brancatti et al.	0.9062	0.8199
	Kolkurl et al. (RGB threshold)	0.8601	0.9189
	Proposed method M1	0.9489	0.9022
	Proposed method M2	0.9065	0.9214
Specificity	Sobotka and pitas	0.5759	0.6844
	Chai and Ngan	0.7336	0.5781
	Hsu et al.	0.6928	0.7702
	Gomez and Morales	0.9236	0.6595
	Kovac et al.	0.8758	**0.8955**
	Brancatti et al.	0.8916	0.8230
	Kolkurl et al. (RGB threshold)	0.8777	0.6298
	Proposed method M1	0.8870	0.5948
	Proposed method M2	**0.9287**	0.7027
F-Measure	Sobotka and pitas	0.7381	0.5450
	Chai and Ngan	0.8565	0.5480
	Hsu et al.	0.8166	0.5585
	Gomez and Morales	0.8399	0.5197
	Kovac et al.	0.8447	0.6047
	Brancatti et al.	0.8885	0.6592
	Kolkurl et al. (RGB threshold)	0.8485	0.7304
	Proposed method M1	**0.9135**	0.6291
	Proposed method M2	0.9081	**0.7688**
D_{prs}	Sobotka and pitas	0.5693	0.7020
	Chai and Ngan	0.3521	0.7508
	Hsu et al.	0.4147	0.6414
	Gomez and Morales	0.2406	0.7413
	Kovac et al.	0.2525	0.5780
	Brancatti et al.	0.1924	0.5149
	Kolkurl et al. (RGB threshold)	0.2469	0.5466
	Proposed method M1	0.1720	0.6609
	Proposed method M2	**0.1482**	**0.4587**

5 Conclusion

In this paper we have presented a new rule-based skin detection method in the CMYK color space. Our method uses thresholds which are based on the relation between CMYK color component in order to recognize skin pixels. The proposed method has been tested on two skin image datasets and provided highly satisfying results when compared to other widely used explicit methods.

In the future works we will focus in improving the results especially the specificity. Mainly by combining thresholding models, since we have obtained better results in some cases using different thresholds. We also aim to add a more efficient decision method and combine the texture information to our color-based skin-detection model.

References

1. Sobottka, K., Pitas, I.: A novel method for automatic face segmentation, facial feature xtraction and tracking. Sig. Process. Image Commun. **12**(3), 263–281 (1998)
2. Chai, D., Ngan, K.N.: Face segmentation using skin-color map in videophone applications. IEEE Trans. Circ. Syst. Video Technol. **9**(4), 551–564 (1999)
3. Gomez, G., Morales, E.: Automatic feature construction and a simple rule induction algorithm for skin detection. In: Proceedings of Workshop on Machine Learning in Computer Vision, pp. 31–38 (2002)
4. Hsu, R.-L., Abdel-Mottaleb, M., Jain, A.K.: Face detection in color images. IEEE Trans. Patt. Anal. Mach. Intell. **24**(5), 696–706 (2002)
5. Jones, M.J., Rehg, J.M.: Statistical color models with application to skin detection. Int. J. Comput. Vis. **46**(1), 81–96 (2002)
6. Kovac, J., Peer, P., Solina, F.: Human skin color clustering for face detection. In: EUROCON. IEEE (2003)
7. Kakumanu, P., Makrogiannis, S., Bourbakis, N.: A survey of skin-color modeling and detection methods. Patt. Recogn. **40**, 1106–1122 (2007)
8. Lee, Y.J., Lee, D.H.: Research on detecting face and hands for motion-based game using web camera. In: Proceedings of the 2008 International Conference on Security Technology, vol. 4, pp. 7–12. SecTech (2008). https://doi.org/10.1109/SecTech.2008.14
9. Cheddad, A., Condell, J., Curran, K., Kevitt, P.M.: A skin tone detection algorithm for an adaptive approach to steganography. Sig. Process. **89**(12), 2465–2478 (2009)
10. Zhang, Z.Z., Shi, Y.X.: Skin color detecting unite YCgCb color space with YCgCr color space. In: Proceedings of 2009 International Conference on Image Analysis and Signal Processing, IASP 2009, pp. 221–225 (2009)
11. Zaidan, A., Ahmed, N., Karim, H.A., Alam, G.M., Zaidan, B.: Increase reliability for skin detector using backprobgation neural network and heuristic rules based on YCbCr. Sci. Res. Essays **5**(19), 2931–2946 (2010)
12. Guo, J.-M., Liu, Y.-F., Chang, C.-H., Nguyen, H.-S.: Improved hand tracking system. IEEE Trans. Circ. Syst. Video Technol. **22**(5), 693–701 (2012)

13. Prema, C., Manimegalai, D.: Survey on skin tone detection using color spaces. Int. J. Appl. Inf. Syst. (IJAIS) **2**(2), 18–26 (2012). ISSN: 2249-0868
14. Prema, C., Manimegalai, D.: A novel skin tone detection using hybrid approach by new color space. Int. J. Comput. Appl., 975–8887 (2012)
15. Yogarajah, P., Condell, J., Curran, K., McKevitt, P., Cheddad, A.: A dynamic threshold approach for skin tone detection in colour images. Int. J. Biom. **4**(1), 38–55 (2012)
16. Kawulok, M., Nalepa, J., Kawulok, J.: Skin detection and segmentation in color images. In: Advances in Low-Level Color Image Processing. Lecture Notes in Computational Vision and Biomechanics, vol. 11, pp. 329–366. Springer, Heidelberg (2014)
17. Al-Mohair, H.K., Saleh, J.M., Suandi, S.A.: Hybrid human skin detection using neural network and k-means clustering technique. Appl. Soft Comput. **33**, 337–347 (2015)
18. Brancati, N., De Pietro, G., Frucci, M., Gallo, L.: Human skin detection through correlation rules between the YCb and YCr subspaces based on dynamic color clustering. Comput. Vis. Image Underst. **155**, 33–42 (2017)
19. Kolkurl, S., Kalbande, D., Shimpi, P., Bapat, C., Jatakia, J.: Human skin detection using RGB, HSV and YCbCr color models (2017). https://doi.org/10.2991/iccasp-16.2017.51
20. Database for hand gesture recognition. http://sun.aei.polsl.pl/~mkawulok/gestures/
21. Human skin detection dataset. http://web.fsktm.um.edu.my/~cschan/downloads_skin_dataset.html
22. CMYK history, 2 May 2016. https://www.clubink.ca/blog/print/history-behind-cmyk-colour-model/. Accessed 17 Feb 2020
23. RGB to CMYK conversion, 1 February 2020. https://www.101computing.net/cmyk-to-rgb-conversion-algorithm/. Accessed 9 Feb 2020

Improved NSGA-II for Minimum Weight Minimum Connected Dominating Set Problem

Hayet Dahmri[(✉)] and Salim Bouamama

Department of Computer Science, University of Ferhat Abbas - Sétif 1,
19000 Sétif, Algeria
{hayet.dahmri,salim.bouamama}@univ-setif.dz

Abstract. Most real-world problems are multiobjective in nature and considerable research efforts have been devoted to propose efficient multiobjective optimization approaches. Nondominated sorting genetic algorithm II (NSGA-II) is one of the well-known algorithms for this purpose which is based on a fast nondominated sorting procedure and an elitist selection strategy. This paper presents an improved NSGA-II algorithm (I-NSGA-II) based on greedy heuristics to tackle the minimum weight minimum connected dominating set problem. To make a trade-off between the size of the connected dominating set and its total weight, two objectives are considered, namely the minimization of the size and the minimization of the total edge-weight. The performance of I-NSGA-II is evaluated on a set of test problem instances with different sizes. Computational experiments show a significant improvement of our approach over NSGA-II with respect to the hypervolume indicator, run-time, and quality of solutions.

Keywords: Minimum connected dominating set · Minimum weight connected dominating set · Multiobjective optimization · Non-dominated sorting genetic algorithm II · Greedy heuristic · Local search

1 Introduction

Given a simple undirected graph $G = (V, E)$, where V is the set of vertices and E is the set of edges, a dominating set (DS) is a subset D of V such that each vertex not in D has at least one neighbor in D. If its induced subgraph $G(D)$ is connected then it represents a connected dominating set (CDS). A CDS of minimum cardinality is called minimum connected dominating set (MCDS). If G is a vertex-weighted graph, a minimum weight connected dominating set (MWCDS) is a CDS with minimum total weight. The two later concepts have utility in several application areas including wireless sensor networks, Mobile ad-hoc networks, and wireless vehicular ad-hoc networks [1,4,11,17]. They are useful

for location-based routing, topology control and energy conservation. Other fields of application can be found in optical networks [15,16] and biological systems [10,12].

The minimum weight minimum connected dominating set (MWMCDS) problem is a bi-objective combinatorial optimization problem defined on an undirected edge-weighted graph and was firstly introduced by Rengaswamy et al. [13]. The aim of this problem is to obtain a trade-off CDS that achieves both cardinality reduction and total edge-weight minimization.

In this work, we propose to solve the MWMCDS problem with an improved Pareto genetic algorithm based on NSGA-II refereed as I-NSGA-II. NSGA-II is a well known, fast sorting, and elite multi-objective genetic algorithm proposed by Deb et al. [7]. It has been successfully applied for solving several real-world multiobjective optimization problems such as, among others, fault diagnosis in power system [19], weighted clique problem [3] and task scheduling problems [14,18]. The original NSGA-II is enhanced by using greedy heuristics for generating the initial population of solutions. In addition, neighbors are obtained by applying either genetic algorithm or local search method. Genetic algorithm generates neighbors randomly which extend the solution space, and the local search method concentrate on obtaining better neighbors based on greedy heuristic which enhance the quality of fronts solutions. Nondominated sorting phase and crowding distance function as developed in the original NSGA-II are employed to obtain better spread of pareto-optimal solutions of non-dominated solutions.

The remaining part of this paper is arranged as follows. Section 2 briefly provides the most relevant studies related to our work. Section 3 models the problem. Section 4 describes the original NSGA-II. Section 5 presents greedy heuristics. The proposed algorithm is presented in Sect. 6. Experimental results are discussed in Sect. 7. Section 8 concludes the paper.

2   Related Work

From the view of complexity theory, both MCDS problem and MWCDS problem have the common feature that belong to the class of NP-hard problems [9] and most available researches on this subject resort to use approximate approaches to solve them efficiently. Example of such approach include Ant Colony Optimization (ACO) [2], Genetic Algorithm (GA) [5], Population Based Iterated Greedy algorithm (PBIG) [5] and Variable Neighborhood Search (VNS) [20].

However, most of them are generally focused on a single objective optimization problem, either to minimize the cardinality (case of MCDS problem) or to minimize the total weight (case of MWCDS problem). By contrast, very few ones are concerned with both objectives, that is, dealing with MWMCDS. Recently, Rengaswamy et al. [13] were the first to present the formulation of MWMCDS problem and solve it by using a multiobjective genetic algorithm (MOGA) based on scalarization technique (also known as the weighted sum method). To convert the bi-objective problem into a single objective one, each objective function is first normalized and multiplied by its corresponding weight. Accordingly, the

global single objective function is obtained by the summation of all weighted objective functions. Following the same technique, Dahmri and Bouamama [6] also recently presented a hybrid simulated annealing approach to solve this problem and showed that their approach has better performance than MOGA.

To the best of our knowledge, there are no attempts in the literature to study MWMCDS problem using Pareto optimality concept. In this paper, we propose an improved Pareto genetic algorithm based on NSGA-II for the MWMCDS problem.

3   The Minimum Weight Minimum Connected Dominating Set Problem

The MWMCDS problem is formally defined on a simple undirected edge-weighted connected graph $G = (V, E, w)$ where V represents the set of vertices and $E \subset V \times V$ represents the set of edges. w denotes the weight function $w : E \mapsto R^+$, with $w_{(u,v)} > 0$ for each edge $(u, v) \in E$. Recall that if two vertices are connected by an edge then they are called neighbors, we denote by $N(v)$ the set of neighbors of v. A subset $D \subseteq V$ is called a dominating set if each vertex $v \in V$ is either in D or adjacent to at least one vertex in D. If D is a dominating set and its induced subgraph $G(D)$ is connected, then D is called a connected dominating set. Vertices in the dominating set are called the dominators. This problem can be then formulated as follow:

minimize $\{F_c(D), F_w(D)\}$

subject to $\forall\, v \in V \backslash D : N(v) \cap D \neq \emptyset$,
$\qquad\qquad D \subseteq V$,
$\qquad\qquad G(D)$ is connected.

In the above definition, we look for a connected dominating set $D \subseteq V$ (a candidate solution) in which two objective functions are simultaneously minimized. Let $|D|$ represents the cardinally of D. The first objective function $F_c(D) := |D|$, named as the cardinality objective function, intends to minimize the size of the candidate solution while the second objective function $F_w(D)$ is calculated as follow.

$$F_w(D) := F_{w1} + F_{w2} \tag{1}$$

$$F_{w1} = \sum_{((u,v)\in E)\wedge(u\in D\wedge v\in D)} w_{(u,v)} \tag{2}$$

$$F_{w2} = \sum_{(u\in V\backslash D)} \mathbf{min}\{w_{(u,v)} \mid (u,v) \in E \wedge v \in D\} \tag{3}$$

$F_w(D)$ is named as the weight objective function, intends to minimize its total weight.

4   NSGA-II Algorithm

NSGA-II has been proven to be an efficient multiobjective optimization algorithm. It consists of two main phases: offspring production and new population selection. These two phases are repeated until termination criterion is satisfied either after a fixed number of generations or when no improvement is obtained after several generations.

4.1   Offspring Production

Offspring population (Q_t) is produced from parent population (P_t) using crossover and mutation operators. In crossover operator, two parents are selected randomly from P_t. Then, they are recombined using a standard recombination operator [8]. In mutation operator, some individual's genes are changed with at low random probability. A combined population R_t is formed from P_t and Q_t $(R_t = P_t \cup Q_t)$.

4.2   New Population Selection

From R_t only the half of individuals will be selected to the next generation (P_{t+1}). Nondominated sorting is applied to sorted and partitioned the population into fronts $(F_1, F2, \ldots,$ etc.$)$ according to the dominating status of each individual. The progress of fast nondominated sorting strategy is explained in Algorithm 1. After constructing the fronts, a selection phase is needed to choose among the fronts the solutions that will form P_{t+1}. If the selection is made between two solutions from different fronts, we prefer the solution with the lower (better) rank. Otherwise, if both solutions belong to the same front, then we use crowding distance operator to decide. More details are described in Algorithm 2.

The crowding distance calculation mechanism works as follows: for each objective function, the boundary solution is assigned an infinite distance value. All other intermediate solutions are assigned a distance value calculated as given in Eq. 4.

$$CD_i = \sum_{m=1}^{M} \frac{f_m(x+1) - f_m(x-1)}{f_m(\mathrm{x_{max}}) - f_m(\mathrm{x_{min}})} \tag{4}$$

where CD_i represents the crowding distance value of the individual i, M represents the number of objectives, $f_m(x+1)$ represents the mth objective function value of the individual $x+1$, and $f_m(\mathrm{x_{max}})$ and $f_m(\mathrm{x_{min}})$ separately represents the mth objective function value's maximum and minimum value. The selected individuals are those with maximum value of CD.

Algorithm 1. fast_nondominated_sorting()

1: **input**: R_t
2: **for all** $p \in R_t$ **do**
3:     $S_p \leftarrow \emptyset$ //S_p is the set of solutions dominated by p
4:     $n_p \leftarrow 0$ //n_p is the domination counter of p
5:     **for all** $q \in R_t$ **do**
6:         **if** $(p < q)$ **then**   //if p dominates q
7:             $S_p \leftarrow S_p \cup \{q\}$
8:         **else if** $(q < p)$ **then**
9:             $n_p \leftarrow n_p + 1$
10:         **end if**
11:     **end for**
12:     **if** $(n_p == 0)$ **then**
13:         $P_{rank} \leftarrow 1$
14:         $F_1 \leftarrow F_1 \cup \{p\}$ //F_1 represents the first front
15:     **end if**
16: **end for**
17: $i \leftarrow 1$
18: **while** $(F_i \neq \emptyset)$ **do**
19:     $Q \leftarrow \emptyset$ //Q will contain the members of the next front
20:     **for all** $p \in F_i$ **do**
21:         **for all** $q \in S_p$ **do**
22:             $n_q \leftarrow n_q - 1$
23:             **if** $(n_q == 0)$ **then**
24:                 $q_{rank} \leftarrow i + 1$
25:                 $Q \leftarrow Q \cup \{q\}$
26:             **end if**
27:         **end for**
28:     **end for**
29:     $i \leftarrow i + 1$
30:     $F_i \leftarrow Q$
31: **end while**
32: **output**: $\{F_1, F_2, ..., F_i\}$

---

5   Greedy Heuristics for the MWMCDS Problem

We have applied three greedy heuristics named as GR1, GR2 and GR3 to determine MWMCDS, the first heuristic (GR1) is similar than the second heuristic of [5], the second heuristic (GR2) and the third heuristic (GR3) are new heuristics proposed for the MWMCDS problem. These three algorithms follow the same process; initially, the set of candidate solution S is empty and the color of all vertices is WHITE. The first vertex that will belong S is selected from WHITE vertices and is colored BLACK while their neighbors are colored GRAY. The following vertices are chosen from GRAY vertices, the selected vertex belongs S and colored BLACK and their WHITE neighbors colored GRAY. BLACK vertices are called dominators. The algorithm terminate when there is no WHITE vertex in the graph.

Algorithm 2. selection()

1: **input**: $\{F_1, F_2, ..., F_i\}$
2: $i \leftarrow 1$
3: $cn \leftarrow 0$
4: **while** $(F_i \neq \emptyset)$ **do**
5:      $cn \leftarrow cn + |F_i|$
6:    **if** $(cn == N)$ **then**
7:        $P_{t+1} \leftarrow$ individuals which are in $[F_1, ..., F_i]$
8:        return P_{t+1}
9:    **else if** $(cn > N)$ **then**
10:        $P_{t+1} \leftarrow$ individuals which are in $[F_1, ..., F_{i-1}]$
11:        $I_{cr} \leftarrow$ crowding_distance(F_i) //I_{cr} contains the individuals selected using crowding distance calculation
12:        $P_{t+1} \leftarrow P_{t+1} \cup \{I_{cr}\}$
13:        return P_{t+1}
14:    **end if**
15:      $i \leftarrow i + +$
16: **end while**
17: **output**:P_{t+1}

The difference between these greedy algorithms lies in the selection method of dominators. Let $d_s(v)$ represents the number of WHITE neighbors of vertex v (current degree with respect to S), $t_w(v)$ denotes the total weight value of edges that connect vertex v with their neighbors $N(v)$, n represents the number of vertices in the graph G, and W_e denotes the total weight of all edges in G. The first heuristic (GR1) select the vertex having the greatest values of d_s.

$$v \leftarrow \mathbf{argmax}\{d_s(v)\} \mid v \in V\} \tag{5}$$

The second heuristic (GR2) select the vertex with the greatest ratio between their d_s and t_w.

$$v \leftarrow \mathbf{argmax}\{\frac{d_s(v)}{t_w(v)} \mid v \in V\} \tag{6}$$

The third heuristic (GR3) chooses the vertex having the minimum value of the difference between the normalized value of its t_w and d_s.

$$v \leftarrow \mathbf{argmin}\{\frac{t_w(v)}{W_e} - \frac{d_s(v)}{n} \mid v \in V\} \tag{7}$$

Figure 1 gives an illustrative example of a MWMCDS problem instance. The latter simple undirected edge-weighted graph which contains 9 vertices and 12 edges. The constructed solution shown in Fig. 1.a, Fig. 1.b and Fig. 1.c corresponds to the one obtained using the first greedy heuristic, the second greedy heuristic and the third greedy heuristic respectively.

In Fig. 1.a, the first vertex choosen is 1 because it has the greatest degree of WHITE neighbors ($d_s(1) = 5$), then the vertex 0 then 5. Thus, $F_c = 3$ and
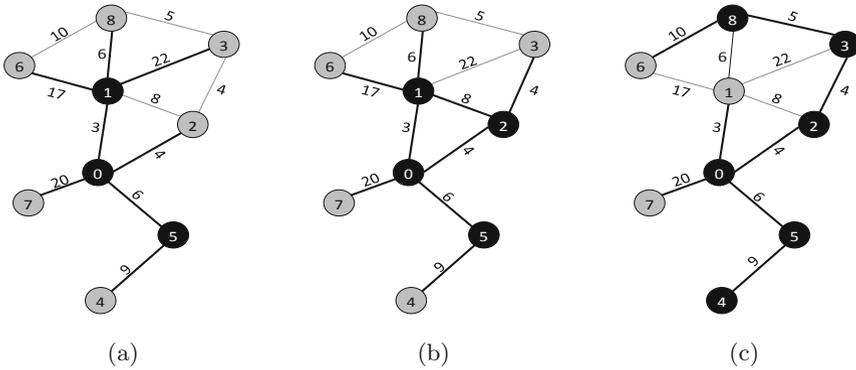
Fig. 1. Illustrative example of a MWMCDS problem instance.

