Jordanian Journal of Computers and Information Technology

April 2017 | VOLUME 03 | NUMBER 01 | ISSN 2415 - 1076 (Online) ISSN 2413 - 9351 (Print)

JJCIT

www.jjcit.org jjcit@psut.edu.jo

An International Peer-Reviewed Scientific Journal
Financed by the Scientific Research Support Fund

Jordanian Journal of Computers and Information Technology (JJCIT)

The Jordanian Journal of Computers and Information Technology (JJCIT) is an international journal that publishes original, high-quality and cutting edge research papers on all aspects and technologies in ICT fields.
JJCIT is hosted by Princess Sumaya University for Technology (PSUT) and supported by the Scientific Research Support Fund in Jordan. Researchers have the right to read, print, distribute, search, download, copy or link to the full text of articles. JJCIT permits reproduction as long as the source is acknowledged.

AIMS AND SCOPE

The JJCIT aims to publish the most current developments in the form of original articles and review articles in all areas of Telecommunications, Computer Engineering and Information Technology and make them available to researchers worldwide.
The JJCIT focuses on topics including, but not limited to: Computer Engineering & Communication Networks, Computer Science & Information Systems and Information Technology and Applications.

INDEXING

JJCIT is indexed in:

- CrossRef:
  http://search.crossref.org/?q=jjcit
- OCLC WorldCat:
  http://www.worldcat.org/search?qt=worldcat_org_all&q=jjcit
- Scilit:
  http://www.scilit.net/journals/387088
- DRJI (Directory of Research Journals Indexing):
  http://drji.org/JournalProfile.aspx?jid=2415-1076

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 3, No. 1, April 2017.

A WEIGHTED SCORE MATCHING ALGORITHM FOR A MULTI-MODAL BIOMETRIC SYSTEM BASED ON FINGERPRINT AND HAND GEOMETRY

Ezdihar N. Bifari[1] and Lamiaa A. Elrefaei[2]

(Received: 30-Aug.-2016, Revised: 04-Nov.-2016, Accepted: 05-Dec.-2016)

ABSTRACT

This paper presents a multi-modal biometric system implemented using MATLAB language. The system fused fingerprint and hand geometry at matching score level by applying a proposed modified weighted sum rule. The fingerprint system was tested using five FVC databases and the hand geometry system was tested using COEP database. Hence, the multi-modal system was tested by merging each FVC database with COEP database. The experimental results show significant improvement in the multi-modal system with an average EER of 3.27%; while it is 8.86% and 8.89% for fingerprint and hand geometry systems, respectively.

KEYWORDS

Multi-biometrics, Fusion, EER, ROC, Minutiae, Gabor filter, FFT filter, Morphological, Chain code.

1. INTRODUCTION

People used a variety of ways trying to obtain safer and more precise methods to identify themselves and protect their personal properties and private information; like using keys, ID card, password or PIN code. However, these traditional security methods do not satisfy the requirements of most companies and governmental agencies at the present time, especially with modern applications that use electronic services operating automatically, like ATMs in banks, e-learning, e-commerce, criminal search tools … and so on. Hence, these applications need effective and stronger protection methods. Biometric systems are becoming popular as a measure to identify a person by what the characteristics of the person are rather than what the person carries. Each biometric system has its advantages and disadvantages, but still offers great convenience and several advantages over traditional security systems [1]-[2]. The next sub-section presents a brief summary on biometric systems.

1.1 Single Biometric Systems

A biometric system is defined as "a system which automatically distinguishes and recognizes a person as a unique individual through a combination of hardware and pattern recognition algorithms based on certain physiological or behavioral traits (characteristics) that are inherent to that person" [3]. According to this definition, the biometric system depends on the person's specific (physiological or behavioral) traits. Therefore, system data cannot be shared, lost, stolen or forgotten [1], [4]. Any human physiological or behavioral trait can be used as a biometric trait, but it has to satisfy some basic requirement factors; shown in Table 1 [1].

Biometric systems are generally classified into verification and identification systems. Verification systems confirm or deny the identity of the person using his/her ID code.