$F_w = 87$. In Fig. 1.b the vertices 2, 0, 5 and 1 are selected in this order by applying Eq. 6. The values of objective functions of the obtained MWMCDS solution are $F_c = 4$ and $F_w = 77$. The MWMCDS solution found by the third heuristic (Eq. 7) is shown in Fig. 1.c, where the vertices 2, 0, 5, 4, 3 and 8 are selected in this order. This solution has cardinality of $F_c = 6$ and a total weight of $F_w = 61$.

6   The Proposed Algorithm I-NSGA-II

6.1   Initial Solution

In order to obtain better results, we generate one initial solution using greedy heuristic, and the remaining are generated randomly.

6.2   Local Search Method

Local search method is used to generate solutions neighbors in the search space. Let $d_g(v)$ represents the number of Gray neighbors of vertex v. For a solution S we generate the neighbor S' by applying the following changes. Initially, we search for the worst vertex in S (the vertex that gives worst results using greedy heuristic). Then, we remove it from the solution and we check if the produced solution stays connected. In that case, we stop, else we add to the solution a vertex greedily and we repeat this step until obtain a connected set.

6.3   Progress of I-NSGA-II Algorithm

Our proposed algorithm is named improved nondominated Sorting genetic algorithm I-NSGA-II, it first generates the initial solution set using greedy heuristic or randomly. Neighbors are generated using local search strategy or genetic algorithm with same probability. Fast nondominated sorting and crowding distance are used to sort and select vertices for next generation. Algorithm 3 shows the main structure of I-NSGA-II. In line 2, the parent population P_t which contains

Algorithm 3. I-NSGA-II for the MWMCDS problem

```
1: input: A problem instance (G, V, E, w), and parameter sol_size ∈ Z⁺
2: Pₜ ← generate_initial_solution(sol_size)
3: repeat
4:     Qₜ ← ∅
5:     for i ← 0 to sol_size do
6:         Sᵢ ← the iᵗʰ solution from Pₜ
7:         p ← random number uniformaly distributed over[0,1]
8:         if p > 0.5 then
9:             S′ᵢ ← local_search(Sᵢ)
10:        else
11:            S′ᵢ ← genetic_algorithm(Sᵢ)
12:        end if
13:        Qₜ ← Qₜ ∪ {S′ᵢ}
14:    end for
15:    Rₜ ← Pₜ ∪ Qₜ
16:    F ← fast_nondominated_sorting(Rₜ) //see algorithm 1
17:    Pₜ₊₁ ← selection(F) //see algorithm 2
18:    Pₜ ← Pₜ₊₁
19: until termination condition is satisfied
20: output: argmin {S^best, F_c(S^best), F_w(S^best)}
```

sol_size individuals (each individual represents a feasible solution S) is constructed greedily or randomly. Lines 5–13 produce the offspring population Q_t by generating a neighbor for each solution in P_t. Neighbors are obtained by either applying local_search(S_i) method or by using genetic_algorithm(S_i) (crossover and mutations operations). R_t is constructed in line 14, then it sorted into fronts on line 15 by applying the function fast_nondominated_sorting(R_t). A selection phase is needed to select the next generation P_{t+1} from the set of fronts F using the function selection(F) (line 16). These steps are repeated until a maximum number of iterations is reached.

7   Experiments and Results

7.1   Experiments

The proposed algorithm I-NSGA-II was implemented using C++ language, on a PC Intel Pentium, 2.20 GHz CPU, and 2 GB RAM. We also implemented three greedy heuristics to evaluate the performance of the proposed algorithm I-NSGA-II.

Benchmark instances used in this work were originally proposed in [5] where each instance consists of an undirected connected vertex-weighted graph with n vertices and m edges. Since our problem requires an edge-weighted graph, we generate the edge-weights randomly in the interval $[20, 120]$.

The instances are divided into two groups: small and medium instances which includes $\{10, 25, 50, 100, 200, 250\}$ vertices, and large instances which contains

$\{500, 750, 1000\}$ vertices. The number of edges m is varied for each vertex count n to observe the impact of the degree of connectivity between vertices on the results. For a specific number of nodes and edges, 10 instances exist for most graphs. The obtained results represent the average of running the algorithms on these 10 instances.

7.2   Results

As MWMCDS problem has two objectives: to minimize the total weight and to minimize the cardinality, the performance comparison of algorithms that deal with this problem must be done based on multiple performance indicators. This paper uses hypervolume indicator (noted by HVI) [21] with reference point (n, W_e). Recall that n represents the number of vertices in the graph G, and W_e denotes the total weight of all edges in G. The algorithm that performs better is the one that has the greatest value of hypervolume indicator. In addition to the hypervolume indicator, we have also compared between the algorithms based on the pareto front for the two objective values F_c and F_w and the execution time.

The experimental results are organized into two parts: the first one corresponds to the results generated by the greedy heuristics and the second represents the results of the proposed algorithm I-NSGA-II and NSGA-II.

Part I: Results of Greedy Heuristics. The results obtained by applying the three greedy heuristics described in Sect. 5 are given in Table 1 and those for large instances are presented in Table 2. These tables are organized as follows. The first two columns define the instance size, in terms of the number of vertices (n) and the number of edges (m). The results of each greedy heuristic are provided in two columns. The first one with heading **Time** shows the computation time (in seconds), and the second one with heading **HVI** provides the hypervolume indicator. The best results are highlighted in boldface in all tables.

Results for Small and Medium Size Instances. From Table 1, the comparison between the results of greedy heuristics with regard to execution time, shows that the three algorithms give similar results.
If we consider the hypervolume indicator, it is clearly seen that GR2 obtains the worst results. GR1 is better than GR2 up to 82.35% and GR3 is better than GR2 at most 88.23%, while GR3 is better than GR1 up to 64.70%.

Results for Large Size Instances. It can be observed from Table 2 that the time results of GR1 is generally the worst. GR2 and GR3 are faster than GR1 in 11 and 12 out of 15 instances respectively. GR3 executes 9 times faster than GR2. With respect to hypervolume indicator. GR2 is the worst, GR1 and GR3 are better than GR2 for all datasets. GR3 is better than GR1 in 9 out of 15 cases.

From the previous results, we can conclude that GR3 is the best heuristic with regard to both execution time and hypervolume indicator in small and medium and large size problem instances. Thus, this heuristic will be used in our proposed algorithm.

Table 1. Results of greedy heuristics for small and medium instances.

n	m	GR1		GR2		GR3	
		Time (s)	HVI	Time (s)	HVI	Time (s)	HVI
10	20	0.000	5632.4	0.000	**5714.2**	0.000	56984.5
	40	0.000	17232.4	0.000	16987.3	0.000	**18025.8**
25	100	0.000	116325.1	0.000	102465.3	0.000	**120124.4**
	250	0.000	**362241.5**	0.000	357412.3	0.000	360063.2
50	100	0.000	144417.4	0.000	**148023.4**	0.000	147251.9
	500	0.001	**1.525E+06**	**0.000**	1.403E+06	0.001	1.502E+06
100	200	0.001	570321.2	**0.000**	463475.1	0.001	**590814.8**
	600	0.002	3.147E+06	**0.001**	2.548E+06	0.002	**3.255E+06**
	1000	0.005	**5.803E+06**	0.004	5.511E+06	**0.003**	5.641E+06
200	400	0.007	**2.125E+06**	0.007	1.741E+06	**0.006**	2.005E+06
	1200	**0.049**	1.048E+07	0.052	1.198E+07	0.053	**1.213E+07**
	2000	0.102	2.323E+07	**0.084**	2.117E+07	0.092	**2.398E+07**
250	500	**0.046**	**3.610E+06**	0.048	2.759E+06	0.050	3.542E+06
	1000	0.091	1.036E+07	**0.084**	9.222E+06	0.086	**1.182E+07**
	1500	0.114	1.579E+07	**0.108**	1.541E+07	0.116	**1.705E+07**
	2000	**0.142**	**2.965E+07**	0.144	2.566E+07	0.145	2.784E+07
	2500	0.186	3.242E+07	0.187	2.895E+07	**0.182**	**3.540E+07**
Average		0.043	8.197E+06	**0.042**	7.496E+06	0.043	**8.556E+06**

Table 2. Results of greedy heuristics for large instances.

n	m	GR1		GR2		GR3	
		Time (s)	HVI	Time (s)	HVI	Time (s)	HVI
500	1000	0.382	**1.324E+07**	0.375	1.071E+07	**0.360**	1.219E+07
	2000	0.508	4.256E+07	**0.456**	3.224E+07	0.512	**4.478E+07**
	3000	1.021	6.908E+07	**0.834**	6.001E+07	1.018	**7.311E+07**
	4000	1.243	8.365E+07	1.190	8.142E+07	**1.162**	**9.003E+07**
	5000	**1.027**	**1.201E+08**	1.281	9.158E+07	1.300	1.183E+08
750	1500	1.363	**3.075E+07**	1.520	2.172E+07	**2.184**	2.202E+07
	3000	1.511	1.043E+08	**1.463**	8.281E+07	1.140	**1.359E+08**
	4500	1.587	1.623E+08	1.542	1.520E+08	**1.489**	**1.706E+08**
	6000	1.713	**2.447E+08**	**1.604**	2.0230E+08	1.668	2.410E+08
	7500	1.855	3.022E+08	1.721	2.612E+08	**1.707**	**3.109E+08**
1000	2000	**1.922**	**4.543E+07**	2.030	4.014E+07	1.975	4.486E+07
	4000	2.365	1.530E+08	2.417	1.311E+08	**2.281**	**1.787E+08**
	6000	2.665	2.914E+08	2.541	2.455E+08	**2.532**	**3.130E+08**
	8000	2.699	**3.611E+08**	2.641	3.210E+08	**2.547**	3.451E+08
	10000	2.802	3.912E+08	**2.784**	3.216E+08	2.786	**4.208E+08**
Average		1.636	1.610E+08	1.630	1.370E+08	**1.606**	**1.680E+08**

Table 3. Results of I-NSGA-II and NSGA-II for small and medium instances.

n	m	I-NSGA-II		NSGA-II	
		Time (s)	HVI	Time (s)	HVI
10	20	**0.051**	**6124.4**	0.077	6032.5
	40	**0.057**	**22082.3**	0.089	19964.2
25	100	**0.326**	**172892.2**	0.555	165076.4
	250	**0.392**	401658.9	0.594	**406391.0**
50	100	**0.759**	183269.5	1.020	**185863.2**
	500	**2.300**	**1.629E+06**	2.574	1.591E+06
100	200	**5.198**	**746347.1**	7.621	589463.8
	600	**7.313**	**3.417E+06**	11.106	3.125E+06
	1000	**11.884**	**6.430E+06**	14.861	5.988+06
200	400	**15.935**	2.324E+06	21.766	**2.564E+06**
	1200	**25.560**	**1.487E+07**	33.901	1.269E+07
	2000	**33.242**	**2.632E+07**	43.705	2.471E+07
250	500	**28.983**	**3.875E+06**	40.550	3.145E+06
	1000	**31.665**	**1.406E+07**	39.751	9.993E+06
	1500	**33.764**	**2.179E+07**	44.810	1.871E+07
	2000	**43.767**	**4.105E+07**	72.504	3.614E+07
	2500	**39.239**	**4.476E+07**	56.019	3.789E+07
Average		**16.496**	**10.709E+06**	23.029	9.308E+06

Table 4. Results of I-NSGA-II and NSGA-II for large instances.

n	m	I-NSGA-II		NSGA-II	
		Time (s)	HVI	Time (s)	HVI
500	1000	**81.350**	**1.827E+07**	127.864	1.331E+07
	2000	**82.486**	**4.958E+07**	125.323	3.774E+07
	3000	**90.205**	**8.502E+07**	129.650	6.390E+07
	4000	**95.639**	**1.326E+08**	136.268	9.706E+07
	5000	**95.695**	**1.843E+08**	136.158	1.510E+08
750	1500	**126.493**	**3.444E+07**	148.314	2.798E+07
	3000	**127.305**	**1.236E+08**	156.980	8.548E+07
	4500	**132.581**	**2.024E+08**	149.362	1.684E+08
	6000	**134.268**	2.610E+08	168.029	**2.655E+08**
	7500	**140.031**	**3.608E+08**	172.690	2.999E+08
1000	2000	**162.100**	**5.208E+07**	169.327	3.645E+07
	4000	**157.104**	**2.129E+08**	184.362	8.500E+07
	6000	**171.302**	**3.482E+08**	178.553	2.540E+08
	8000	**166.121**	**4.784E+08**	181.267	3.633E+08
	10000	**182.563**	**6.321E+08**	205.581	5.210E+08
Average		**129.682**	**2.117E+08**	157.981	1.646E+08

Part II: Results of I-NSGA-II and NSGA-II. Table 3 reported the results of I-NSGA-II and NSGA-II for small and medium instances and The results for large instances are presented in Table 4. Figure 2 and Fig. 3 visualize the pareto

front for the two objective functions F_c and F_w for small and medium instances and large instances respectively.

Results for Small and medium size instances From Table 3 is clearly seen that I-NSGA-II runs faster in all cases compared with NSGA-II. The hypervolume indicator results of I-NSGA-II are better than NSGA-II results at most 82.35%. With respect to F_c and F_w values, we can observe from Fig. 2 that in all the problem instances, I-NSGA-II gives better results, especially in pursuing the objective F_c.

Results for Large Size Instances. As shown in Table 4, I-NSGA-II performs better than NSGA-II in all instances with respect to the run time. In terms of the hypervolume indicator, I-NSGA-II obtains better results up to 93.33%. Figure 3 shows that solutions produced by I-NSGA-II distribute with well diversity and dominate that of NSGA-II.

(a) 10 vertices   (b) 25 vertices   (c) 50 vertices

(d) 100 vertices   (e) 200 vertices   (f) 250 vertices

Fig. 2. F_c and F_w results of I-NSGA-II and NSGA-II for small and medium instances.
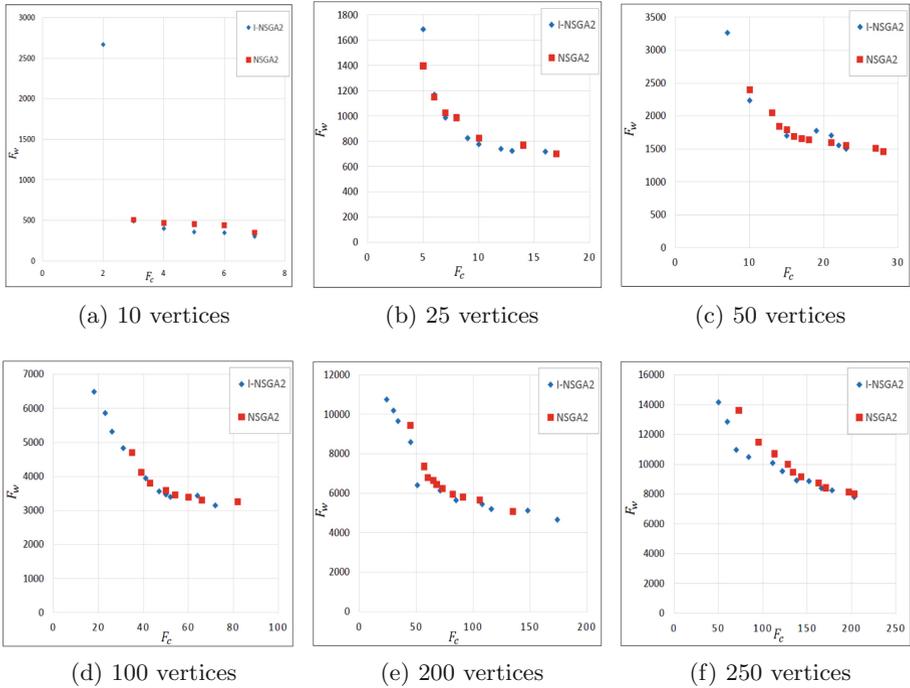
(a) 500 vertices   (b) 750 vertices   (c) 1000 vertices
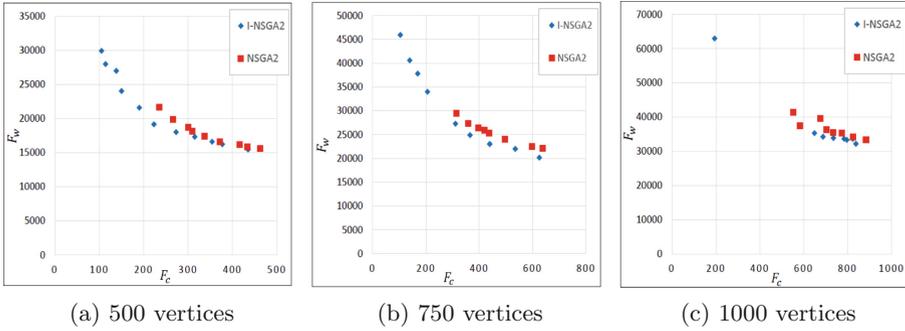
Fig. 3. F_c and F_w results of I-NSGA-II and NSGA-II for large instances.

8   Conclusion

In this paper, we have studied a specific NP-hard variant of the dominating set problem known as minimum weight minimum connected dominating set (MWM-CDS) problem. The latter is a bi-objective optimization problem that seeks to minimize both the cardinality and the total weight of the connected dominating set. The two objectives are not necessarily conflicting. To solve the bi-objective model, we developed an improved version of NSGA-II referred to as I-NSGA-II. The main feature of I-NSGA-II lies in the fact that greedy heuristics are used both for generating initial solutions and for exploring local search neighborhoods.

Furthermore, we have implemented three greedy heuristics for the problem and the best performing one (on average) among them is chosen to be used in the development of our approach. The experimental results, in comparison with NSGA-II show that I-NSGA-II outperforms NSGA-II with regard to solution quality and computation time. As future work, we plan to combine I-NSGA-II with other optimization techniques in order to enhance its performance and perform a more detailed analysis on large-size problem instances.