1. E. Bifari is with the Computer Science Department, King Abdulaziz University, Jeddah, Saudi Arabia. E-mail: ezdihar_b@hotmail.com.
2. L. Elrefaei is with the Computer Science Department, King Abdulaziz University, Jeddah, Saudi Arabia, and with the Electrical Engineering Department, Benha University, Cairo, Egypt. E-mail: laelrefaei@kau.edu.sa, lamia.alrefaei@feng.bu.edu.eg.

Therefore, the output decision query is only between one person and only one user template, which is stored in the database (one to one matching process). On the other hand, identification systems are used to determine the identity of one person with respect to all user templates in the database (one to many matching process) [1]. Any biometric system should include four main steps:

Table 1. Comparison of various biometric traits usage basic requirements [5].
Usage basic requirements; High: H, Medium: M, Low: L.

	Universality	**Distinctiveness**	**Permanence**	**Collectability**	**Performance**	**Acceptability**	**Circumvention**
Face	H	L	M	H	L	H	H
Fingerprint	M	H	H	M	H	M	M
Hand Geometry	M	M	M	H	M	M	M
Iris	H	H	H	M	H	L	L
Voice	M	L	L	M	L	H	H
Signature	L	L	L	H	L	H	H

1. Pre-processing: Improvement of sample quality, elimination of the noise resulting from sensor device type or environment noise and making the sample appropriate for the next step.
2. Feature extraction: Extraction of details of biometric trait that distinguishes between persons.
3. Feature matching: In this step, the similarity score will be estimated to decide whether the extracted features from two biometric traits are matched or not.
4. Decision making: The final step is used to decide whether the sample is accepted or rejected by selecting an appropriate threshold value and comparing it with the similarity score.

Figure 1 shows the block diagram of how biometric (verification and identification) systems work with the four main steps mentioned above.

Biometric systems offer good recognition performance, but are not free of errors. However, even the most advanced biometric systems are still facing numerous problems; including data, algorithm used and system design, which cause a negative impact on the performance.

The main drawbacks of biometric systems can be summarized in the following points [3]:

- Noisy data because of a noisy environment or a problem during the use of sensor device which makes the feature extraction process more difficult.
- Not all biometric characteristics are universal because of various birth defects and accidents that lead to the absence of the biometric trait in some people; like a cut hand or finger.
- Lack of individuality, where the biometric trait is similar among a group of people, so that the system cannot distinguish between them, as in the case of twins.
- Susceptibility to circumvention as when an impostor presents a fake biometric sample to the system, such as using gummy fingers.

To resolve the previous problems, upgraded hardware could be used, employing liveliness detection techniques, using robust algorithms to improve the quality of the sample and matching process, or using multi-biometric systems and integrating information from different sources [4]. The next sub-section is oriented toward giving a brief idea on the latter solution; multi-

biometric systems, which is the focus of our study.

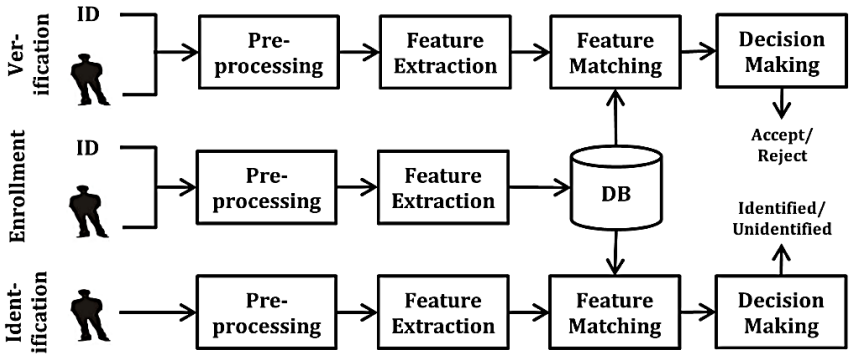

Figure 1. Biometric (verification and identification) systems.

1.2 Multi-biometric Systems

Multi-biometric systems can offer significant improvement in the performance accuracy of a biometric system, population coverage, preventing spoofed attacks and reducing the failure rate of enrollment data to the system [6]. The term multi-biometric indicates "the presence and use of more than one biometric aspect (modality, algorithm, instance and/or sensor) in some form of combined use for making a specific biometric verification/identification decision" [7]. According to the previous definition, multi-biometric systems are classified into four main categories as shown in Figure 2.

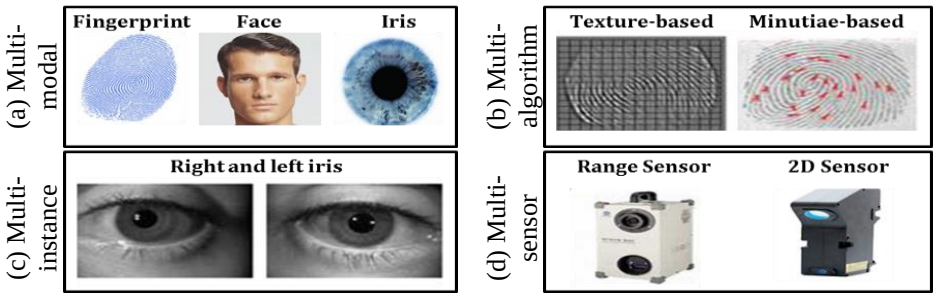

Figure 2. Multi-biometric system categories.

Even though fusing all these categories may offer very high performance results, this procedure was avoided due to increased time consumption and cost effect. Thus, our interest was focused on the multi-modal biometric system which is one of the widely applicable systems.

Different modality of single biometric systems can fuse their information in different levels and produce a multi-modal biometric system. The common fusion levels are [8]:

1. Feature level: Merging feature sets extracted from single systems into one feature set.
2. Score level: In the feature matching step, the similarity scores of each single system are normalized and fused using rules like max and sum rules.
3. Decision level: The simplest method, because it combines the final decisions of each system using rules like AND, OR and Majority voting.

With regard to existing biometric systems used for recognition, there is still room for improvement through development of devices, techniques or algorithms. Therefore, in this

paper, a simple and effective multi-modal biometric system based on fingerprint and hand geometry was introduced. We applied the fingerprint system mentioned in our previous paper [9], where the steps were explained here in more detail. In addition, we adopted the proposed weighted feature matching algorithm in [9] for the proposed hand geometry system and multi-modal biometric system as explained in Section 3.

The remainder of the paper is organized as follows: Section 2 presents the related works. In Section 3, the proposed multi-modal biometric system based on fingerprint and hand geometry at score level fusion is described. Section 4 shows the experimental results of the system with statistics, graphs and tables for illustration. Finally, Section 5 concludes the paper and provides some suggestions for future investigation.

2. RELATED WORKS

2.1 Overview of Fingerprint System

Fingerprint system is one of the most common and reliable systems used for security purposes, because of its uniqueness, universality, invariability and extraction facilities [10]. Figure 3 displays an example of fingerprint surface pattern, which consists of ridge and valley lines. The local ridge lines have a lot of information and form a set of distinct features called minutiae, which is classified into two popular types: ridge ending and ridge bifurcation. These minutiae are used in the biometric system to distinguish between people [9], [11]-[12].

Figure 3. Example of fingerprint structure.

Many algorithms have been suggested for enhancement in the specialized literature for fingerprint recognition, either to improve fingerprint image quality, extract the features or match feature approaches [10], [13]. Hong *et al.* [11] published the major article that used Gabor filter to improve fingerprint image quality. They evaluated the goodness index method of minutiae extraction in the verification system. Popovic *et al.* [14] produced directional log-Gabor filter and applied it using low quality images on frequency domain. Chikkerur *et al.* [15] implemented a new algorithm based on short time Fourier transform analysis (STFT).

For feature extraction methods, most studies extracted ridge ending and ridge bifurcation using Crossing Number (CN) concepts [16]-[17]. This method depends on the eight-neighbourhood of each ridge pixel [16]. Moreover, Babatunde *et al.* [17] modified the crossing number method to give the same result with lower cost. Xin *et al.* [18] extracted the minutiae from grey-scale images by using Gabor phase field method. Tico and Kuosmane [19] presented a method that validates and deletes many false minutiae, such as spikes, holes, bridges, ladders and spurs. With regard to matching algorithms, Wei *et al.* [20] implemented an algorithm using local ridge line sets to deal with image distortion and false features. They proposed an equation to calculate final matching score for merging two matching algorithms; a graph matching with global orientation field matching. Bengueddoudj *et al.* [21] used local and global structures of the fingerprint to modify the existent minutiae matching algorithm. Jie *et al.* [22] compensated the lack of fingerprint information by improving the effect of orientation field, minutiae, reconnected ridges and recovered missing minutiae.