References

1. Blum, J., Ding, M., Thaeler, A., Cheng, X.: Connected dominating set in sensor networks and manets. In: Handbook of Combinatorial Optimization, pp. 329–369. Springer, Heidelberg (2004)
2. Bouamama, S., Blum, C., Fages, J.G.: An algorithm based on ant colony optimization for the minimum connected dominating set problem. Appl. Soft Comput. **80**, 672–686 (2019)
3. Cai, D., Gao, Y., Yin, M.: NSGA-II with local search based heavy perturbation for bi-objective weighted clique problem. IEEE Access **6**, 51253–51261 (2018)
4. Chinnasamy, A., Sivakumar, B., Selvakumari, P., Suresh, A.: Minimum connected dominating set based RSU allocation for smartcloud vehicles in VANET. Cluster Comput. **22**(5), 12795–12804 (2019)

5. Dagdeviren, Z.A., Aydin, D., Cinsdikici, M.: Two population-based optimization algorithms for minimum weight connected dominating set problem. Appl. Soft Comput. **59**, 644–658 (2017)
6. Dahmri, H., Bouamama, S.: Multiobjective simulated annealing algorithm for minimum weight minimum connected dominating set problem. In: Accepted for Publication International Conference on Defense systems: Architectures and Technologies, DAT 2020, pp. 1–6. IEEE (2020)
7. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)
8. Emmerich, M.T.M., Deutz, A.: A tutorial on multiobjective optimization: fundamentals and evolutionary methods. Nat. Comput. **17**(3), 585–609 (2018)
9. Gary, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. WH Freeman and Company, New York (1979)
10. Milenković, T., Memišević, V., Bonato, A., Pržulj, N.: Dominating biological networks. PloS ONE **6**(8) (2011)
11. Misra, R., Mandal, C.: Minimum connected dominating set using a collaborative cover heuristic for ad hoc sensor networks. IEEE Trans. Parallel Distrib. Syst. **21**(3), 292–302 (2009)
12. Nazarieh, M., Wiese, A., Will, T., Hamed, M., Helms, V.: Identification of key player genes in gene regulatory networks. BMC Syst. Biol. **10**(1), 88 (2016)
13. Rengaswamy, D., Datta, S., Ramalingam, S.: Multiobjective genetic algorithm for minimum weight minimum connected dominating set. In: International Conference on Intelligent Systems Design and Applications, pp. 558–567. Springer, Heidelberg (2017)
14. Salimi, R., Motameni, H., Omranpour, H.: Task scheduling using NSGA II with fuzzy adaptive operators for computational grids. J. Parallel Distrib. Comput. **74**(5), 2333–2350 (2014)
15. Sen, A., Banerjee, S., Ghosh, P., Murthy, S., Ngo, H.: Brief announcement: on regenerator placement problems in optical networks. In: Proceedings of the Twenty-Second Annual ACM Symposium on Parallelism in Algorithms and Architectures, pp. 178–180. ACM (2010)
16. Sen, A., Murthy, S., Bandyopadhyay, S.: On sparse placement of regenerator nodes in translucent optical network. In: 2008 IEEE Global Telecommunications Conference, IEEE GLOBECOM 2008, pp. 1–6. IEEE (2008)
17. Shukla, K.K., Sah, S.: Construction and maintenance of virtual backbone in wireless networks. Wirel. Netw. **19**(5), 969–984 (2013)
18. Sofia, A.S., GaneshKumar, P.: Multi-objective task scheduling to minimize energy consumption and makespan of cloud computing using NSGA-II. J. Netw. Syst. Manage. **26**(2), 463–485 (2018)
19. Wang, S., Zhao, D., Yuan, J., Li, H., Gao, Y.: Application of NSGA-II algorithm for fault diagnosis in power system. Electr. Power Syst. Res. **175**, 105893 (2019)
20. Wu, X., Lü, Z., Galinier, P.: Restricted swap-based neighborhood search for the minimum connected dominating set problem. Networks **69**(2), 222–236 (2017)
21. Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C.M., Da Fonseca, V.G.: Performance assessment of multiobjective optimizers: an analysis and review. IEEE Trans. Evol. Comput. **7**(2), 117–132 (2003)

Ontology Matching Using Neural Networks: Evaluation for OAEI Tracks

Meriem Ali Khoudja[(✉)], Messaouda Fareh, and Hafida Bouarfa

LRDSI Laboratory, Faculty of Sciences, Saad Dahleb University, Blida, Algeria
alikhoudjameriem@gmail.com, farehm@gmail.com, hafidabouarfa@hotmail.com

Abstract. Ontology matching is a proper method to establish inter-operability among heterogeneous ontologies. In this paper, we evaluate the ontology matching approach that we proposed previously and developed on a high level of accuracy. For that purpose, we performed a very detailed experimental study on six test cases of different domains from four OAEI tracks of various campaigns. The experimental results, adopting a cross-validation procedure and basing on standard evaluation measures, show a very high accuracy of matching. The proposed approach has proven its matching efficiency in front of all OAEI matching systems of the selected campaigns with major scores of all evaluation metrics adopted for all matching challenges. That permits to perfectly increase the performance of the ontology matching task.

Keywords: Ontology matching · Artificial neural networks · Supervised learning · Alignment · Experimental tests · OAEI tracks

1 Introduction

Knowledge representation is defined by the set of tools of which the objective is to organize human knowledge in order to be used and shared. Ontologies are the cornerstone of knowledge representation. They allow sharing knowledge of a given domain as a communication tool for its applications developed in a different way. According to [1], an ontology is a specification of a conceptualization, that is, a description of the concepts and relationships that may exist for a particular domain.

However, the rapid development of the Semantic Web, and the fact that the majority of applications require access to multiple ontologies from different domains, lead to heterogeneity at different levels. So, identifying semantic equivalences between entities of heterogeneous ontologies is very interesting. These equivalences, called alignments, are the core issue of the ontology matching task, which is the best solution to this ontology semantic heterogeneity problem.

Artificial neural networks are machine learning tools, biologically inspired the from human brain, and the way how it learns from examples. They have been widely used in several fields, particularly in ontology matching.

In our latter work [2], we proposed a new automatic ontology matching approach based on neural networks. It consists of training the network to learn weights for the top ranked matching tools in order to define a matching function that leads to generate the ideal set of alignments between ontologies. In this paper, we evaluate this matching approach through a very detailed experimental study adopting a cross validation procedure and according to six different test cases from OAEI tracks: CONFERENCE, BIODIVERSITY AND ECOLOGY, PROCESS MODEL MATCHING and ONTOLOGY MATCHING FOR QUERY ANSWERING.

The remainder of this paper is organized as follows. After this introduction, we first review existing work in the area of ontology matching, particularly systems based on neural networks. In Sect. 3 we present an overview of our ontology matching approach. In Sect. 4, we describe the evaluation design where we conduct our experiments, we present their results and we analyze them and discuss the performance of our approach. Finally, we conclude this paper and outline our future work in Sect. 5.

2 Related Work

Ontology matching is an effective method to establish interoperability between heterogeneous ontologies. For that object, several matching approaches have been proposed in the literature. Ontology matching process is generally based on computing similarity between ontologies. On this basis, that various classifications are given for ontology matching techniques as in [3–5].

Artificial neural networks have been widely used in the field of ontology matching. In our work [6], we study and classify the different ontology matching approaches based on neural networks. This study has shown that the main difference between them resides in the matching strategy, and in the purpose of applying neural networks in ontology matching. Some of them are for the purpose of a function approximation, whereas others aim at classifying concepts of ontologies for generating alignments between them. The analysis of all these techniques allows us to outline contributions of the ontology matching approach that we proposed in [2]. In the following, we present three different matching systems.

- In work [7], a new supervised learning based method for compound metric creation is proposed. A training set is used to create a neural network model, performs sensitivity analysis on it to select appropriate metrics among a set of existing ones, and finally constructs a neural network model to combine the result metrics into a compound one.
- PRIOR+ [8–10] is a generic and adaptive ontology mapping approach, based on propagation theory, information retrieval techniques and artificial intelligence. It first measures both linguistic and structural similarity of ontologies in a vector space model using classic information retrieval techniques, and aggregates them using an adaptive method based on their harmonies.

Then, the interactive activation and competition neural network is selectively activated to solve the constraint satisfaction problem in the context of ontology mapping.

– CIDER is a schema-based ontology alignment algorithm. It compares each pair of ontology terms by extracting their ontological contexts and combining different elementary ontology matching techniques. Its participations at the OAEI campaigns are presented in [11–13]. CIDER first extracts the ontological contexts for each ontology terms pair up to a certain depth and enriches it by applying lightweight inference rules, and then combines the different elementary ontology matching techniques using artificial neural networks in order to generate alignments between ontologies.

3   Overview of the Approach

The processing flow of our approach is illustrated in Fig. 1. The first step consists of constructing the dataset on N sets of alignments between the two ontologies to be matched. Each set is given by an efficient matching system using its own specific matching technique. N depends on choice criteria. These sets are refined according to the environmental conditions of our approach. The second step, which is the core process of our matching approach, consists of applying a supervised learning procedure based on neural networks, in order to; first adjust an importance value for each system using a linear perceptron of N inputs and one output, and the back-propagation learning method; and then define the matching function that leads to generate alignments between the two ontologies. The final step consists of filtering alignments in order to get the final alignments set using a defined threshold.

4   Experiments

In order to better evaluate our ontology matching approach, we conducted an experimental procedure described as follows.

4.1   Experimental Design

We evaluate our approach according to several campaigns of the Ontology Alignment Evaluation Initiative (OAEI),[1] which is an international initiative for evaluating ontology matching systems on the same basis using different types and sizes of test ontologies. We select all tracks of normal scale where the reference mapping of the corresponding ontologies, generated alignments of participating systems and their evaluation results are available. Each track is taken for the most recent year providing not to exceed five recent years. Thus, we set up our experiments on six test cases from four OAEI campaigns (two of them comprise two test cases that we consider as sub-tracks), one from OAEI'2015, two from OAEI'2017, and three from OAEI'2018 campaign. We present below a description of each test case.

---

[1]   http://oaei.ontologymatching.org/.
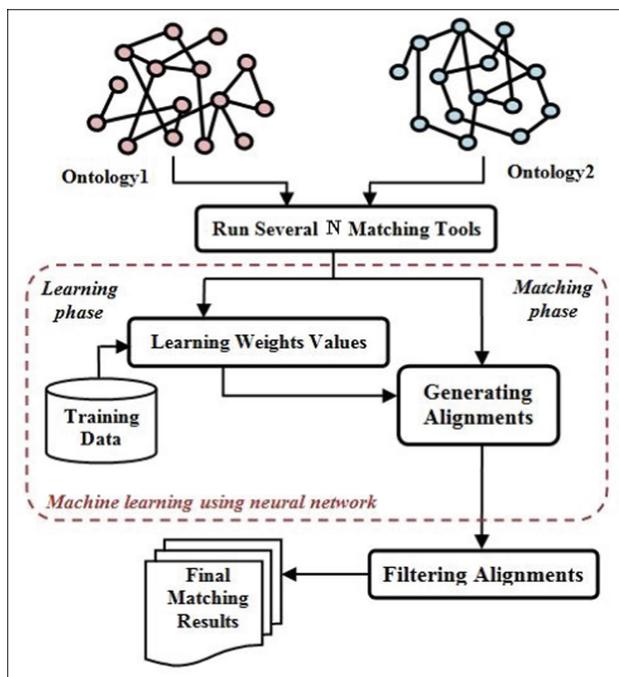
Fig. 1. Processing flow of the proposed approach [2].

OAEI'2018-CONFERENCE Track (Conference). The goal of this track[2] is to find alignments within a collection of 16 ontologies describing the domain of organising conferences. These ontologies, developed within OntoFarm project, are suitable for ontology matching task because of their heterogeneous character of origin. Tests are performed on a suite of 21 matching tasks of the pairwise combination of seven moderately expressive ontologies describing the same domain. Participant systems results had been evaluated based on crisp reference alignment, its uncertain version and logical reasoning evaluation based on violations of conservative principle.

OAEI'2018-BIODIVERSITY AND ECOLOGY Track (BioDiv). The goal of this track[3] is to find pairwise alignments between four ontologies being used in various projects and particularly useful for biodiversity and ecology research. They are semantically rich and very overlapping. The reference alignments are produced using established matching systems to produce an automated consensus alignment, and then manually validating the unique results produced by each system, and finally adding manually generated correspondences. The track features two challenges:

---

[2] http://oaei.ontologymatching.org/2018/conference/index.html.
[3] http://oaei.ontologymatching.org/2018/biodiv/index.html.

FLOPO-PTO. This challenge consists on finding alignments between the Flora Phenotype Ontology (FLOPO) (24199 classes) and the Plant Trait Ontology (PTO) (1504 classes).

ENVO-SWEET. This challenge consists on finding alignments between the Environment Ontology (ENVO) (6909 classes) and the Semantic Web for Earth and Environment Technology Ontology (SWEET) (4543 classes).

OAEI'2017-Process Model Matching Track (PM). This track[4] is a spinoff from the Process Model Matching Contest. It is concerned with the task of matching process models, originally represented in BPML. These models have been converted to an ontological representation. The resulting matching task is a special case of an interesting instance matching problem. The track comprises two test cases:

University ADMISSION. The dataset of this task consists of nine process models that describe the process of university admission for different German universities. The BPMN representation of the process models was converted to a set of assertions (ABox) using the vocabulary defined in the BPMN 2.0 ontology (TBox).

BIRTH REGISTRATION. The dataset of this task consists of process models that describe the process of registering a newborn child in different countries and related administrative tasks. These process models were originally available as Petrinets as *.pnml. These datasets have also been converted into ontologies, more precisely into ABoxes.

OAEI'2015-ONTOLOGY ALIGNMENT FOR QUERY ANSWER-ING Track (OA4QA). The goal of the OA4QA track[5] is to measure the ability of generating alignments to answer a set of queries in an ontology-based data access scenario where several ontologies exist. The dataset is based on the Conference track, and extended with synthetic Aboxes extracted from the DBLP dataset. The reference answer set used is the publicly available reference alignment of the Conference track and a manually repaired version of it from conservativity and consistency violations.

The method of constructing the dataset determines the size and components of the dataset. Thus, it has a big impact on our alignments results. We worked on three completeness levels and compared their results in order to conclude the best strategy for producing the dataset:

– **Intersection:** consists of keeping only common alignments pairs between chosen mappings.

---

4 http://web.informatik.uni-mannheim.de/oaei/pm17/.
5 http://oaei.ontologymatching.org/2015/oa4qa/index.html.

– **_Majority:_** consists of taking alignments pairs generated by the majority of the matching tools.
– **_Union:_** consists of taking alignments pairs for which at least a chosen matching system has given a value.

We adopt a cross validation to effectively control the network while training and testing. We use the standard evaluation measures: precision, recall as well as three variants of F_b-measure, against the reference alignments. The main choice criterion of the first step of our approach is based on F_1-measure, because it is the harmonic mean of precision and recall where both receive equal weight.

4.2   Experimental Results

Aiming to study the effectiveness of our ontology matching approach, we compare its results, by its three different variants and for each one of the six test cases, with the results of all OAEI participant systems for the same test challenge, and adopting the same cross validation procedure. This comparison is done for the global dataset in terms of the five standard evaluation metrics defined previously.

In the following, we mean by APP-I, APP-M and APP-U our matching approach by intersection, majority and union dataset construction method respectively.

In order to effectively evaluate our matching approach according to **OAEI'2018-CONFERENCE Track**, we compare its results with those of all participant matching systems for the same test case. Table 1 summarizes this comparison, and graphs in Fig. 2 illustrate the global cross validation values of the five standard evaluation measures obtained.
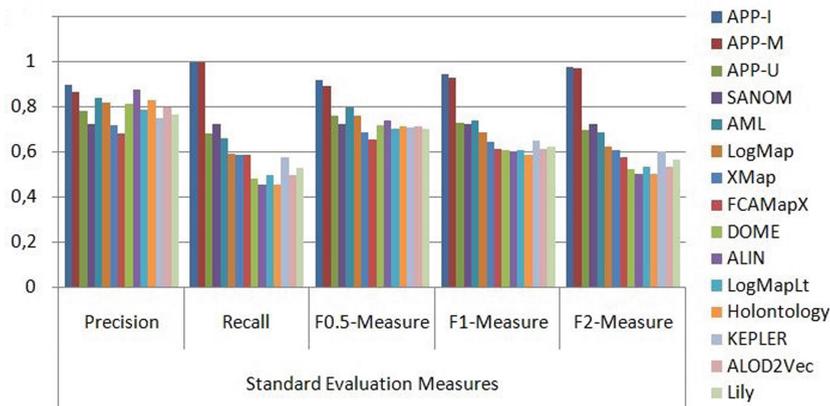

Fig. 2. Standard evaluation measures values of the global dataset obtained by our approach for intersection, majority, and union, and by OAEI matching systems for Conference'18 track.

Table 1. Global Precision, Recall and F-measures values of our approach by its three variants and of the OAEI participating systems for Conference'18 track.

System	Standard evaluation measures				
	Prec	Rec	F0.5-M	F1-M	F2-M
APP-I	**0.901**	**1.000**	**0.919**	**0.947**	**0.978**
APP-M	0.869	**1.000**	0.892	0.929	0.970
APP-U	0.783	0.681	0.760	0.728	0.699
SANOM	0.722	0.724	0.722	0.722	0.723
AML	0.840	0.658	0.796	0.738	0.688
LogMap	0.818	0.59	0.759	0.685	0.625
XMap	0.717	0.586	0.686	0.644	0.608
FCAMapX	0.683	0.586	0.653	0.613	0.578
DOME	0.816	0.482	0.717	0.606	0.525
ALIN	0.878	0.456	0.740	0.600	0.504
LogMapLt	0.785	0.495	0.702	0.607	0.534
Holontology	0.832	0.456	0.712	0.587	0.500
KEPLER	0.750	0.573	0.707	0.650	0.602
ALOD2Vec	0.799	0.495	0.711	0.611	0.536
Lily	0.768	0.530	0.703	0.625	0.564

The results of the global dataset hail from those of the several partitions. As shown in Fig. 2: 1. Our approach gives very good results (the minimum value is 0.681 given for recall by APP-U, which gives better good values for the four other metrics), especially Intersection and Majority versions that present complete values of recall (1.0) and excellent values of precision and the three F-measures. Thus, they present by far the best matching results. 2. Also for us, APP-I gets better performance than APP-M and better than APP-U. Contrary to the latter, the two other versions get high values of precision than of recall. 3. OAEI matching systems, as they are classified in Table 1, present almost a descending order of their recall values and no order of their precision values, thus, their F-measures values are also ordered decreasingly but less sharply. 4. As expressed by F1-measure, which better shows the real quality of the matching results, APP-I results are slightly better than those of APP-M, but they are roughly better than the others (the difference exceeds 0.2), including APP-U of which the results are close to those of AML and competitive to those of all the other matching systems.

The evaluation results of our approach according to **OAEI'2018-BIODIV Track**, and their comparison with those of all participant matching systems for this test case are summarized in Table 2. Plots in Fig. 3 and Fig. 4 illustrate the five standard evaluation measures values obtained for **FLOPO-PTO** and **ENVO-SWEET** Sub-Tracks respectively.

Table 2. Global Precision, Recall and F-measures values of our approach by its three variants and of the OAEI participating systems for BioDiv'18 track.

System	Standard evaluation measures									
	FLOPO-PTO					ENVO-SWEET				
	Prec	Rec	F0.5-M	F1-M	F2-M	Prec	Rec	F0.5-M	F1-M	F2-M
APP-I	**0.983**	**1.000**	**0.986**	**0.991**	**0.996**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
APP-M	0.796	**1.000**	0.822	0.871	0.939	0.767	**1.000**	0.795	0.848	0.926
APP-U	0.766	0.841	0.767	0.780	0.809	0.681	0.841	0.698	0.733	0.787
AML	0.865	0.852	0.857	0.850	0.849	0.708	0.884	0.727	0.765	0.824
LogMap	0.790	0.770	0.781	0.772	0.769	0.707	0.628	0.689	0.664	0.642
LogMapBio	0.779	0.770	0.772	0.766	0.766	0.718	0.626	0.695	0.663	0.634
XMap	0.968	0.556	0.822	0.688	0.601	0.750	0.626	0.721	0.682	0.647
LogMapLite	0.954	0.556	0.766	0.634	0.559	0.694	0.710	0.694	0.697	0.704
POMap	0.863	0.697	0.822	0.769	0.724	0.721	0.647	0.703	0.679	0.659
Lily	0.750	0.504	0.676	0.596	0.536	0.853	0.536	0.755	0.651	0.576

From Fig. 3, it can be seen that: 1. The best precision values are given by APP-I, then by XMap and LogMapLite, then by AML and POMap, and then come the other systems with the best value obtained by APP-M. 2. Constantly, APP-I and APP-M achieved complete global recall values. A very good value is obtained by APP-U as well. Then, the OAEI systems are ordered decreasingly, except POMap which is out of order. 3. For F0.5-measure, APP-I has by far the higher score. The other matching systems have close values around 0.8. For F1-measure and F2-measure, APP-I has the highest values followed by APP-M. The other systems are of a descending order starting from AML and APP-U, and more sharply for F2-measure.

It is clear from Fig. 4 that: 1. Globally, APP-I has complete scores for all the five evaluation measures. The next best results are given by APP-M (complete value of recall). Then, AML and APP-U obtain good results, and then come the other matching systems with acceptable results, presenting a slight descending order, excepting Lily and LogMapLite which are out of this order with higher scores for precision and recall respectively. 2. For F-measures, the OAEI values are closer in F0.5-measure than in F1-measure and than in F2-measure.

The results of evaluating our ontology matching approach and those of all participating systems according to **OAEI'2017-PM Track** as well as their comparison are summarized in Table 3. Plots in Fig. 5 and Fig. 6 outline the global cross validation values of the five standard evaluation measures obtained for **UA** and **BR** respectively.

We can see from Fig. 5 that: Globally, we can easily notice the same systems disparities for all evaluation measures. APP-I gets the highest results (with a complete recall value). APP-M gives excellent scores, followed by APP-U and AML with good and so close values. Log-Map and I-match have the smallest performance with scores around 0.5.
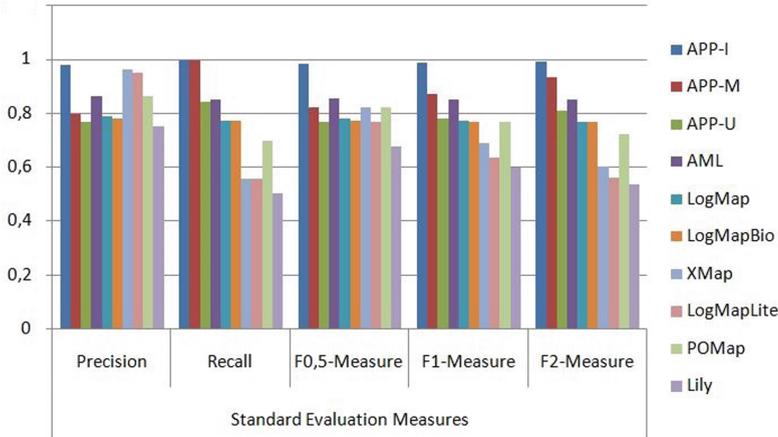
Fig. 3. Standard evaluation measures values of the global dataset obtained by our approach for intersection, majority, and union, and by OAEI matching systems for BioDiv-FLOPO-PTO'18 track.