2.2 Overview of Hand Geometry System

Hand geometry becomes popular as a biometric trait and is applied in various identification systems of real-world applications. Each human has special hand geometry features that can be useful in distinguishing between persons; such as different shape of the hand, size of the palm, length and width of the fingers [23]-[24]. Hand geometry is considered for low-medium security, but with advantages of low computational cost algorithm, low resolution images, high user acceptance and low storage size [2], [25]. Some hand geometry sensor devices came with pegs to fix the position of the user's hand. Others do not rely on these devices as they are captured using a camera in different positions making them more challenging, especially in the field of research, investigation and crime scenes [26]. Many research studies about hand geometry systems have been conducted using different numbers of features or different algorithms.

Saxena *et al.* [2] proposed a new threshold algorithm for hand segmentation. It depends on 21 features from palm width, finger length and width. The system was tested using 6 different distance functions and compared between 96 users with 5 images for each user. Kang and Wu [27] presented a segmentation method to extract fingers from hand images using the principle of a modified Otsu. Al-Ani and Abd Rajab [26] implemented a method for feature extraction using a two-dimensional discrete cosine transform. Their research depended only on extracting the length and width of fingers. In addition, they evaluated the system performance by calculating matching metric correlation. Furthermore, Fierrez *et al.* [28] implemented a system with 17 features extracted from fingers and palm. The features were grouped into 8 sets; each set consisted of a combination of several features. Xiong *et al.* [29] studied the problem of applying hand in the biometric system without using any pegs to determine hand position. They produced an elliptical model to get optimal alignments of hands and improved the traditional geometry measurement methods to extract about 37 features from fingers. Yildirim and Tulay [30] introduced a new approach to identify the hand by its pattern. They classified and recognized patterns using general regression neural networks.

2.3 Overview of Multi-modal Biometric System

There are many studies in multi-modal biometric systems field. These studies vary in terms of the biometric characteristics selected, the level of the fusion or the algorithm used. Jain and Ross [31] developed a multi-modal system using fingerprint, face and hand geometry. They compared between user-specific threshold and their proposed user-specific weight methods which were used for fusion of the biometric traits. Charfi *et al.* [32] suggested fusion between hand shape and palmprint in verification system. They extracted the features by improving Scale Invariant Feature Transform algorithm. The system was tested on IITD hand database and fused at score level using max, min, product and sum rules. Abdolahi *et al.* [33] implemented a novel fusion method to combine iris and fingerprint at decision level. This method depended on Hamming distance, fuzzy logic and weighted code.

Dehache and Souici-Meslati [34] proposed a multi-modal verification system using two traits: fingerprint and signature. They classified the single modes by neural multi-layer perception and fused their matching scores using the support vector machine approach. Meghanathan *et al.* [35] proposed an integration weight optimization technique to improve the performance of biometric models. This technique reduced the computational complexity and used a small number of training samples. They applied their technique using fingerprint and voice. The system used sum rule to fuse the scores and was tested using FVC2002 database for fingerprint and ELSDSR database for voice. Mohi-ud-Din *et al.* [36] presented a multi-modal system based on palmprint and fingerprint using two levels of fusion; feature level and score level. Directional energy-based feature vectors were used to fuse feature vector for each module in one vector, while the sum and product rules were applied in score level fusion.

"A Weighted Score Matching Algorithm for a Multi-modal Biometric System Based on Fingerprint and Hand Geometry", Ezdihar N. Bifari and Lamiaa A. Elrefaei.

3. The Architecture of The Proposed Multi-modal Biometric System

The proposed multi-modal system was implemented at score level fusion based on two models; fingerprint and hand geometry, as shown in Figure 4. The fingerprint was suggested for its strength and uniqueness, while hand geometry was suggested for its speed. The human body has various biometric traits; our focus was on hand, because different traits can be obtained from it using one sensor device, such as fingerprints, palm prints, vein patterns and hand geometry [23]. Hence, there will be no extra inconvenience to the user when using the system, while the accuracy of the system may be increased due to the addition of several features [37].

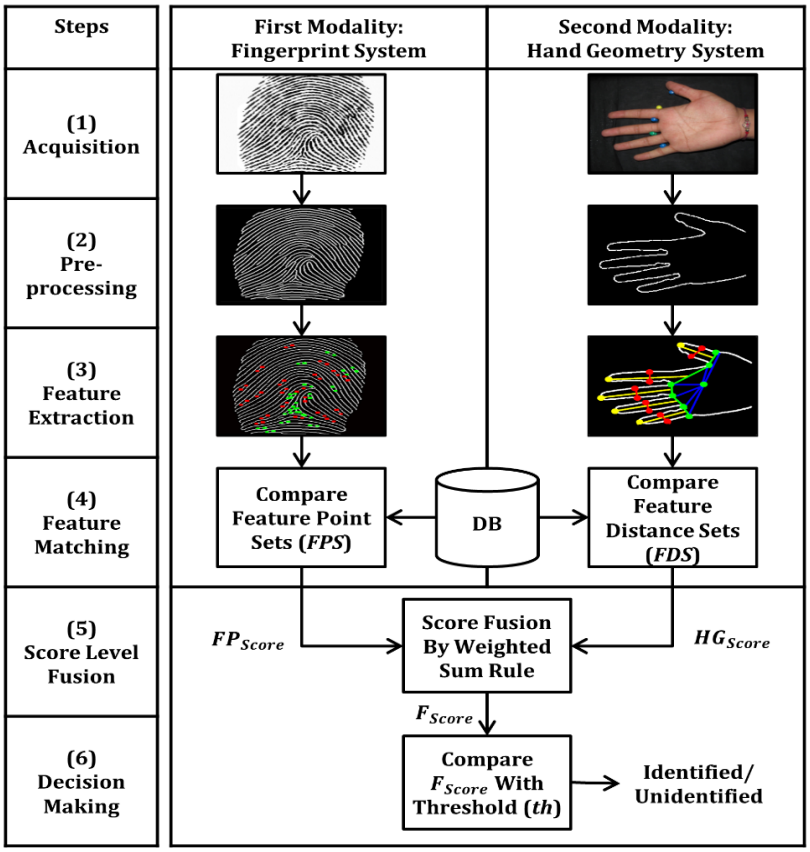

Figure 4. Overview of the proposed multi-modal biometric system.

3.1 The First Modality: Fingerprint System

Although very effective solutions are currently available, there is still need for improvement. In this work, the fingerprint system in our previous paper [9] is implemented; where the weighted feature matching algorithm was modified for more accurate results. Hence, the feature point weight table has been updated. The detailed steps of the proposed fingerprint system [9] is explained in the following subsections.

3.1.1 Pre-processing

When you take fingerprint images by a sensor device, this may result in several image qualities due to the noise in the environment. Therefore, the goal of pre-processing is improving fingerprint image quality and facilitating feature extraction [38]. The basic idea to enhance the image is taken from the classical Gabor filter technique [11]. It was used for contextual enhancement using local characteristics information on fingerprint; i.e., ridge orientation and

ridge frequency to correct ridge endings and bifurcation [39]-[40]. Figure 5 displays the pre-processing steps in the proposed fingerprint system.