Table 3. Global Precision, Recall and F-measures values of our approach by its three variants and of the OAEI participating systems for PM'17 track.

System	UNIVERSITY ADMISSION					BIRTH REGISTRATION				
	Prec	Rec	F0.5-M	F1-M	F2-M	Prec	Rec	F0.5-M	F1-M	F2-M
APP-I	0.905	1.000	0.922	0.949	0.979	0.843	1.000	0.870	0.914	0.963
APP-M	0.812	0.949	0.830	0.864	0.909	0.697	**1.000**	0.741	0.819	0.918
APP-U	0.709	0.681	0.702	0.693	0.685	0.514	0.294	0.421	0.354	0.314
AML	0.709	0.681	0.702	0.693	0.685	0.441	0.409	0.425	0.411	0.408
LogMap	0.450	0.517	0.462	0.481	0.502	0.551	0.271	0.426	0.343	0.295
I-Match	0.527	0.431	0.499	0.467	0.443	0.689	0.254	0.461	0.345	0.283

From Fig. 6, it is clear that: 1. The best precision score, given by APP-I, exceeds 0.8. The next one is obtained by APP-M, then by I-Match. The other matching systems give values around 0.5. 2. Once again, APP-I and APP-M get the complete performance for recall with values equal to 1.0. Whereas the other matching systems give very small values, the score of AML is higher, but there is still a huge difference between their results and those of APP-I and APP-M. Those systems (contrary to APP-I and APP-M) have worse results for F2-measure than for F1-measure and than for F0.5-measure. 3. For F-measures, the results obtained by APP-I and APP-M are excellent and by far better than the other matching systems, of which the F0.5-values variances depend on those of precision and F2-measure variances depend on those of recall whereas F1-measure results are balanced and didn't achieve the average. 4. The distance between the two groups (APP-I and APP-M, and the other systems) is smaller for precision than the other evaluation measures.
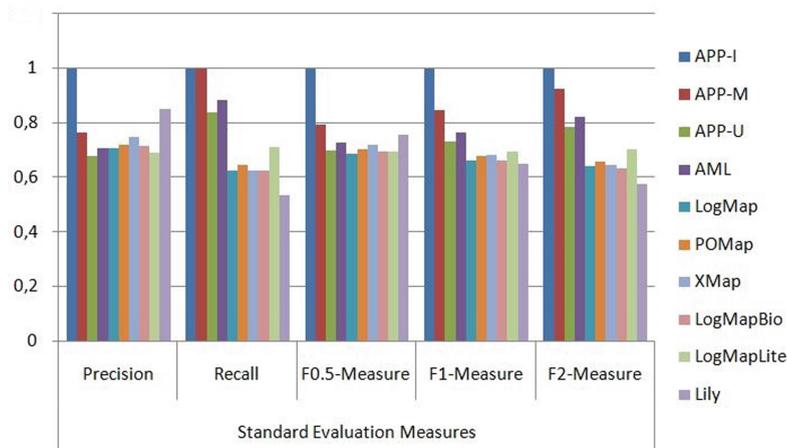
Fig. 4. Standard evaluation measures values of the global dataset obtained by our approach for intersection, majority, and union, and by OAEI matching systems for BioDiv-ENVO-SWEET'18 track.

The evaluation results of our approach according to **OAEI'2015-OA4QA Track**, and their comparison with those of all participant matching systems for this test case are summarized in Table 4, and graphs in Fig. 7 illustrate the five standard evaluation measures values obtained.

From Fig. 7, we can observe that: 1. Precision results of all matching systems are better than recall values (around 0.8), excepting APP-I and APP-M which have complete values of recall. Thus, their F0.5-measure results are also better than those of F1-measure and better than F2-measure results (also excluding APP-I and APP-M), but with sharp-less differences. 2. Recall values, excluding APP-I and APP-M with values equal to 1.0, and YAM++ and MaasMatch with a value equal to 0.7 and 0.63 respectively, did not exceed 0.6. 3. Since precision scores are more balanced and close to each other than recall results, the variances of the three F-measures depend on those of recall, but with different intensities, sharply for F2-measure than F1-measure and than F0.5-measure. 4. In total, the best performances belong to APP-I and APP-M with high and far scores. After-ward, APP-U is among the best 9 systems (from 23 systems) which obtain good re-sults. 3. The minimum value given for all the evaluation measures is 0.4. 4. It is clearly seen that precision results are better than recall results (excepting APP-I and APP-M which have values equal to 1.0 for recall). That gives some decreasing order of F-measures results; F0.5-measure results are better than F1-measure results and better than those of F2-measure.

4.3   Experimental Summary

According to the previous results, we can conclude the following:
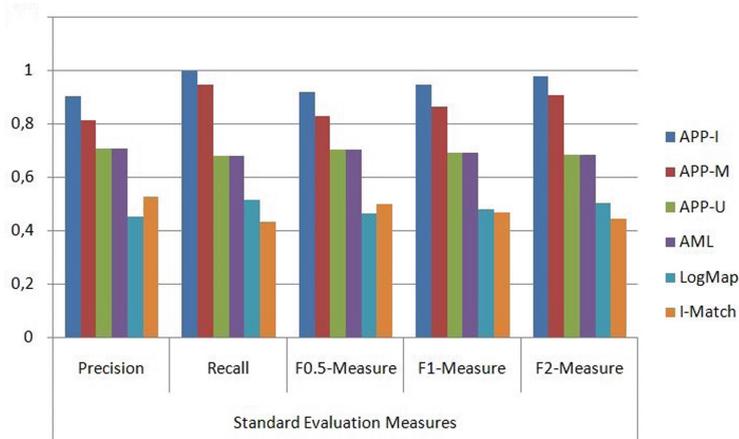
Fig. 5. Standard evaluation measures values of the global dataset obtained by our approach for intersection, majority, and union, and by OAEI matching systems for PM-UA'17 track.

The challenge with the maximum average precision values is **BioDiv-FLOPO-PTO** (around 0.85). The next one is OA4QA with values around 0.8. Then, Conference has precision values around 0.75 and BioDiv-ENVO-SWEET's precision scores exceed 0.7. Finally, comes PM where the minimum average precision results belong to its **PM-BR** sub-track. Precision values are separated in PM track and close to each other in the other test cases, less closely in BioDiv-FlOPO-PTO and OA4QA. In BioDiv-ENVO-SWEET, systems values are close and lower and far from our precision results.

All maximum precision values of all the six test cases of this experimental study are given by the intersection version of our ontology matching approach. The highest one is equal to 1.0 for BioDiv-ENVO-SWEET and the lowest one is equal to 0.84 for PM-BR. The highest minimum precision value is equal to 0.75 and given by Lily for BioDiv-FLOPO-PTO. The lowest minimum precision value in this experimental procedure is equal to 0.44 and given by AML for PM-BR tests.

The track with the maximum average recall values is **BioDiv** with its two sub-tracks (around 0.75). The next task is PM-UA with values around 0.73 and then Conference with values around 0.6. PM-BR average results are slightly higher than 0.5. Finally, **OA4QA** has the poorest average recall values (around 0.5). Recall values are separated in all tracks excepting PM-BR and OA4QA, where recall values are somewhat close to each other but the gap between the results of APP-I and APP-M and those of the other systems is big.

The maximum recall value in all this experimental procedure is equal to 1.0 and given by both APP-I and APP-M for all test cases, excluding PM-UA, where this complete value is obtained only by APP-I, and APP-M has a slightly lower but still excellent score. The highest minimum recall score is equal to 0.53 and

given by Lily for BioDiv-ENVO-SWEET, and the lowest minimum recall value is equal to 0.25 and given by I-Match for PM-BR sub-track.
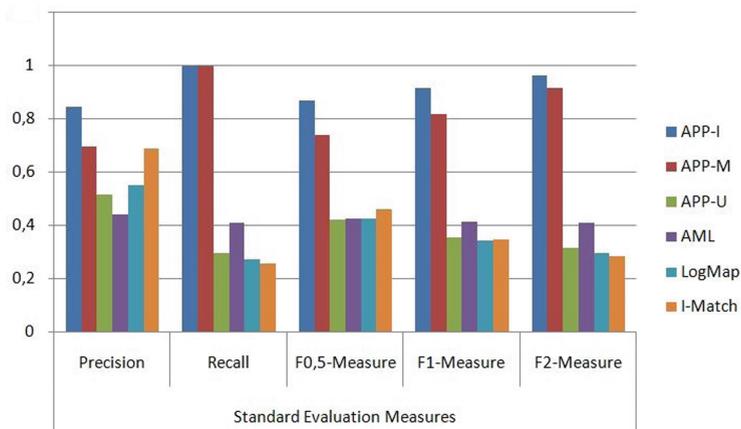

Fig. 6. Standard evaluation measures values of the global dataset obtained by our approach for intersection, majority, and union, and by OAEI matching systems for PM-BR'17 track.

The track with the highest average $F_{0.5}$-measure values is ***BioDiv-FLOPO-PTO*** (around 0.82), and next, Conference with average values equal to 0.75. Then, BioDiv-ENVO-SWEET and OA4QA have average $F_{0.5}$-measure results superior to 0.7. Finally, ***PM-BR*** has the lowest average scores (around 0.54). $F_{0.5}$-measure values are separated only in PM-UA and close to each other in the other test cases. In BioDiv-ENVO-SWEET, PM-BR and OA4QA, the $F_{0.5}$-measure results of our intersection and majority versions (even union in the 1st test case) are positively far from the other systems results.

All the maximum $F_{0.5}$-measure values in these experiments are given by APP-I. The highest one is equal to 1.0 for BioDiv-ENVO-SWEET and the lowest one is equal to 0.87 for PM-BR. The highest minimum $F_{0.5}$-measure value is equal to 0.68 and given by LogMap for BioDiv-ENVO-SWEET, and the lowest minimum $F_{0.5}$-measure value is equal to 0.42 and given by APP-U for PM-BR sub-track.

F_1-measure shows the real quality of matching. The task which has the maximum average F_1-measure results is ***BioDiv-FLOPO-PTO*** (around 0.8). BioDiv-ENVO-SWEET has a good one as well (around 0.74). The next track is Conference with values around 0.7. Then, PM-UA and OA4QA have average results so close to 0.7. Finally, ***PM-BR*** has the lowest results (around 0.56). F_1-measure values are separated in BioDiv-FLOPO-PTO and PM-UA and close to each other in the other tasks. Also in BioDiv-ENVO-SWEET, PM-BR and OA4QA, F_1-measure results of our APP-I and APP-M versions (even APP-U in the 1st test case) are positively far from the other systems results.

Table 4. Global Precision, Recall and F-measures values of our approach by its three variants and of the OAEI participating systems for OA4QA'15 track.

System	Standard evaluation measures					System	Standard evaluation measures				
	Prec	Rec	F0.5-M	F1-M	F2-M		Prec	Rec	F0.5-M	F1-M	F2-M
APP-I	**0.896**	**1.000**	**0.914**	**0.944**	**0.977**	HotMatch	0.817	0.514	0.731	0.631	0.556
APP-M	0.868	**1.000**	0.891	0.929	0.970	IAMA	0.811	0.478	0.711	0.601	0.521
APP-U	0.802	0.586	0.747	0.678	0.620	MaasMatch	0.857	0.633	0.800	0.727	0.667
AML	0.867	0.557	0.780	0.678	0.600	MapSSS	0.805	0.505	0.720	0.620	0.545
LogMap	0.804	0.593	0.751	0.682	0.626	ODGOMS	0.815	0.508	0.727	0.626	0.549
XMapSiG1_4	0.727	0.403	0.625	0.518	0.442	ODGOMS1_2	0.819	0.593	0.761	0.688	0.628
XMapGen	0.726	0.449	0.645	0.554	0.486	OntoK2	0.800	0.472	0.702	0.593	0.514
XMapGen1_4	0.744	0.449	0.690	0.623	0.569	RIMOM2013	0.824	0.574	0.755	0.674	0.610
XMapSiG1_3	0.741	0.482	0.668	0.583	0.518	ServOMap	0.887	0.551	0.788	0.678	0.595
CroMatcher	0.807	0.444	0.648	0.537	0.474	StringsAuto	0.807	0.537	0.733	0.645	0.576
LogMapLite	0.794	0.495	0.708	0.610	0.535	SYNTHESIS	0.799	0.446	0.690	0.572	0.489
AMLback	0.871	0.580	0.792	0.696	0.621	WeSeEMatch	0.881	0.465	0.747	0.609	0.514
CIDER_CL	0.809	0.531	0.732	0.641	0.570	WikiMatch	0.815	0.491	0.720	0.613	0.534
HerTUDA	0.814	0.505	0.725	0.623	0.546	YAM++	0.894	0.691	0.844	0.779	0.724

All the six maximum F_1-measure values of this experimental study are given by APP-I. The highest one is equal to 1.0 for BioDiv-ENVO-SWEET and the lowest one is equal to 0.91 for PM-BR. The highest minimum F_1-measure value is equal to 0.65 and given by Lily for BioDiv-ENVO-SWEET test case. The lowest minimum F_1-measure value is equal to 0.34 and given by LogMap for PM-BR sub-track.

The challenge that has the maximum average F_2-measure results is ***BioDiv-FLOPO-PTO*** (around 0.77). The other sub-track average results are not far as well. The next one is PM-UA with values around 0.71. Then, OA4QA and Conference have average values superior to 0.6. And finally, ***PM-BR*** has the poorest scores (around 0.58). F_2-measure values are close to each other in all test cases, excepting PM-BR and OA4QA where the F_2-measure results of our intersection and majority versions are positively far from the other matching systems results.

Also for this evaluation measure, all the maximum F_2-measure values are given by APP-I. The highest one is equal to 1.0 for BioDiv-ENVO-SWEET and the lowest one is equal to 0.96 for PM-BR. The highest minimum F_2-measure value is equal to 0.57 and given by Lily for BioDiv-ENVO-SWEET task. And, the lowest minimum F_2-measure value is equal to 0.28 and given by I-Match for PM-BR sub-track.

BioDiv-ENVO-SWEET sub-track has the maximum average recall results and BioDiv-FLOPO-PTO has the maximum average scores for precision and the three F-measures. Consequently, BioDiv is the track with the best results. OA4QA track has the lowest scores of recall, and the minimum average values of the other four evaluation measures belong to PM-BR.

The matching systems present better results for precision than for recall in four test cases: Conference (excepting APP-I, APP-M and ALIN),
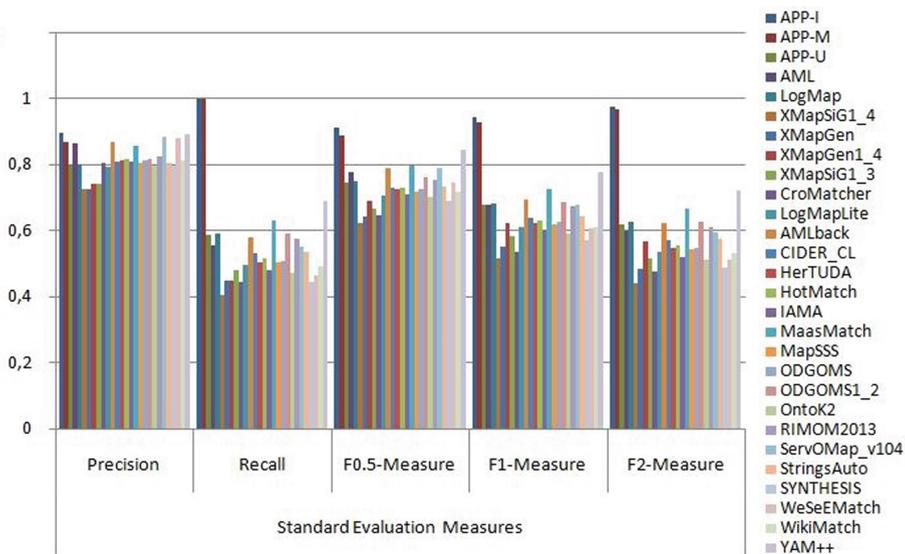
Fig. 7. Standard evaluation measures values of the global dataset obtained by our approach for intersection, majority, and union, and by OAEI matching systems for OA4QA'15 track.

BioDiv-FLOPO-PTO, PM-BR and OA4QA. Recall results are higher than precision results in the two other tasks: ENVO-SWEET (with a slight difference) and PM-UA. These variances are reflected on F-measures, where $F_{0.5}$-measure is mostly affected by precision and F_2-measure by recall, whereas F_1-measure combines them evenly.

For us, the results of APP-I are better than those of APP-M and better than those of APP-U for the six matching challenges. In Conference, PM-BR and OA4QA, APP-U results are far from those of APP-I and APP-M.

All in all, the detailed experimental procedure that we performed on six test cases shows that, the three variants of the ontology matching approach that we propose in our previous work [2], present better results than the OAEI matching systems, especially the Intersection and Majority variants which present the best matching performance. That is clear from the results of all the five evaluation measures adopted, especially from F_1-measure results. All the maximum scores of all the evaluation measures adopted for all test cases of this study have been achieved by APP-I. APP-M has also presented excellent results positively far from the OAEI systems results. And APP-U has given very good results which are competitive with the two best OAEI matching systems.

5   Conclusion

Ontology matching is an effective process to establish interoperability between heterogeneous ontologies. In this paper, we evaluate our ontology matching

approach which is proposed previously and developed on a high level of accuracy. For that objective, we conducted much precise experiments on six different test cases from four OAEI tracks of various campaigns. The experimental results show that the proposed approach has proven its efficiency in front of all the OAEI matching systems in terms of the evaluation measures adopted with very excellent scores for all matching tasks. That permits to significantly increase the performance of the ontology matching task.

Our experimental results are very encouraging. As future work, we aim at adopting the approach to accurately match large scale ontologies, and fully evaluating it on real and large ontologies.

References

1. Gruber, T.R.: A translation approach to portable ontology specifications. Knowl. Acquis. **5**(2), 199–220 (1993)
2. Ali Khoudja, M., Fareh, M., Bouarfa, H.: A new supervised learning based ontology matching approach using neural networks. In: International Conference Europe Middle East & North Africa Information Systems and Technologies to Support Learning (2018)
3. Shvaiko, P., Euzenat, J.: Ontology matching: state of the art and future challenges. IEEE Trans. Knowl. Data Eng. **25**(1), 158–176 (2013)
4. Otero-Cerdeira, L., Rodríguez-Martínez, F.J., Gómez-Rodríguez, A.: Ontology matching: a literature review. Exp. Syst. Appl. **42**(2), 949–971 (2015)
5. Ardjani, F., Bouchiha, D., Malki, M.: Ontology-alignment techniques: survey and analysis. Int. J. Mod. Educ. Comput. Sci. **7**(11), 67 (2015)
6. Ali Khoudja, M., Fareh, M., Bouarfa, H.: Ontology matching using neural networks: survey and analysis. In: 2018 International Conference on Applied Smart Systems (ICASS), pp. 1–6 (2018)
7. Hariri, B.B., Abolhassani, H., Sayyadi, H.: A neural-networks-based approach for ontology alignment. In: SCIS & ISIS SCIS & ISIS 2006, pp. 1248–1252. Japan Society for Fuzzy Theory and Intelligent Informatics (2006)
8. Mao, M., Peng, Y., Spring, M.: A profile propagation and information retrieval based ontology mapping approach. In: Semantics, Knowledge and Grid, Third International Conference, pp. 164–169 (2007)
9. Mao, M., Peng, Y., Spring, M.: An adaptive ontology mapping approach with neural network based constraint satisfaction. Web Seman. Sci. Serv. Agents World Wide Web **8**(1), 14–25 (2010)
10. Mao, M., Peng, Y., Spring, M.: An adaptive ontology mapping approach with neural network based constraint satisfaction. J. Web Semant. **8**(1), 14–25 (2010)
11. Gracia, J., Mena, E.: Ontology matching with CIDER: evaluation report for the OAEI 2008. In: Proceedings of the 3rd International Conference on Ontology Matching, vol. 431, pp. 140–146. CEUR-WS.org (2008)
12. Gracia del Río, J., Bernad, J., Mena, E.: Ontology Matching with CIDER: evaluation report for OAEI 2011, pp. 126–133 (2011)
13. Gracia, J., Asooja, K.: Monolingual and cross-lingual ontology matching with CIDER-CL: evaluation report for OAEI 2013. In: Proceedings of the 8th International Conference on Ontology Matching, vol. 1111. CEUR-WS.org (2013)

Software Technology and Model Transformations

Transforming UML Diagrams to YAWL Models for Business Processes Analysis

Aissam Belghiat[1,2(✉)] ⓘ, Dalal Oukhaf[1], and Allaoua Chaoui[2] ⓘ

[1] Department of Computer Science, University of Mohamed Seddik Benyahia, Jijel, Algeria
{aissam.belghiat,dalal.oukhaf}@univ-jijel.dz
[2] MISC Laboratory, University of Constantine 2-Abdelhamid Mehri, Constantine, Algeria
allaoua.chaoui@univ-constantine2.dz

Abstract. Business processes modeling and verification are become essential to master and guarantee organization evolution. UML Activity diagrams have been used for this purpose, although they were not designed for that at the beginning. These diagrams lack for formal semantics which prohibits their verification. YAWL is a language developed for modeling workflows. It has a formal semantics based on Petri nets, and it is supported by open source toolsets. Transforming UML activity diagrams to YAWL is very beneficial. It allows profiting from the intuitiveness and widespread use of UML activity diagrams for modeling business processes and workflows, on one hand, and enabling their verification by using YAWL tools on the other hand.