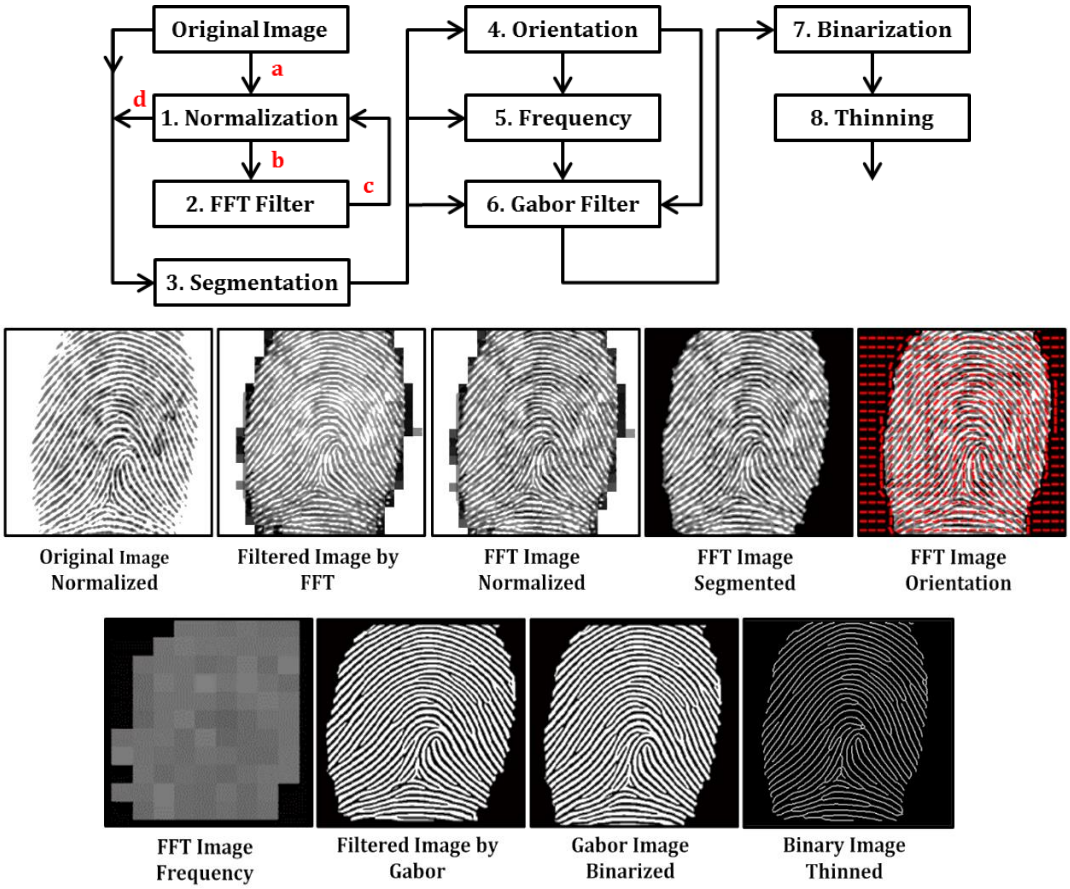

Figure 5. Pre-processing steps in fingerprint system with an example.

We propose to change the classic technique by applying an adaptive histogram equalization algorithm in the normalization step and adding a new step to filter the image using Fast Fourier Transform (FFT) filter. This filter will connect the broken ridges and fill the holes [39]. Image enhanced by FFT filter is used as an input for Gabor filter technique instead of using the normalized image, as it is more accurate and clear. In this way, we combine the advantages of the two filters to get an image with better quality. In addition, images are segmented in a good way. Figure 6 displays some results to clarify the difference between classic and proposed filters.

In our system, the segmentation process will create two masks; the first one (I_{mask1}) will be used to isolate the foreground region from the background region. The second mask (I_{mask2}) which is smaller than I_{mask1} will play a role in the feature extraction process. Furthermore, in the last step, two thinning images were created using morphological thinning operation for the feature extraction process:

1. Using binary image (I_{Binary}) to create a thinned image ($I_{Thinned1}$), which is used to extract ridge ending points.
2. Inverted binary image ($\sim I_{Binary}$) was applied to create a second thinned image ($I_{Thinned2}$), which is used to locate the positions of the ridge bifurcation points.