Keywords: Business process · MDE · UML activity diagram · YAWL

1 Introduction

The business process is the core concept of the process-driven technology. It is a set of orderly linked activities which realize collectively a business objective in an organization [1]. A workflow is the automation of all or a part of a business process, involving humans and automated tasks in a distributed and computerized organizational environment, during which information circulates from one activity to another (or from one participant to another), according to a set of management rules [1]. The workflow permits to rationalize, coordinate and control a business process. The workflow management system (WFMS) is the set of software tools for defining processes, performing workflows, and administering and monitoring process instances [2].

UML (Unified Modeling Language) activity diagrams [3] are used for modeling the behavior of object-oriented systems. They are used, also, to specify business processes and workflows, because they offer the necessary elements and mechanisms to do that easily. However, UML does not provide formal semantics for its diagrams, so, it cannot support formal verification of their models. On the other side, the YAWL (Yet Another Workflow Language) [4] is a language developed for modeling workflows, and it is

doted by a formal semantics using Petri net. Furthermore, it is supported by open source toolsets which allow different verification features, such as the YAWL tool [4].

UML and YAWL have complementary characteristics. UML can be used for workflow modeling since its widespread using and intuitiveness, while YAWL can be used for verification. Thus, transforming UML activity diagrams towards YAWL models yields many benefits, as it allows interconnecting two independent modeling communities, and allowing verification of business process models specified in UML.

In this paper, a transformation between UML activity diagrams and YAWL is presented. It is done using a combined meta-modeling and graph grammar technique implemented in the AToM^3 tool [6]. We have proposed a meta-model for UML activity diagrams and another meta-model for YAWL models. Then, we have generated an environment for both formalisms. Each environment is expected to support the visual modeling of activity diagrams and YAWL models respectively. Afterwards, we have proposed and developed two graph grammars. The first one is a model to model transformation that is used to automatically mapping each UML activity diagram described in the canvas of our AToM^3 integrated tool, to its corresponding YAWL model. The second one is a model to text graph grammar that transforms the YAWL models resulted to their XML representation, i.e. ".yawl" files. The derived ".yawl" file is uploaded automatically in the YAWL tool to start analysis and verification.

The rest of the paper is organized as follows. In Sect. 2, some related works are exposed. In Sect. 3, a background of the work is presented. In Sect. 4, the proposed approach is explained. In Sect. 5, the approach is illustrated by an example. Section 6 ends the paper and gives some perspectives.

2 Related Works

Since we started using activity diagrams for modeling workflows, they were a large amount of works which tried to bridging the gap between them at a side, and the business process modeling languages at the other side. In [7, 8], the authors tackled the comparison and the convergence between UML activity diagrams and BPMN. In [9, 10], a mapping of activity diagrams to BPEL is done.

Moreover, many other papers in the literature have tried to transform activity diagrams to formal methods known by their suitability for modeling business processes. The aim was to enable the formal verification of those diagrams. The most formal methods used are process algebra and Petri nets, such as the mapping to the Pi-calculus done in [11–13], to the CSP in [14, 15], and to the Petri nets in [16, 17].

Likewise, there were multiple contributions around mapping business process models to YAWL in order to benefit from its expressiveness. In [18], the authors have proposed a transformation of UML activity diagrams to the YAWL language, to connect the two formalisms. In [19], the authors have transformed EPC (Event-driven Process Chain) to YAWL. In [20, 21], the business process modeling notation (BPMN) is mapped to YAWL, while the BPEL is transformed to YAWL in [22].

In this paper, we are interested by transforming UML activity diagrams to YAWL language. We think, first of all, such a passage will bridge the gap between two communities of workflows modeling, i.e. the UML community and the YAWL community.

This allows the usage of a widespread intuitive formalism, i.e. UML activity diagrams, for workflows modeling. Besides, it makes it possible to verify business process models specified in UML activity diagrams using YAWL tools. The work done in [18] is the very closest to ours. However, our contribution differs from it by using the graph transformation that permits developing a tool that allows the automation of the translation easily.

3   Background

3.1   UML Activity Diagram

UML activity diagrams [3, 5] are developed for modeling dynamic behavior of object-oriented systems. They are immediately used in organizational processes (i.e. workflows) due to their rich expressiveness. Actually, they offer an excellent tool that provides a set of graphical elements for representing the sequence of activities and actions, with support for choice, iteration and concurrency which make them very suitable for workflow modeling. Figure 1 shows the main elements of the formalism.
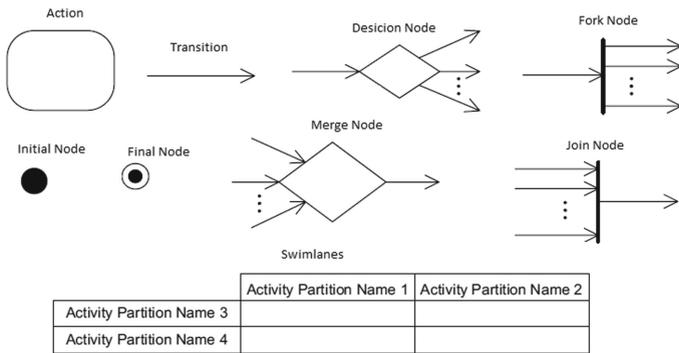
Fig. 1. The main elements of activity diagrams

3.2   YAWL

YAWL (Yet Another Workflow Language) [4] is a business process modeling language based on workflow models. It is developed to remedy the expressiveness limits of Petri nets in terms of control flow modeling. The architecture of this language includes several components, the main component include a YAWL designer, a YAWL engine and a YAWL repository. In YAWL, a workflow model is made up of task, conditions and a flow relationship between tasks and conditions. Figure 2 shows the main elements of the formalism.
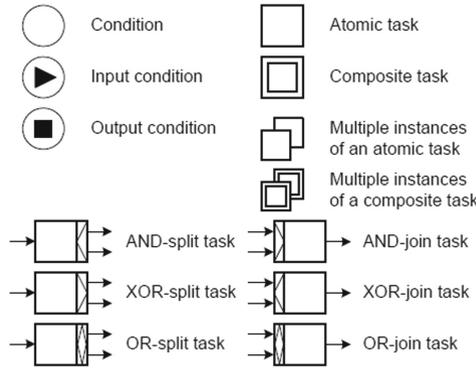
Fig. 2. The main elements of YAWL [4].

3.3   Meta-modeling and Graph Transformation in AToM^3

AToM^3 (A Tool for Multi-formalism Meta-Modeling) [6] is a modeling tool developed using Python at the university of McGill. It uses a combination of meta-modeling and graph transformation to implement model transformations.

Meta-modeling in AToM^3 is used to model the abstract syntax and the concrete syntax of any formalism.

Graph transformation in AToM^3 is used to realize the translation between formalisms using graph grammars [24]. A graph grammar consists of three essential compartments; an initial action, a set of rules and a final action. The initial action is used to provide necessary information before launching the execution of the rules, while the final action is used to provide necessary information after ending the execution of the rules. Regarding the rules, each one has essentially two compartments; a graph on its left hand side (LHS) and a graph on its right hand side (RHS). During the execution, a matched LHS to the source graph will be replaced by its RHS. A condition may be present on the rule, and it must be satisfied to apply it. An action could also be performed when the rule is carried out. A rewriting system allows applying iteratively the matching rules to the source graph until no more rules can be applicable. The rules respect an order of execution using a user-assigned priority.

4   The Proposed Approach

The approach consists of using a combined meta-modeling/graph grammar technique. We propose two meta-models for UML activity diagrams and YAWL respectively, which contain their syntax and semantic. We generate from these meta-models two environments which allow modeling activity diagrams and YAWL models. Then, two graph grammars are developed to realize the transformation. The first one transforms the UML activity diagram meta-model to the YAWL meta-model. The second one translates the YAWL meta-model to its textual representation.

4.1 Activity Diagrams Meta-model

The proposed meta-model is named "UML_AD_META". It is composed of 8 classes and 6 associations developed by the meta-formalism *"CD_classDiagramsV3"*. It provides an integrated AToM³ tool that provides the necessary elements to model activity diagrams. The abstract syntax of the meta-model is given in Fig. 3, which holding also its concrete syntax implicitly by drawing the appearance of each class.
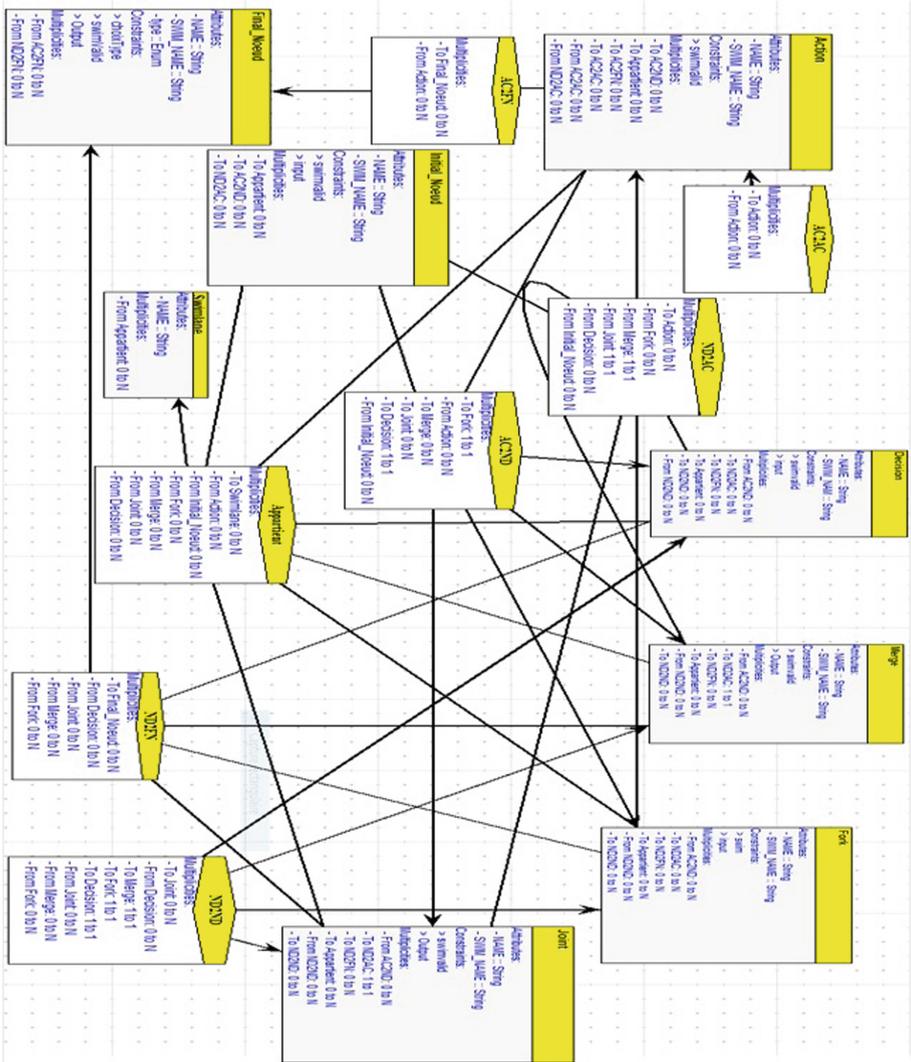
Fig. 3. Meta-model of activity diagrams.

4.2   Meta-model of YAWL Models

The proposed meta-model is named "YAWL_META". It is composed of 10 classes and 8 associations developed by meta-formalism *"CD_classDiagramsV3"*. It provides an integrated AToM³ tool which offers the necessary tools to model YAWL formalisms. The abstract syntax of the meta-model is given in Fig. 4, which holding also its concrete syntax implicitly by drawing the appearance of each class.
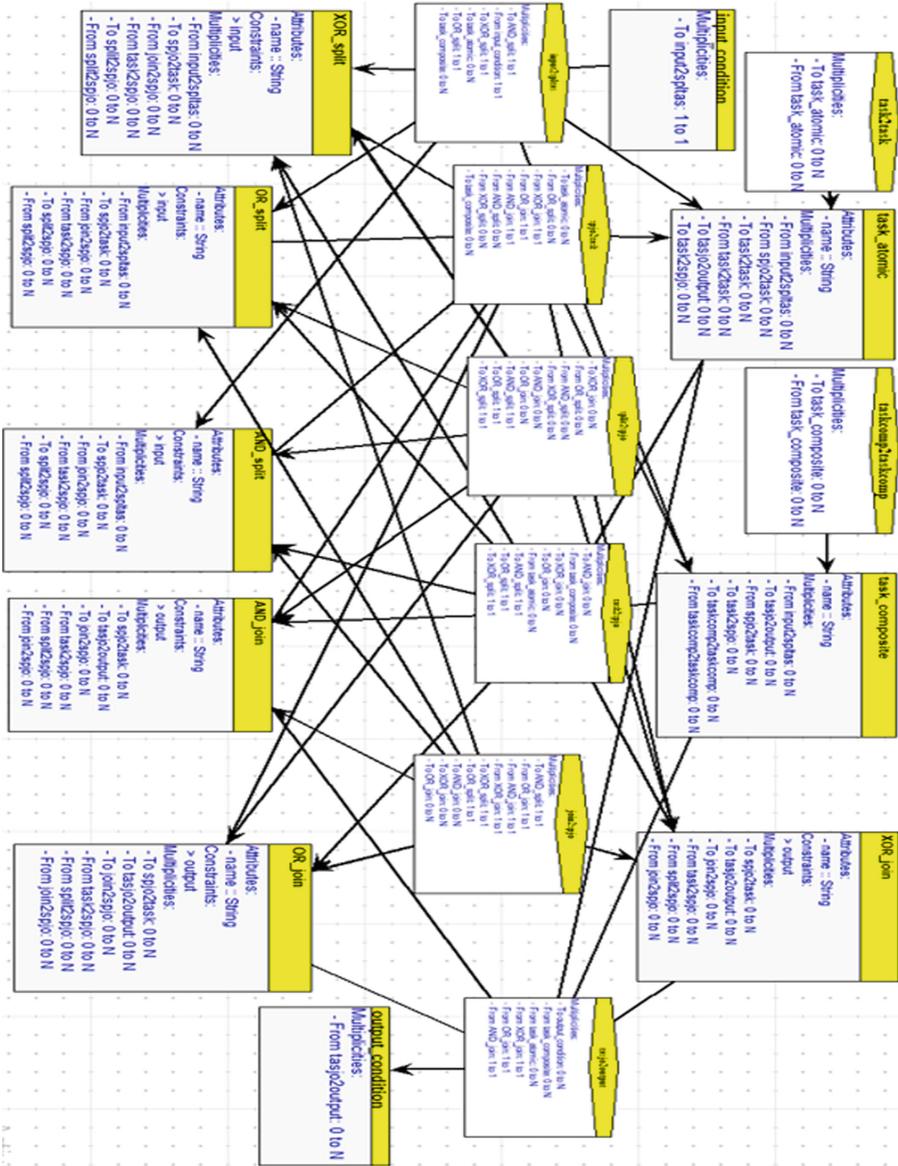
Fig. 4. Meta-model of YAWL models.

4.3   Transformation Rules

The transformation from UML activity diagrams to YAWL models requires the proposition of some correspondence rules between them. This proposition represents a crucial step in the construction of the transformation because it implies a good understanding of both formalisms. Table 1 summarizes the passage.

Table 1. The essential transformation rules.

Activity diagram	YAWL
Initial node	Entry condition
Final node	Exit condition
Action	Atomic task
Decision	XOR-Split
Merge	XOR-Join
Fork	AND-Split
Join	AND-Join

### 4.4   The First Graph Grammar, UML-AD to YAWL

To implement the rules seen previously, we have proposed a graph grammar named *"dgrmactv2yawl" and* composed of 45 rules. We show in Table 2 some details of this implementation; each rule has a name, a priority, a condition, an LHS, an RHS and an action. Due to space constraints, we present here an overview of the graph grammar that enables understanding the translation.

Table 2. The UML-AD to YAWL graph grammar.

Rule1: IN2IC, Priority=1, Condition= if not IN2IC	Rule2: AC2AT, Priority=2, Condition= if not AC2AT
An initial state in an activity diagram is transformed to an entry condition in YAWL.	An action in an activity diagram is transformed to an atomic task in YAWL, the action name becomes the atomic task name.
Rule3: JN2AJ, Priority=3, Condition= if not JN2AJ	**Rule4: FR2AS, Priority=4, Condition= if not FR2AS**
A join node in an activity diagram is transformed to an AND-Join node in YAWL.	A fork node in an activity diagram is transformed to an AND-Split node in YAWL
Rule5: DC2XS, Priority=6, Condition= if not DC2XS	**Rule6: MR2XJ, Priority=8, Condition= if not MR2XJ**
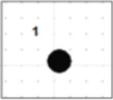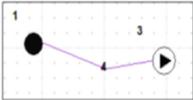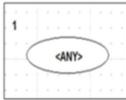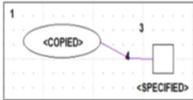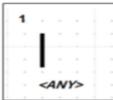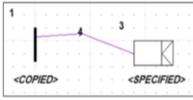
(continued)

Table 2. (*continued*)

LHS	RHS	LHS	RHS
1 `<ANY>`	:=	1 3 `<COPIED>` `<SPECIFIED>`	1 `<ANY>`	:=	1 3 `<COPIED>` `<SPECIFIED>`

A decision node in an activity diagram is transformed to an XOR-Split node in YAWL.	A merge node in an activity diagram is transformed to an XOR-Join node in YAWL

Rule7: IA2IT, Priority=9, Condition= if not IA2IT | **Rule8: deleteIA, Priority=10, Condition= if not deleteIA**

LHS	RHS	LHS	RHS

An initial state linked to an action is transformed to an input condition linked to an atomic task in YAWL. There are rules similar to this rule, the initial state linked to a fork (IF2CA) and the initial state linked to a decision: (ID2CX).	To delete an initial state in an activity diagram, the RHS is left empty.

Rule9: AA2TT, Priority=11, Condition= if not AA2TT | **Rule10: DA2XT, Priority=16, Condition= if not DA2XT**

LHS	RHS	LHS	RHS

An action linked to another action is transformed to an atomic task linked to another atomic task in YAWL. The other similar rules to this rule are: an action linked to a decision (AD2TX), an action linked to a fork (AF2TA), an action linked to a merge (AM2TX) and an action linked to a join (AJ2TA).	A decision linked to an action is transformed to an XOR-split linked to an atomic task in YAWL. The other similar rules to this rule are: a merge linked to an action (MA2TX), a fork linked to an action (FA2AT) and a join linked to an action (JA2AT).

Rule11: FF2AA, Priority=21, Condition= if not FF2AA | **Rule12: DD2XX, Priority=20, Condition= if not DD2XX**

LHS	RHS	LHS	RHS

A fork linked to a fork in an activity diagram is transformed to an AND-Split node linked to an AND-Split in YAWL. The other similar rules to this rule are: a fork linked to a decision (FD2AX), a fork linked to a join (FJ2AA), a fork linked to a merge (FM2AX).	A decision linked to a decision in an activity diagram is transformed to an XOR-Split linked to an XOR-Split in YAWL. The other similar rules to this rule are: a decision linked to a fork (DF2XA), a decision linked to a join (DJ2XA), a decision linked to a merge (DM2XX).

Rule13: DDDYY, Priority=30, Condition= if not DDDYY | **Rule14: MM2XX, Priority=35, Condition= if not MM2XX**

(*continued*)

Table 2. (*continued*)

LHS	RHS	LHS	RHS
:=		:=	
A join linked to a join in an activity diagram is transformed to an AND-Join linked to an AND-Join in the YAWL model. The other similar rules to this rule are: a join linked to a fork (JF2AA), a join linked to a join (JD2AX), a join linked to a merge (JM2AX).		A merge linked to a merge in an activity diagram is transformed to an XOR-Join linked to an XOR-Join in YAWL. The other similar rules to this rule are: a merge linked to a fork (MF2XA), a merge linked to a join (MJ2XA), a merge linked to a decision (MD2XX).	

Rule15: deleteDC, Priority=36, Condition= if not deleteDC		Rule16: deleteFR, Priority=37, Condition= if not deleteFR	
LHS	RHS	LHS	RHS
:=		:=	
To delete a decision in an activity diagram, the RHS part is left empty.		To delete a fork node in an activity diagram, the RHS part is left empty.	

Rule17: EF2OC, Priority=38, Condition= if not EF2OC		Rule18: AF2TO, Priority=39, Condition= if not AF2TO	
LHS	RHS	LHS	RHS
:=		:=	
A final activity node or a final flow node in an activity diagram is transformed to an exit condition in YAWL		An action linked to a final node is transformed to an atomic task linked to an exit condition in YAWL. Other similar rules to this rule are: a join linked to a node (JF2AO) and a merge linked to a node (MF2XO).	

Rule19: deleteFE, Priority=42, Condition= if not deleteFE		Rule20: deleteAC, Priority=43, Condition= if not deleteAC	
LHS	RHS	LHS	RHS
:=		:=	
To delete a final node in an activity diagram, the RHS part is left empty.		To delete an action in an activity diagram, the RHS part is left empty.	

Rule21: deleteMR,Priority=44, Condition= if not deleteMR		Rule22: deleteJN, Priority=45, Condition= if not deleteJN	
LHS	RHS	LHS	RHS
:=		:=	
To delete a merge in an activity diagram, the RHS part is left empty.		To remove a join node in an activity diagram, the RHS part is left empty.	

4.5   The Second Proposed Graph Grammar, YAWL to Text

In order to produce the textual specification of the YAWL models, a second graph grammar "*yawl2xml_GG_exec*" is proposed. It has an initial action, 12 rules, and a final action. Applying this grammar to a YAWL model leads to the generation of a file which contains its XML description. An XML schema provided in [23] shows how to switch

from the YAWL meta-model to its XML representation ".yawl". We present only some rules, the others are similar.

- **The initial action:** In the initial action of the graph grammar, we use Python to create a file named *"fichyawl.yawl"* with sequential access (i.e. the data are saved in the le one after the other) and storing the generated code (Fig. 5).

```
obFichier = open('fichyawl.yawl','w')
obFichier.write('\n')
obFichier.write('<?xml version="1.0" encoding="UTF-8"?>')
obFichier.write('\n')
obFichier.write('<specificationSet
xmlns="http://www.yawlfoundation.org/yawlschema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="4.0"
xsi:schemaLocation="http://www.yawlfoundation.org/yawlschema
http://www.yawlfoundation.org/yawlschema/YAWL_Schema4.0.xsd">')
                    .
                    .
                    .
obFichier.close()
```

Fig. 5. Extract of the initial action.

- **Set of rules:** the graph grammar is made up of 12 rules. Each rule is characterized by a name, a property, a condition and an action, such as the rule ICT2ICT below.

  - Rule 1: ICT2ICT (priority: 1) (Fig. 6)

Condition :

```
node = self.getMatched(graphID, self.LHS.nodeWithLabel(2))
return not hasattr(node, "ICT2ICT_executed")
```

Action :

```
node1 = self.getMatched(graphID, self.LHS.nodeWithLabel(1))
node = self.getMatched(graphID, self.LHS.nodeWithLabel(2))
taskname = node.name.getValue()
node.ICT2ICT_executed = True

obFichier = open('fichyawl.yawl','a')
obFichier.write('<inputCondition id="InputCondition"> \n  <flowsInto> \n
<nextElementRef id= " '+taskname+' "/> \n </flowsInto> \n  </inputCondition>')
```
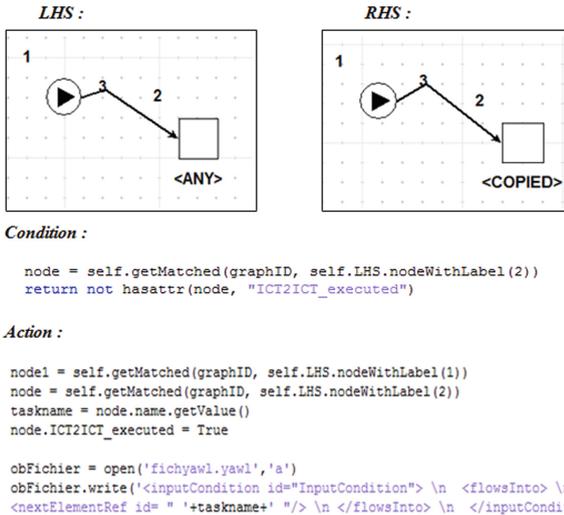
Fig. 6. Mapping a YAWL initial condition linked to a task to xml.

– **The final action** (Fig. 7):

```
obFichier = open('fichyawl.yawl','a')
obFichier.write('<outputCondition id="OutputCondition" />')
obFichier.write('\n')
obFichier.write('</processControlElements>')
.
.
.

obFichier.close()
```

Fig. 7. Extract of the final action.

5 Example

In order to illustrate our proposed approach, we take an example of an activity diagram inspired by [5] that models the use of a bank ATM by a client.

First, the client inserts his/her card and enters his/her card code. If it is valid, the client chooses the operation. If the code is invalid the client must reenter the code. If the client cancels the operation, the card will be returned to him/her. The operations available are either withdrawal or deposit of money, and the client must enter, in both, the amount of money, and selects the account. Then, if it is to withdraw, he/she requests a withdraw authorization and stays waiting while watching an advertisement in parallel. Getting the authorization, the client will be served money. If it is to deposit, he/she inserts money bills. At the end of the two operations, the card is returned to the client (Fig. 8).

Applying the approach, we must use our integrated tool AToM^3 to describe the activity diagram, the figure below shows its specification. The first graph grammar is appeared and it is ready to start transforming the UML activity diagram (Fig. 9).

After the execution of the first graph grammar, we obtain the YAWL model in Fig. 10. Then, the second graph grammar is appeared and it is ready to start translating the resulted YAWL model to its XML format.

By applying the second graph grammar, we obtain a XML file ".yawl" which contains all the information existing in the graphical model. This file is loaded to the YAWL tool [4] to start the verification. Figure 11 shows an extract of the file.

**Fig. 8.** Example of an activity diagram [5].

**Fig. 9.** Example of an activity diagram in the integrated tool.

Fig. 10. The resulted YAWL model.

Fig. 11. An extract of the generated ".yawl" file

6 Conclusion

In this paper, we have presented an automatic transformation of UML activity diagrams to the YAWL models. The proposed approach consists of using a combined meta-modeling and graph grammar technique implemented in the AToM^3 tool. The meta-modeling is used to build the activity diagrams environment and the YAWL environment. Then, two graphs grammars are used. The first one tackled the transformation of the UML

activity graphical diagrams to the YAWL graphical models, while the second one has been used to derive the textual code of the generated YAWL graphical models. This code is uploaded automatically in YAWL supported toolset to start the verification task.

In future work, we plan to consider other advanced elements of the UML activity diagrams, and other diagrams which could offer more information, so that the translation will be optimal.

References

1. Workflow Management Coalition: Workflow management coalition specification-terminology & glossary (WFMC-TC-1011) (1999). www.wfmc.org
2. Workflow Management Coalition: The workflow reference model (WFMC-TC-1003) (1995). www.wfmc.org
3. UML (Unified Modeling Language): Superstructure, v2.5. http://www.omg.org/
4. Yawl (Yet Another Workflow Language). http://www.yawlfoundation.org/
5. Laurent Audibert: Cours UML. https://laurent-audibert.developpez.com/Cours-UML/?page=diagramme-activites
6. AToM³ Home Page. http://atom3.cs.mcgill.ca/
7. Geambaşu, C.V.: BPMN vs UML activity diagram for business process modeling. Account. Manag. Inf. Syst. **11**(4), 637–651 (2012)
8. Peixoto, D., Batista, V., Atayde, A., Borges, E., Resende, R., Pádua, C.I.P.S.: A comparison of BPMN and UML 2.0 activity diagrams. In: VII Simposio Brasileiro de Qualidade de Software, vol. 56, p. 012010 (May 2008)
9. Gardner, T.: UML modelling of automated business processes with a mapping to BPEL4WS (2003)
10. Ouyang, C., et al.: Translating standard process models to BPEL. In: Advanced Information Systems Engineering, pp. 417–432 (2006)
11. Yang, D., Zhang, S.S.: Using π-calculus to formalize UML activity diagrams. In: 10th International Conference and Workshop on the Engineering of Computer-Based Systems, pp. 47–54. IEEE Computer Society (2004)
12. Lam, V.S.: On π-calculus semantics as a formal basis for UML activity diagrams. Int. J. Softw. Eng. Knowl. Eng. **18**(04), 541–567 (2008)
13. Belghiat, A., Chaoui, A.: A graph transformation of activity diagrams into pi-calculus for verification purpose. In: Proceedings of the 3rd Edition of the International Conference on Advanced Aspects of Software Engineering (ICAASE18), Constantine, Algeria, 1–2 December (2018). http://ceur-ws.org
14. Bolton, C., Davies, J.: On giving a behavioral semantics to activity graphs. In: UML 2000 Workshop Dynamic Behavior in UML Models: Semantic Questions (2000). http://www.disi.unige.it/person/ReggioG/UMLWORKSHOP/ACCEPTED.html. Accessed 26 July 26 2006
15. Bolton, C., Davies, J.: Activity graphs and processes. In: 2nd International Conference on Integrated Formal Methods, LNCS 1945, pp. 77–96 (2000)
16. Storrle, H., Hausmann, J.H.: Towards a formal semantics of UML 2.0 activities. In: German Software Engineering Conference 2005 (2005). http://www.pst.ifi.lmu.de/personen/stoerrle/V/AD-11-Limits.pdf. Accessed 26 July 2006

17. Barros, J.P., Gomes, L.: Actions as activities and activities as Petri nets. In: UML 2003 Workshop on Critical Systems Development with UML (2003)
18. Han, Z., Zhang, L., Ling, J.: Transformation of UML activity diagram to yawl. In: Enterprise Interoperability IV, pp. 289–299. Springer, London (2010)
19. Mendling, J., Moser, M., Neumann, G.: Transformation of yEPC Business Process Models to YAWL. ACM, New York (2006)
20. Hong, Y.J., et al. Transformation of BPMN to YAWL. In: 2008 International Conference on Computer Science and Software Engineering (2008)
21. Decker, G., et al.: Transforming BPMN diagrams into YAWL nets. In: Business Process Management, pp. 386–389 (2008)
22. Brogi, A., Popescu, R.: From BPEL processes to YAWL workflows. In: Web Services and Formal Methods, pp. 107–122 (2006)
23. https://github.com/yawlfoundation/yawl/blob/master/schema/YAWLSchema:xsd
24. Rozenberg, G.: Handbook of Graph Grammars and Computing by Graph Transformation. World Scientific, Singapore (1999)

Configuration-Dependent Stochastic Reward Nets

Samir Tigane$^{(\boxtimes)}$, Laid Kahloul , and Samir Bourekkache

LINFI Laboratory, Biskra University, Biskra, Algeria
{s.tigane,s.bourekkache}@univ-biskra.dz, kahloul2006@yahoo.fr

Abstract. Nowadays, manufacturing companies integrate several advanced technologies such as Internet of things, cyber-physical systems, smart factories, etc. to adapt themselves, as soon as possible, to the volatile market demands as well as to increase their throughput and profit. At the design level, the use of classical formal approaches such as Petri nets (either low or high-level) becomes neither convenient nor useful in the modeling and verification of modern manufacturing systems due to their rigid structure that cannot handle the reconfigurability of such systems. In this paper, we introduce configuration-dependent stochastic reward nets formalism that allows designers to study the reconfiguration in smart factories. The proposed formalism transforms configuration-dependent stochastic reward nets into stochastic reward nets in order to use the existing tools proposed for stochastic reward nets.

Keywords: Petri nets · Stochastic reward nets · Reconfigurable systems · Graph transformations · Modeling and verification

1 Introduction

In the present time, companies face fierce and volatile market environment. They are forced to make great efforts to be more competitive and maximize their incomes [14]. At this end, manufacturing companies integrate several advanced technologies such as Internet of things (IoT), cyber-physical systems (CPSs), smart products, smart factories, etc. to adapt themselves, as soon as possible, to the volatile market demands; and to make a small batch even with one product can be manufactured profitably to meet the individual customer needs [12]. This new industrial revolution is known as Industry 4.0 [21].

Reconfigurability is a key-feature of Industry 4.0 as it allows the adaptation in real-time to new demands at the last minute and the introduction of new products without interrupting the production lines [14].

Brettel et al. [3] have assigned 330 research articles related to Industry 4.0 to sub-topics such as mass customization, flexible manufacturing systems (FMSs), reconfigurable manufacturing systems (RMSs), self-optimization, etc. With 146 counts for both FMSs and RMSs and 46 for supply-chain flexibility, the reconfigurability finds wide attention in this field which reflects its significant role.

Petri nets (PNs) have been considered as a formal tool in [1,2,5,8,17,23] to design supervisors that prevent either FMSs or RMSs from reaching a deadlock. The authors in [9] use PNs to evaluate the information security risk in CPSs. To address the product traceability issue in cyber-physical manufacturing systems, Petri nets are used in [13] to formulate dynamic manufacturing processes and allow traceability analysis. The work proposed in [27] describes a PNs-based method to propose a control architecture for manufacturing systems in Industry 4.0. The paper presented in [10] uses IoT and timed colored Petri nets (TCPNs) to propose a simulation-based self-adaptive collaboration between production and logistics in a CPS. The authors in [19] and [20] model and simulate the flexibility aspect of production systems and their availability in Industry 4.0 based on extended colored stochastic Petri nets (ECSPNs). The work described in [22] develops an approach for real-time operation planning for distributed manufacturing networks, where hierarchical TCPNs (HTCPNs) formalism is used.

Nevertheless with the great development of smart factories, the use of PNs (either low or high-level) becomes neither intuitive nor sufficient in this field due to their rigid structures that impede their use for studying reconfigurability in modern manufacturing systems.

Indeed, manufacturing systems that can cope with volatile markets, individual customer needs, continuous product variation, high productivity, high performance, flexible machines and reconfigurable structures are expected to be remarkably complex [6]. The design and implementation of such complex manufacturing systems are challenging issues and their complexity is growing more and more. Therefore, designers should dispose of the convenient approaches, models and tools required to manage this complexity [12]. To satisfy the new designers' needs, researchers have introduced the reconfigurability in several classes of PNs.

Net rewriting systems (NRSs), reconfigurable Petri nets (RPNs) and marked controlled-reconfigurable Petri nets (MC-RPNs) formalisms are defined in [18] to introduce dynamic structures in PNs. Improved NRSs (INRSs) [16] restrict NRSs-based reconfiguration to preserve important PNs properties after any INRSs-based reconfiguration. The authors in [15] exploit reconfigurable object nets (RONs) for the design, simulation and verification of RMSs. MC-RPNs are extended in [29] and [28] to deal with reconfigurability in stochastic Petri nets (SPNs) and generalized stochastic Petri nets (GSPNs), where the reconfiguration is still limited to the topology level.

The work described in [30] proposed a reconfigurable formalism, called dynamic GSPNs (D-GSPNs), which allows modeling of all transformation forms, and transforming D-GSPNs into equivalent GSPNs, hence, D-GSPNs can be analyzed based on their equivalent GSPNs.

In [31] and [32], restricted reconfiguration forms for GSPNs are proposed by which several properties are preserved after each reconfiguration. Hence, preserved properties are decidable even if the system can be structurally unbounded.

Stochastic reward nets (SRNs) [7] are a generalization of stochastic Petri nets that can provide guards, marking-dependent arc multiplicities, etc. often

resulting in more compact nets (modeling level), and specify output measures as reward-based functions, for the evaluation of complex systems (analysis level).

SRNs have found a wide range of applications in the modeling, verification and performance evaluation of dynamic systems [24]. The authors in [4] propose an analytic model for the experience-availability analysis of cloud computing services using SRNs. The work presented in [25] examines and evaluates the impact of various failures on network infrastructures based on SRNs. An SRN-based approach dealing with the availability analysis of cloud computing services considering the failure of servers and virtual machines is provided in [26]. Matrix rewriting SRNs with fuzzy parameters (MFRSRNs) [11] introduce the reconfiguration to fuzzy SRNs (FSRNs) using rewriting rules, in which an FSRN model of a given system is first established (for the performance evaluation) and then it is folded into an MFRSRN (yielding a compact view of the system).

Except for [11], to the best of our knowledge, there is no other work in the literature that treats the reconfigurability in SRNs to study the reconfiguration in modern reconfigurable systems.

In this paper, we propose a new formalism, called configuration-dependent SRNs (CD-SRNs), that introduces reconfigurability in SRNs. The proposed formalism is inspired from R-SPNs defined in [29]. Moreover, we provide an algorithm that unfolds CD-SRNs into semantically-equivalent SRNs. Therefore, designers can use existing tools to perform the analysis steps.

The remainder of this paper is organized as follows. Section 2 recalls the underlying formalism, namely, SRNs. Section 3 describes formal definition of the proposed extension. Section 4 provides the transformation steps of CD-SRNs into basic SRNs. The semantic equivalence between any given CD-SRN and its proposed transformation into SRN is proven in Sect. 5. In Sect. 6, we present an illustrative example to demonstrate the use of the new formalism. Finally, we conclude the paper in Sect. 7.

2   Stochastic Reward Nets

SRNs provide guards, marking-dependent arc multiplicities, a more general approach to the specification of priorities, and the ability to decide in a marking-dependent fashion whether the firing time of a transition is exponentially distributed or zero, often resulting in more compact nets (modeling level), and specify output measures as reward-based functions, for the evaluation of complex systems (analysis level). The formal definition of an SRN is given as follows.

SRN is a tuple $G = \langle P, T, D^-, D^+, D^o, g, >, \mu_0, \lambda, w, \rho \rangle$ where:

1. P is a finite set of places.
2. T is a finite set of transitions, $P \cap T = \emptyset$.
3. $\forall\, p \in P, \forall\, t \in T, D^-_{p,t} : \mathbb{N}^{|P|} \longrightarrow \mathbb{N}, D^+_{p,t} : \mathbb{N}^{|P|} \longrightarrow \mathbb{N}$ and $D^o_{p,t} : \mathbb{N}^{|P|} \longrightarrow \mathbb{N}$ are the marking-dependent multiplicities of the arc from p to t, the arc from t to p, and the inhibitor arc from p to t, respectively. A transition t is said to be arc-enabled in marking μ iff $\forall\, p \in P, D^-(p,t)(\mu) \leq \#(p,\mu) \wedge (D^o(p,t)(\mu) > \#(p,\mu) \vee (D^o(p,t)(\mu) = 0)$, where $\#(p,\mu)$ is the token number of p at μ.

4. $\forall\ t \in T, g_t : \mathbb{N}^{|P|} \longrightarrow \{true, false\}$ is the guard for transition t. If $g_t(\mu) = false$, t is disabled in μ.
5. $>$ is a transitive and irreflexive relation imposing a priority among transitions. In a marking μ, t_1 is marking-enabled iff it is arc-enabled, $g_{t_1}(\mu) = true$, and no other transition t_2 exists such that $t_2 > t_1$, t_2 is arc-enabled and $g_{t_2}(\mu) = true$.
6. μ_0 is the initial marking.
7. $\forall\ t \in T, \lambda_t : \mathbb{N}^{|P|} \longrightarrow \mathbb{R}^+ \cup \infty$ is the rate of the exponential distribution for the firing time of transition t. If $\lambda_t(\mu) = \infty$, the firing time of t in μ is zero i.e. t is an immediate transition.
8. $\forall\ t \in T, w_t : \mathbb{N}^{|P|} \longrightarrow \mathbb{R}^+$ describes the weight assigned to the firing of enabled immediate transition t, whenever its rate λ_t evaluates to ∞.
9. $\rho : \mathbb{N}^{|P|} \longrightarrow \mathbb{R}$ is a reward rate function over the underlying stochastic process, such that $\rho(\mu)$ is a reward rate at which the reward is accumulated when the marking is μ. For steady state distribution $[\pi_0, \pi_1, \ldots, \pi_n]$ (the probabilities of being in tangible markings $\{\mu_0, \mu_1, \ldots, \mu_n\}$), the expected reward rate $\mathcal{R}$ is given by $E[\mathcal{R}] = \sum_{i=0}^{n} = \rho(\mu_i)\pi_i$. Recall that a marking is called tangible if it does not enable immediate transitions, otherwise, it is called vanishing.

3 Configuration-Dependent Stochastic Reward Nets

In order to study reconfiguration in SRNs, this paper proposes a new formalism, called configuration-dependent SRNs (CD-SRNs), that supports reconfigurability and structural/behavioral analysis proposed for SRNs.

We define a CD-SRN as a tuple $N = \langle \mathcal{P}, \mathcal{T}, \mathcal{C}, (\mathcal{R}, >), \gamma_0 \rangle$, such that:

1. $\mathcal{P}$ is a non-empty and finite set of places.
2. $\mathcal{T}$ is a non-empty and finite set of transitions.
3. $\mathcal{C} = \{C_0, C_1, \ldots, C_n\}$ is a finite set of SRNs each of which models a configuration, such that, $\forall\ C_i \in \mathcal{C}, (P^{C_i} = \mathcal{P})\ \wedge\ (T^{C_i} = \mathcal{T})$, where P^{C_i} and T^{C_i} are sets of places and transitions of C_i, respectively.
4. $\mathcal{R} = \{r_0, \ldots, r_k\}$ is a finite set of reconfiguration rules.
5. $>$ is a transitive and irreflexive relation imposing a priority among rules.
6. γ_0 is an initial configuration of N.

Rule $r \in \mathcal{R}$ is a tuple $r = \langle C_i, C_j, \mathbb{M}^i, \varphi \rangle$, such that:

1. C_i and C_j are source and target configurations of r.
2. $\mathbb{M}^i$ is a required minimum marking of net C_i so that r can be applied.
3. $\varphi : \mathbb{N}^{|\mathcal{P}|} \longrightarrow \mathbb{R}^+$ is an exponentially-distributed applying time of rule r.

A rule $r = \langle C_i, C_j, \mathbb{M}^i, \varphi \rangle$ is applicable to N iff:

1. The current configuration of N is C_i.
2. $\mu(p) \geq \mathbb{M}^i(p), \forall\ p \in C_i$, such that μ is a current marking of C_i.
3. No other rule r' exists such that $r' > r$ and r' is applicable to C_i at $\mathbb{M}^i$.

4   Transformation of CD-SRNs into Basic SRNs

The set of the well-known algorithms and methods that are used to verify the qualitative/quantitative properties of SRNs motivate us to reuse them in the analysis of CD-SRNs. Therefore, we propose an algorithm that unfolds any given CD-SRN N to an equivalent SRN G so that we can reuse the proposed verification techniques of SRNs.

   The reconfiguration of an SRN $S = \langle P, T, D^-, D^+, D^o, g, >, \mu_0, \lambda, w, \mathcal{M} \rangle$ changes $D^-, D^+, D^o, g, >, \lambda, w$ or $\mathcal{M}$. Subsequently, its structure and behavior are changed. Therefore, an equivalent net G obtained by the unfolding algorithm must be able to model these changes. In this scope, we adapt the algorithm proposed for R-SPNs [29] to consider the reconfiguration in SRNs. The proposed algorithm is described as follows.

   Let $\mathring{P} = \{\mathring{p}_0, \mathring{p}_1, \dots, \mathring{p}_n\}$ be a set of associated places with configurations $C_0, C_1, \dots, C_n$, respectively.

   Let $\mathring{\mu}_0 = (1, 0, \dots, 0)$ be an initial making of $\mathring{p}_0, \mathring{p}_1, \dots, \mathring{p}_n$, respectively.

   Let $G = \langle P, T, D^-, D^+, D^o, g, >, \eta_0, \lambda, w, \mathcal{M} \rangle$ be an equivalent SRN of CD-SRN N.

   Let us consider the following notations:

- X_Y denotes the concatenation of vectors X and Y.
- $\eta = \mu\_\mathring{\eta}$ denotes a reachable marking of G where $\mathring{\eta}$ is the marking of places in $\mathring{P}$ associated with configurations in $\mathcal{C}$ and μ is the marking of remaining places.
- $\mu\_\mathring{\eta}^i$ denotes a reachable marking of G where the marking of place $\mathring{p}_i$ associated with configuration C_i is one.
- $\dot{t}$ denotes a representation of transition $t \in \mathcal{T}$ in equivalent SRN G.
- $D_{p,t}^{-,i}$ denotes a marking-dependent multiplicities of an arc from p to t in C_i.
- $D_{p,t}^{+,i}$ denotes a marking-dependent multiplicities of an arc from t to p in C_i.
- $D_{p,t}^{o,i}$ denotes a marking-dependent multiplicities of an inhibitor arc from p to t in C_i.
- g_t^i denotes the guard function of transition t in configuration C_i.
- $>^i$ denotes the priority relation in configuration C_i.
- $>_t^i = >_t^j$ denotes that priority of transition t in configuration C_i is the same as in configuration C_j, i.e., $\forall\, t' \in \mathcal{T}$, if $t >^i t'$ then $t >^j t' \wedge$ if $t' >^i t$ then $t' >^j t$
- λ_t^i (reps. w_t^i) denotes the firing rate (resp. weight) function of transition t in configuration C_i.
- $\mathcal{M}^i$ denotes the measures function in configuration C_i.
- ρ^i denotes a reward rate in configuration C_i.
- $[\mu_0\rangle t_0 [\mu_1\rangle t_1 \dots [\mu_x\rangle r_0^i [\mu_x\rangle t_q \dots [\mu_z\rangle t_p$ denotes a fireable sequence of transitions and rules in a R-SRN N, where $[\mu_0\rangle t_0 [\mu_1\rangle t_1 \dots [\mu_x$ denotes a fireable sequence of transitions in configuration C_i, and $[\mu_x\rangle r_k^j [\mu_x$ denotes an applicable rule r from configuration C_k at marking μ_x to configuration C_j.

   To create an equivalent net $G = \langle P, T, D^-, D^+, D^o, g, >, \eta_0, \lambda, w, \mathcal{M} \rangle$ to a CD-SRN $N = \langle \mathcal{P}, \mathcal{T}, \mathcal{C}, (\mathcal{R}, \succcurlyeq), \gamma_0 \rangle$, first we copy all places belonging to $\mathcal{P}$ with

their initial marking in P. Second, places in $\mathring{P}$ associated with the configurations are appended to P. We note that a token in $\mathring{p}_i$ means that the current system reconfiguration is C_i, such that $\sum_{i=0}^{n} \eta(\mathring{p}_i) = 1$.

Now, we consider the unfolding of transitions, where two cases arise.

First, for each transition t that preserves its priority in each configuration in $\mathcal{C}$ (i.e., $>_t^i = >_t^j, \forall\, C_i, C_j \in \mathcal{C}$) we add transition $\dot{t}$ representing transition t to set T, and for each configuration $C_i \in \mathcal{C}$, let

- $D_{p,\dot{t}}^{-}(\mu\_\mathring{\eta}) = D_{p,t}^{-,i}(\mu)$ if $\eta(\mathring{p}_i) = 1$,
- $D_{p,\dot{t}}^{+}(\mu\_\mathring{\eta}) = D_{p,t}^{+,i}(\mu)$ if $\eta(\mathring{p}_i) = 1$,
- $D_{p,\dot{t}}^{o}(\mu\_\mathring{\eta}) = D_{p,t}^{o,i}(\mu)$ if $\eta(\mathring{p}_i) = 1$,
- $g_{\dot{t}}(\mu\_\mathring{\eta}) = g_t^i(\mu)$ if $\eta(\mathring{p}_i) = 1$,
- $\lambda_{\dot{t}}(\mu\_\mathring{\eta}) = \lambda_t^i(\mu)$ if $\eta(\mathring{p}_i) = 1$,
- $w_{\dot{t}}(\mu\_\mathring{\eta}) = w_t^i(\mu)$ if $\eta(\mathring{p}_i) = 1$,
- for each $(\rho^i, d_t^i) \in \mathcal{M}^i$, $\rho(\mu\_\mathring{\eta}) = \rho^i(\mu)$ if $\eta(\mathring{p}_i) = 1$ and $d_{\dot{t}}(\mu\_\mathring{\eta}) = d_t^i(\mu)$ if $\eta(\mathring{p}_i) = 1$,
- and $\mathcal{I}(t, C_i) = \dot{t}$ denote $\dot{t}$ the image (via the unfolding) of transition t in configuration C_i.

Motivated by the fact that in SRNs the flow relations, guard, firing rate and measure functions are marking-dependent, we opt to model their change by the mean of the marking of places in $\mathring{P}$ that are associated with configurations. Hence, the flow relations, guard, firing rate and measure functions are the same as they are in configuration C_i, if the marking of place $\mathring{p}_i = 1$ (the associated place with configuration C_i).

Now, we consider the priority relation. Let $\mathcal{T}_1 \subset \mathcal{T}$ be the subset of transitions that preserve their priority relations. In net G, we preserve the priority relation among transitions in $\mathcal{T}_1$ as the same as in N. Hence, $\forall\, t, t' \in \mathcal{T}_1, \dot{t} > \dot{t}'$ if $t >^0 t'$ and $\dot{t}' > \dot{t}$ if $t' >^0 t$. We arbitrary use relation $>^0$ in C_0, since $t, t' \in \mathcal{T}_1$ preserve their priority relations in each configuration.

Secondly, if transition t does not preserve its priority, we duplicate the representation of t in G. The purpose of duplicating t is to model the change in its priority via reconfiguration. Let $\succ_t = \bigcup_{C_i \in \mathcal{C}} \{C_j \in \mathcal{C}, >_t^i = >_t^j\}$ be the set of subsets of configurations, where in each subset transition t has the same priority. Then, for each transition t that does not preserve its priority, we duplicate the representation of t in G as many as it changes its priority. Hence, for each $s_k \in \succ_t$ we operate as follows.

- Add transition $\dot{t}_k$ in T, formally $T \longleftarrow T \cup \{\dot{t}_k\}$.
- For each configuration $C_i \in s_k$,
  - $D_{p,\dot{t}_k}^{-}(\mu\_\mathring{\eta}) = D_{p,t}^{-,i}(\mu)$ if $\eta(\mathring{p}_i) = 1$,
  - $D_{p,\dot{t}_k}^{+}(\mu\_\mathring{\eta}) = D_{p,t}^{+,i}(\mu)$ if $\eta(\mathring{p}_i) = 1$,
  - $D_{p,\dot{t}_k}^{o}(\mu\_\mathring{\eta}) = D_{p,t}^{o,i}(\mu)$ if $\eta(\mathring{p}_i) = 1$,
  - $g_{\dot{t}_k}(\mu\_\mathring{\eta}) = (g_t^i(\mu) \wedge \bigvee_{C_j \in s_k} \eta(\mathring{p}_j) = 1)$ if $\eta(\mathring{p}_i) = 1$,
  - $\lambda_{\dot{t}_k}(\mu\_\mathring{\eta}) = \lambda_t^i(\mu)$ if $\eta(\mathring{p}_i) = 1$,

- $w_{\dot{t}_k}(\mu\_\mathring{\eta}) = w_t^i(\mu)$ if $\eta(\mathring{p}_i) = 1$,
- for each $(\rho^i, d_t^i) \in \mathcal{M}^i$, $\rho(\mu\_\mathring{\eta}) = \rho^i(\mu)$ if $\eta(\mathring{p}_i) = 1$ and $d_{\dot{t}_k}(\mu\_\mathring{\eta}) = d_t^i(\mu)$ if $\eta(\mathring{p}_i) = 1$,
- and $\mathcal{I}(t, C_i) = \dot{t}_k$ denote $\dot{t}_k$ the image (via the unfolding) of transition t in configuration C_i.

Let $\mathcal{T}_2 \subset \mathcal{T}$ be the subset of transitions not preserving their priority. For each 3-tuple "transition $t \in \mathcal{T}_2$, transition $t' \in \mathcal{T}$ and configuration $C_i \in \mathcal{C}$", let $\mathcal{I}(t, C_i) > \mathcal{I}(t', C_i)$ if $t >^i t'$ and $\mathcal{I}(t', C_i) > \mathcal{I}(t, C_i)$ if $t' >^i t$.

Finally, each rule $r = \langle C_i, C_j, \mathbb{M}^i, \varphi \rangle \in \mathcal{R}$ is represented by a transition $\dot{r}$. In CD-SRN N, rule r can be applied when the current configuration is C_i, $\mu(p) \geq \mathbb{M}^i(p), \forall\ p \in C_i$ (here μ is the current marking of C_i) and no other applicable rule r' on C_i at marking $\mathbb{M}^i$ exists such that $r' > r$.

Accordingly, each $r = \langle C_i, C_j, \mathbb{M}^i, \varphi \rangle \in \mathcal{R}$ is unfolded as follows.

- Add transition $\dot{r}$ in T, formally $T \longleftarrow T \cup \{\dot{r}\}$,
- $D_{\mathring{p}_i, \dot{r}}^- = 1$. Recall that a token in $\mathring{p}_i$ means that the current configuration is C_i which is required to apply r,
- $D_{\mathring{p}_j, \dot{r}}^+ = 1$ to model a switching to configuration C_j,
- $g_{\dot{r}}(\eta) = 1$ if $\forall\ p \in \mathcal{P}, \eta(p) \geq \mathbb{M}^i(p)$,
- $\lambda_{\dot{r}}(\mu\_\mathring{\eta}) = \varphi(\mu)$,
- for each transition $\dot{t} \in T$ in the equivalent net that is an image of a transition in $\mathcal{T}$, let $\dot{r} > \dot{t}$, that is, applying rules is prioritized on firing transitions,
- for each transition $\dot{r}' \in T$ that is an image of a rule in $\mathcal{R}$, let $\dot{r} > \dot{r}'$ if $r > r'$ and $\dot{r}' > \dot{r}$ if $r' > r$,

5   Proofs

In this section, we aim to prove the equivalence between any given CD-SRN N and its equivalent SRN G obtained by the application of the algorithm described above. To prove the equivalence, we consider the following lemmas and theorem.

Lemma 1. *If the current configuration is C_i, the current marking is μ, transition t in C_i is fireable at μ and μ' is the obtained marking after firing t (in net N), then $\mathcal{I}(t, C_i)$ (the image of t in configuration C_i at G) is fireable at $\eta = \mu\_\mathring{\eta}^i$ in equivalent net G (where $\eta(\mathring{p}_i) = 1$) and the obtained marking is $\mu'\_\mathring{\eta}^i$.*

Lemma 2. *If the current configuration is C_i, the current marking is μ, rule r is applicable from C_i at μ and the obtained configuration is C_j (in net N), then transition $\dot{r}$ (the unfolding of r at G) is fireable at $\eta = \mu\_\mathring{\eta}^i$ in equivalent net G (where $\eta(\mathring{p}_i) = 1$) and the obtained marking is $\mu\_\mathring{\eta}^j$ (where $\eta(\mathring{p}_j) = 1$).*

Proof. We start by proving Lemma 1. Let $T^i \subset T$ be the subset of transitions that are the images of transitions in T_{C_i}. If the current configuration is C_i and $\mathcal{I}(t, C_i) = \dot{t}_i$ is the image of $t \in T_{C_i}$ in G, then we have by definition:

– $D^-_{p,t_i}(\mu\_\mathring{\eta}) = D^{-,i}_{p,t}(\mu)$ since $\eta(\mathring{p}_i) = 1$,
– $D^+_{p,t_i}(\mu\_\mathring{\eta}) = D^{+,i}_{p,t}(\mu)$ since $\eta(\mathring{p}_i) = 1$,
– $D^o_{p,t_i}(\mu\_\mathring{\eta}) = D^{o,i}_{p,t}(\mu)$ since $\eta(\mathring{p}_i) = 1$,
– $g_{i_i}(\mu\_\mathring{\eta}) = g^i_t(\mu)$ since $\eta(\mathring{p}_i) = 1$,
– $\forall\ t' \in T_{C_i}, \mathring{t}_i > \mathcal{I}(t', C_i)$ if $t >^i t'$ and $\mathcal{I}(t', C_i) > \mathring{t}_i$ if $t' >^i t$,

which means that if the current configuration is C_i, the current marking is μ, transition t in C_i is fireable at μ, μ' is the obtained marking after firing t (in net N), then $\mathring{t}_i$ is fireable at $\eta = \mu\_\mathring{\eta}^i$ in equivalent net G and the obtained marking is $\mu'\_\mathring{\eta}^i$.

Lemma 2 is proved as follows. If transition $\mathring{r} \in T$ models rule r from configuration C_i to configuration C_j at marking $\mathbb{M}$, then we have by definition:

– $D^-_{\mathring{p}_i, \mathring{r}} = 1$,
– $D^+_{\mathring{p}_j, \mathring{r}} = 1$,
– $g_{\mathring{r}}(\eta) = 1$ if $\forall\ p \in \mathcal{P}, \eta(p) \geq \mathbb{M}(p)$,
– $\forall\ t \in \mathcal{T},\ \forall\ C_i \in \mathcal{C}, \mathring{r} > \mathcal{I}(t, C_i)$,
– $\forall\ r' \in \mathcal{R}, \mathring{r} > \mathring{r'}$ if $r \gg r'$ and $\mathring{r'} > \mathring{r}$ if $r' \gg r$,

which means that if the current configuration is C_i, the current marking is μ (where $\mu(p) \geq \mathbb{M}(p), \forall\ p \in \mathcal{P}$), rule r is applicable from C_i at μ and the obtained configuration is C_j, then $\mathring{r}$ is fireable at $\eta = \mu\_\mathring{\eta}^i$ in G and the obtained marking is $\mu\_\mathring{\eta}^j$.

Theorem 1. *If $[\mu_0\rangle \overset{o}{t_0}[\mu_1\rangle t_1 \ldots [\mu_x\rangle \overset{o}{r^i_0}[\mu_x\rangle \overset{i}{t_q} \ldots [\mu_z\rangle \overset{j}{t_p}$ is a fireable sequence of transitions and rules in CD-SRN N, then $[\mu_0\_\mathring{\eta}^0\rangle \mathring{t}_{0_0}[\mu_1\_\mathring{\eta}^0\rangle \mathring{t}_{1_0} \ldots [\mu_x\_\mathring{\eta}^0\rangle \mathring{r}^i_0[\mu_x\_\mathring{\eta}^i\rangle\ \mathring{t}_{q_i} \ldots [\mu_z\_\mathring{\eta}^j\rangle \mathring{t}_{p_j}$ is a fireable sequence in G obtained by the unfolding algorithm, and vice versa.*

Proof. In the previous proofs, we prove that in a CD-SRN N:

– if $[\mu\rangle \overset{i}{t}[\mu'$, then $[\mu\_\mathring{\eta}^i\rangle \mathring{t}_i[\mu'\_\mathring{\eta}^i$ in G obtained by the unfolding algorithm, and vice versa,
– if $[\mu_x\rangle \overset{j}{r^j_k}[\mu_x$, then $[\mu_x\_\mathring{\eta}^k\rangle \mathring{r}^j_k[\mu_x\_\mathring{\eta}^j$ in G obtained by the unfolding algorithm, and vice versa.

Hence, if $[\mu_0\rangle \overset{o}{t_0}[\mu_1\rangle t_1 \ldots [\mu_x\rangle \overset{o}{r^i_0}[\mu_x\rangle \overset{i}{t_q} \ldots [\mu_z\rangle \overset{j}{t_p}$ is a fireable sequence of transitions and rules in R-SRN N, then $[\mu_0\_\mathring{\eta}^0\rangle \mathring{t}_{0_0}[\mu_1\_\mathring{\eta}^0\rangle \mathring{t}_{1_0} \ldots [\mu_x\_\mathring{\eta}^0\rangle \mathring{r}^i_0[\mu_x\_\mathring{\eta}^i\rangle \mathring{t}_{q_i} \ldots [\mu_z\_\mathring{\eta}^j\rangle \mathring{t}_{p_j}$ is a fireable sequence in G obtained by the unfolding algorithm, and vice versa.

6   Illustrative Example

In this section, we aim to show the applicability of the proposed approach to an RMS. First, we describe the RMS structure and behavior via a CD-SRN N, then we use the unfolding algorithm to compute its equivalent SRN.

Let us consider an RMS composed of two machines M_1 and M_2; and two buffers $\mathtt{buf_a}$ and $\mathtt{buf_b}$ with a capacity of 15 and 30 spaces, respectively. This RMS produces two type of products P having high quality and P' having low quality. Machine M_1 loads raw material from buffer $\mathtt{buf_a}$ once it is idle, processes the loaded raw material and produces a product P. Machine M_2 loads raw material from buffer $\mathtt{buf_a}$ (resp. $\mathtt{buf_b}$) once it is idle, processes the loaded raw material and produces a product P (resp. product P').

This RMS has two configurations: C_0 and C_1. In configuration C_0, machine M_1 processes single raw material each time. As for machine M_2, it loads raw material form $\mathtt{buf_a}$ if buffer $\mathtt{buf_b}$ is empty, that is, the producing of a product P' is prioritized on producing a product P by M_2. Whereas, at configuration C_1 the producing of a product P by M_2 is prioritized on producing a product P'. Also, M_1 can processes two raw material at once, at C_1.

At first, the system's configuration is C_0 depicted in Fig. 1. The RMS switches to configuration C_1 shown in Fig. 2, when the number of waiting raw materials in $\mathtt{buf_a}$ exceeds ten item.
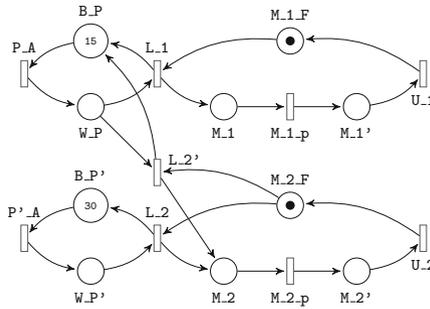

Fig. 1. SRN model of the first configuration.

The interpretation of places and transitions of C_0 and C_1 is given as follows

1. Places:
   (a) B_P (resp. B_P'): The number of tokens, inside this place, models the number of available spaces in buffer $\mathtt{buf_a}$ (resp. $\mathtt{buf_b}$).
   (b) W_P (resp. W_P'): The number of tokens in W_P (resp. W_P') models the number of waiting raw materials in buffer $\mathtt{buf_a}$ (resp. $\mathtt{buf_b}$).
   (c) M_1 (resp. M_1'): A token in M_1 (resp. M_1') means that M_1 has begun (resp. finished) processing a product.
   (d) M_2 (resp. M_2'): A token in M_2 (resp. M_2') means that M_2 has begun (resp. finished) processing a product.
   (e) M_1_F (resp. M_2_F): A token in M_1_F (resp. M_2_F) means that M_1 (resp. M_2) is free.
2. Transitions:
   (a) P_A (resp. P'_A): A raw material is loaded in buffer $\mathtt{buf_a}$ (resp. $\mathtt{buf_b}$).
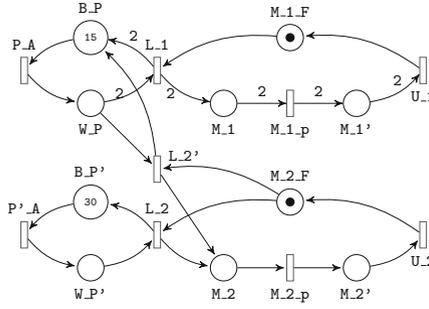
Fig. 2. SRN model of configuration C_1.

(b) L_1: Machine M_1 loads a raw material from buffer buf_a.

(c) L_2 (resp. L_2'): Machine M_2 loads a raw material from buffer buf_b (resp. buf_a), where L_2 $>^0$ L_2' and L_2' $>^1$ L_2.

(d) U_1 (resp. U_2): Machine M_1 (resp. M_2) unloads a finished product.

(e) M_1_p (resp. M_2_p): Machine M_1 (resp. M_2) processes a product.

When the number of waiting raw material in buffer buf_a exceeds ten (i.e., $\mu(\text{W\_P}) \geq 10$), the RMS switches to configuration C_1. This switching is modeled by rule $r_0 = \langle C_0, C_1, \mathbb{M}^0, \varphi_0 \rangle$, where:

1. C_0 and C_1 are the source and target configurations of r_0.
2. $\mathbb{M}^0(\text{W\_P}, \text{M\_1\_F}) = (10, 1)$.
3. φ_0 is the exponentially-distributed applying time of rule r_0.

Once the number of waiting raw materials in buf_a is less than five, the RMS switches to configuration C_0. This switching is modeled by rule $r_1 = \langle C_1, C_0, \mathbb{M}^1, \varphi_1 \rangle$, where:

1. C_1 and C_0 are the source and target configurations of r_1.
2. $\mathbb{M}^1(\text{P\_A}, \text{M\_1\_F}) = (11, 1)$.
3. φ_1 is the exponentially-distributed applying time of rule r_1.

Now, we consider the analysis aspect. To verify the system properties, first we create equivalent SRN G of CD-SRN $N = \langle \mathcal{P}, \mathcal{T}, \mathcal{C}, (\mathcal{R}, \succ), \gamma_0 \rangle$ described above, where:

1. $\mathcal{P} = \{\text{B\_P}, \text{B\_P'}, \text{W\_P}, \text{W\_P'}, \text{M\_1}, \text{M\_1'}, \text{M\_1\_F}, \text{M\_2}, \text{M\_2'}, \text{M\_2\_F}\}$.
2. $\mathcal{T} = \{\text{P\_A}, \text{P'\_A}, \text{L\_1}, \text{M\_1\_p}, \text{U\_1}, \text{L\_2}, \text{L\_2'}, \text{U\_2}, \text{M\_2\_p}\}$.
3. $\mathcal{C} = \{C_0, C_1\}$.
4. $\mathcal{R} = \{r_0, r_1\}$ is the set of rules given above.
5. $r_0 \succ r_1$.
6. γ_0 is an initial configuration C_0 shown in Fig. 1.

$G = \langle P, T, D^-, D^+, D^o, g, >, \eta_0, \lambda, w, \mathcal{M} \rangle$, the equivalent net of N, is computed using the algorithm proposed above.

To obtain $P = \mathcal{P} \cup \mathring{P}$, we create $\mathring{P} = \{\mathring{p}_0, \mathring{p}_1\}$ where $\mathring{p}_0$ and $\mathring{p}_1$ are associated with C_0 and C_1, respectively. The initial marking of places in $\mathring{P}$ is $\eta_0(\mathring{p}_0, \mathring{p}_1) = (1, 0)$.

Considering the unfolding of transitions, we give examples about L_1, L_2 and L_2'.

Transition L_1 does not change its priority, therefore L_1 has a unique representation in G which is $\dot{\text{L}}\_1$ ($T \longleftarrow T \cup \{\dot{\text{L}}\_1\}$) where

- $D^-_{\text{W\_P},\dot{\text{L}}\_1}(\mu\_\mathring{\eta}) = 1$ if $\eta(\mathring{p}_0) = 1$,
- $D^-_{\text{W\_P},\dot{\text{L}}\_1}(\mu\_\mathring{\eta}) = 2$ if $\eta(\mathring{p}_1) = 1$,
- $D^-_{\text{M\_1\_F},\dot{\text{L}}\_1}(\mu\_\mathring{\eta}) = 1$,
- $D^+_{\text{B\_P},\dot{\text{L}}\_1}(\mu\_\mathring{\eta}) = 1$ if $\eta(\mathring{p}_0) = 1$,
- $D^+_{\text{B\_P},\dot{\text{L}}\_1}(\mu\_\mathring{\eta}) = 2$ if $\eta(\mathring{p}_1) = 1$,
- $D^+_{\text{M\_1},\dot{\text{L}}\_1}(\mu\_\mathring{\eta}) = 1$ if $\eta(\mathring{p}_0) = 1$,
- $D^+_{\text{M\_1},\dot{\text{L}}\_1}(\mu\_\mathring{\eta}) = 2$ if $\eta(\mathring{p}_1) = 1$,
- and $\mathcal{I}(\text{L\_1}, C_0) = \mathcal{I}(\text{L\_1}, C_1) = \dot{\text{L}}\_1$.

As for transitions L_2 and L_2', they change their priorities over configurations C_0 and C_1, hence L_2 (resp. L_2') has two representations $\dot{\text{L}}\_2_0$ and $\dot{\text{L}}\_2_1$ (resp. $\dot{\text{L}}\_2\text{'}_0$ and $\dot{\text{L}}\_2\text{'}_1$), where:

- $D^-_{\text{W\_P'},\dot{\text{L}}\_2_0}(\mu\_\mathring{\eta}) = 1$,
- $D^-_{\text{W\_P},\dot{\text{L}}\_2\text{'}_0}(\mu\_\mathring{\eta}) = 1$,
- $D^-_{\text{M\_2\_F},\dot{\text{L}}\_2_0}(\mu\_\mathring{\eta}) = 1$,
- $D^-_{\text{M\_2\_F},\dot{\text{L}}\_2\text{'}_0}(\mu\_\mathring{\eta}) = 1$,
- $D^+_{\text{B\_P'},\dot{\text{L}}\_2_0}(\mu\_\mathring{\eta}) = 1$,
- $D^+_{\text{B\_P},\dot{\text{L}}\_2_1}(\mu\_\mathring{\eta}) = 1$,
- $D^+_{\text{M\_2},\dot{\text{L}}\_2_0}(\mu\_\mathring{\eta}) = 1$,
- $D^+_{\text{M\_2},\dot{\text{L}}\_2_1}(\mu\_\mathring{\eta}) = 1$,
- $\mathcal{I}(\text{L\_2}, C_0) = \dot{\text{L}}\_2_0$, $\mathcal{I}(\text{L\_2}, C_1) = \dot{\text{L}}\_2_1$, $\mathcal{I}(\text{L\_2'}, C_0) = \dot{\text{L}}\_2\text{'}_0$ and $\mathcal{I}(\text{L\_2'}, C_1) = \dot{\text{L}}\_2\text{'}_1$,
- $g_{\dot{\text{L}}\_2_0}(\mu\_\mathring{\eta}) = (g^0_{\text{L\_2}}(\mu) \wedge \eta(\mathring{p}_0) = 1)$,
- $g_{\dot{\text{L}}\_2_1}(\mu\_\mathring{\eta}) = (g^1_{\text{L\_2}}(\mu) \wedge \eta(\mathring{p}_1) = 1)$,
- $g_{\dot{\text{L}}\_2\text{'}_0}(\mu\_\mathring{\eta}) = (g^0_{\text{L\_2'}}(\mu) \wedge \eta(\mathring{p}_0) = 1)$,
- $g_{\dot{\text{L}}\_2\text{'}_1}(\mu\_\mathring{\eta}) = (g^1_{\text{L\_2'}}(\mu) \wedge \eta(\mathring{p}_1) = 1)$,
- $\dot{\text{L}}\_2_0 > \dot{\text{L}}\_2\text{'}_0$ and $\dot{\text{L}}\_2\text{'}_1 > \dot{\text{L}}\_2_1$.

Finally, we add two transitions $\dot{r}_0$ and $\dot{r}_1$ to model the switching from C_0 to C_1 and from C_1 to C_0, respectively. $\dot{r}_0$ which models the switching to C_1 is given as follows.

- $D^-_{\mathring{p}_0,\dot{r}_0} = 1$,
- $D^+_{\mathring{p}_1,\dot{r}_0} = 1$,
- $g_{\dot{r}_0}(\eta) = 1$ if $\eta(\text{W\_P}) \geq 10 \wedge \eta(\text{M\_1\_F}) = 1$,

- $\lambda_{\dot{r}_0}(\mu_-\mathring{\eta}) = \varphi_0(\mu)$,
- for each transition $\dot{t} \in T$ in the equivalent net that is an image of a transition in $\mathcal{T}$, let $\dot{r}_0 > \dot{t}$,
- $\dot{r}_0 > \dot{r}_1$,
- and $d_{\dot{r}_0}(\mu_-\mathring{\eta}) = d_0(\mu)$.

Obtained equivalent SRN G is illustrated in Fig. 3.
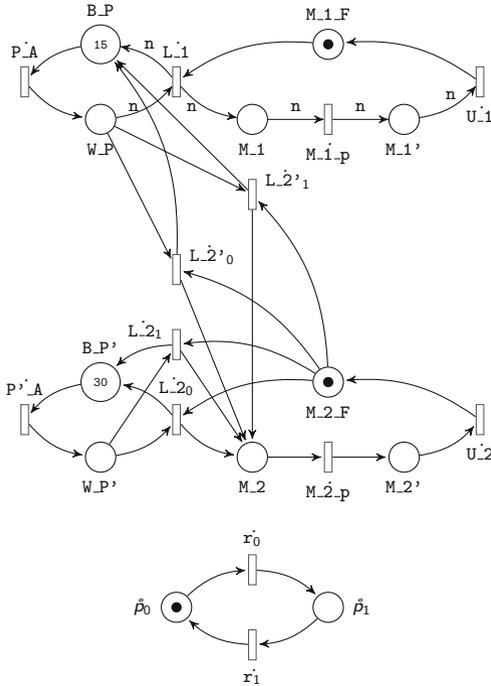
Fig. 3. Equivalent SRN G, where $n = \#(\mathring{p}_0) + 2\#(\mathring{p}_1)$, and $\#(p)$ stands for the marking of place p.

7   Conclusion

SRNs have found a wide application in the design of discrete event systems. However, their use for the modeling and analyzing reconfigurable systems is not convenient, since they do not support dynamic structures. In this paper, we have proposed an extension of SRNs, called configuration-dependent SRN, that allows the modeling and analysis of such systems.

The proposed approach is inspired from R-SPNs. In future work, we focus on developing an approach that supports adding and/or removing places and/or transitions in a given CD-SRN.

References

1. Bashir, M.: A minimal supervisory structure to optimally enforce liveness on Petri net models for flexible manufacturing systems. IEEE Access **5**, 15731–15749 (2017)
2. Bashir, M., Liu, D., Uzam, M., Wu, N., Al-Ahmari, A., Li, Z.: Optimal enforcement of liveness to flexible manufacturing systems modeled with Petri nets via transition-based controllers. Adv. Mech. Eng. **10**(1) (2018). https://doi.org/10.1177/1687814017750707
3. Brettel, M., Friederichsen, N., Keller, M., Rosenberg, M.: How virtualization, decentralization and network building change the manufacturing landscape: an Industry 4.0 perspective. Int. J. Mech. Ind. Sci. Eng. **8**(1), 37–44 (2014)
4. Cao, Y., Zhao, L., Zhang, R., Yang, Y., Zhou, X., Li, K.: Experience-availability analysis of online cloud services using stochastic models. In: 2018 IFIP Networking Conference (IFIP Networking) and Workshops, pp. 1–9 (2018)
5. Chen, Y., Li, Z., Al-Ahmari, A., Wu, N., Qu, T.: Deadlock recovery for flexible manufacturing systems modeled with Petri nets. Inf. Sci. **381**, 290–303 (2017)
6. Chryssolouris, G., Efthymiou, K., Papakostas, N., Mourtzis, D., Pagoropoulos, A.: Flexibility and complexity: is it a trade-off? Int. J. Prod. Res. **51**(23–24), 6788–6802 (2013)
7. Ciardo, G., Blakemore, A., Chimento, P.F., Muppala, J.K., Trivedi, K.S.: Automated generation and analysis of Markov reward models using stochastic reward nets. In: Linear Algebra, Markov Chains, and Queueing Models, pp. 145–191. Springer, New York (1993)
8. Cong, X.: Design of optimal Petri net supervisors for flexible manufacturing systems via weighted inhibitor arcs. Asian J. Control **20**(1), 511–530 (2018)
9. Fu, Y., Zhu, J., Gao, S.: CPS information security risk evaluation system based on Petri net. In: Proceedings of IEEE Second International Conference on Data Science in Cyberspace, pp. 541–548 (June 2017)
10. Guo, Z., Zhang, Y., Zhao, X., Song, X.: A timed colored Petri net simulation-based self-adaptive collaboration method for production-logistics systems. Appl. Sci. **7**(3) (2017)
11. Guțuleac, E., Zaporojan, S., Moraru, V., Sclifos, A.: Performance modeling of network defense in breadth systems by matrix rewriting SRN with fuzzy parameters. J. Eng. Sci. **3**, 38–53 (2019). https://doi.org/10.5281/zenodo.3444051
12. Henning, K.: Recommendations for implementing the strategic initiative INDUSTRIE 4.0 (2013)
13. Huang, J., Zhu, Y., Cheng, B., Lin, C., Chen, J.: A Petri net-based approach for supporting traceability in cyber-physical manufacturing systems. Sensors **16**(3) (2016)
14. Jackson, K., Efthymiou, K., Borton, J.: Digital manufacturing and flexible assembly technologies for reconfigurable aerospace production systems. Procedia CIRP **52**, 274–279 (2016). The Sixth International Conference on Changeable, Agile, Reconfigurable and Virtual Production (CARV2016)
15. Kahloul, L., Bourekkache, S., Djouani, K.: Designing reconfigurable manufacturing systems using reconfigurable object Petri nets. Int. J. Comput. Integr. Manuf. **29**(8), 889–906 (2016)
16. Li, J., Dai, X., Meng, Z.: Automatic reconfiguration of Petri net controllers for reconfigurable manufacturing systems with an improved net rewriting system-based approach. IEEE Trans. Autom. Sci. Eng. **6**(1), 156–167 (2009)

17. Liu, M., He, Z., Wu, N., Al-Ahmari, A., Li, Z.: Resource configuration analysis for a class of Petri nets based on strongly connected characteristic resource subnets. IEEE Access **5**, 26376–26386 (2017)
18. Llorens, M., Oliver, J.: Introducing structural dynamic changes in Petri nets: marked-controlled reconfigurable nets. In: Proceedings of International Symposium on Automated Technology for Verification and Analysis, pp. 310–323. Springer, Heidelberg (2004)
19. Long, F., Zeiler, P., Bertsche, B.: Potentials of coloured Petri nets for realistic availability modelling of production systems in Industry 4.0. In: Proceedings of the ESREL 2015 Conference, vol. 7 (2015)
20. Long, F., Zeiler, P., Bertsche, B.: Modelling the flexibility of production systems in Industry 4.0 for analysing their productivity and availability with high-level Petri nets. IFAC-PapersOnLine **50**(1), 5680–5687 (2017). 20th IFAC World Congress
21. Lu, Y.: Industry 4.0: a survey on technologies, applications and open research issues. J. Ind. Inf. Integr. **6**, 1–10 (2017)
22. Lv, Y., Lin, D.: Design an intelligent real-time operation planning system in distributed manufacturing network. Ind. Manage. Data Syst. **117**(4), 742–753 (2017)
23. Ma, Z., Li, Z., Giua, A.: Characterization of admissible marking sets in Petri nets with conflicts and synchronizations. IEEE Trans. Autom. Control **62**(3), 1329–1341 (2017)
24. Nguyen, T.A., Min, D., Choi, E., Tran, T.D.: Reliability and availability evaluation for cloud data center networks using hierarchical models. IEEE Access **7**, 9273–9313 (2019)
25. Nguyen, T.A., Han, K., Min, D., Choi, E., Thang, T.D., Choi, Y.J.: A stochastic reward net-based assessment of reliability, availability and operational cost for a software-defined network infrastructure. J. Supercomput. **75**(8), 4657–4683 (2019)
26. Shojaee, R., Latifi, A., Yazdani, N.: A stochastic reward net approach to model availability of cloud virtualization. In: 7th International Symposium on Telecommunications, IST2014, pp. 683–688 (2014)
27. da Silva, R.M., Junqueira, F., Filho, D.J.S., Miyagi, P.E.: Control architecture and design method of reconfigurable manufacturing systems. Control Eng. Pract. **49**, 87–100 (2016)
28. Tigane, S., Kahloul, L., Bourekkache, S.: Generalized stochastic Petri nets with rewritable topology. In: Proceedings of First International Conference on Embedded Distributed Systems (EDiS), pp. 1–6 (December 2017)
29. Tigane, S., Kahloul, L., Bourekkache, S.: Reconfigurable stochastic Petri nets: a new formalism for reconfigurable discrete event systems. In: Proceedings of International Conference on Mathematics and Information Technology (ICMIT), p. 301–308 (2017)
30. Tigane, S., Kahloul, L., Baarir, S., Bourekkache, S.: Dynamic GSPNs: formal definition, transformation towards GSPNs and formal verification. In: Proceedings of the 13th EAI International Conference on Performance Evaluation Methodologies and Tools, VALUETOOLS 2020, pp. 164–171. ACM, New York, NY, USA (2020)
31. Tigane, S., Kahloul, L., Benharzallah, S., Baarir, S., Bourekkache, S.: Reconfigurable GSPNs: a modeling formalism of evolvable discrete-event systems. Sci. Comput. Program. **183**, 102302 (2019). https://doi.org/10.1016/j.scico.2019.102302
32. Tigane, S., Kahloul, L., Bourekkache, S., Baarir, S.: Extending GSPNs for the modelling, analysis and performance evaluation of dynamic systems. Int. J. Crit. Comput. Based Syst. **8**(1), 25–44 (2018)

Author Index