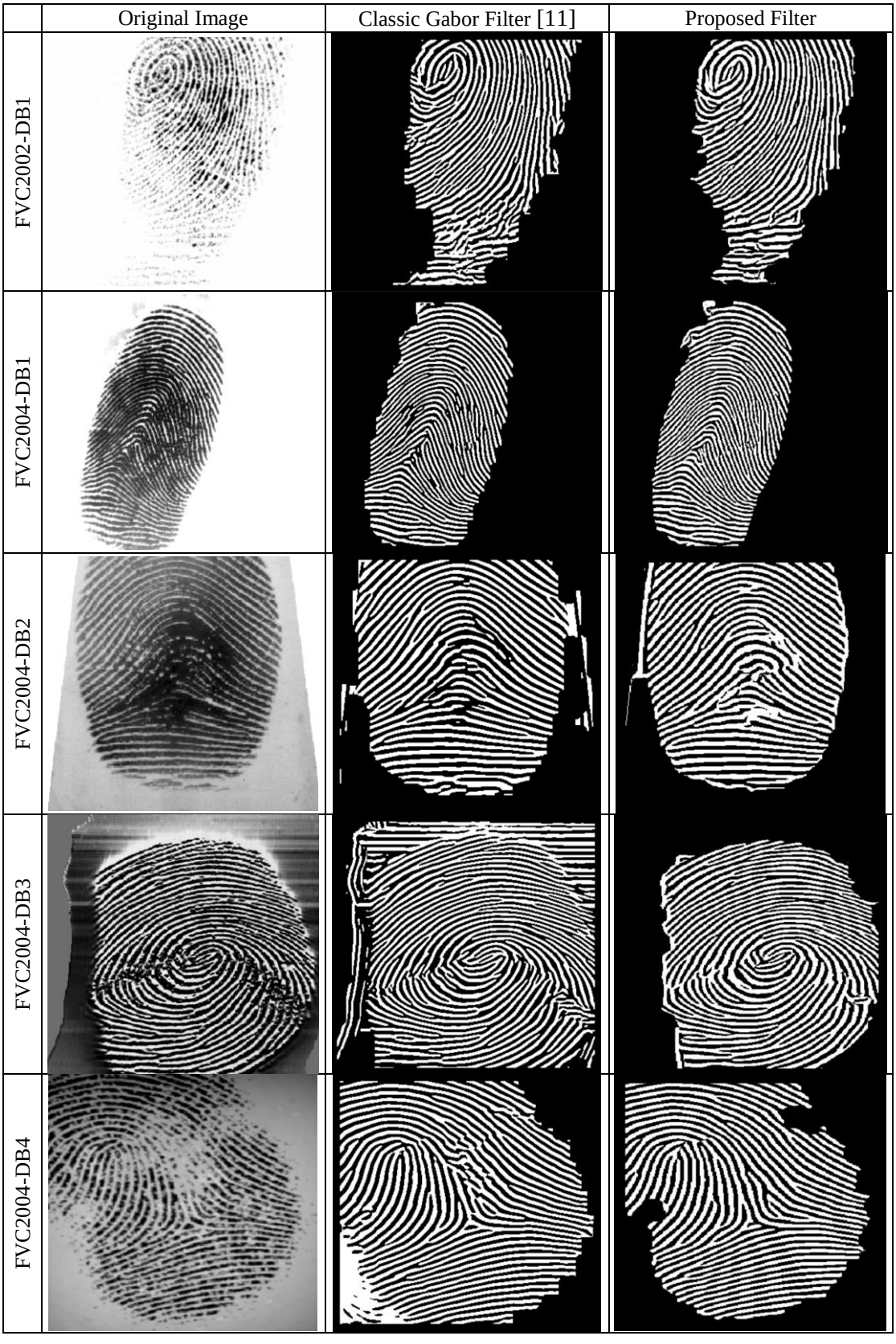

Figure 6. Some results from pre-processing steps in the fingerprint system.

3.1.2 Feature Extraction

After the fingerprint image is enhanced, fingerprint features represented in minutiae and ridge points can be extracted easily [20]. The combination between these points was suggested to give more information about the fingerprint that our proposed matching algorithm can use, especially if the image is corrupted or distorted. The proposed feature extraction method is applied to extract only one type of minutiae that is a ridge ending with its ridge point. Therefore, the method will be executed twice: first using ($I_{Thinned1}$) to extract a ridge ending and then using ($I_{Thinned2}$) to extract also a ridge ending, but its coordinates correspond to ridge bifurcation coordinates in ($I_{Thinned1}$).

1. Exclusion of Ridge Pixels on the Border: Ridge pixels on the border can cause a large number of spurious minutiae of ridge ending type. To remove these pixels, the second mask (I_{mask2}) produced in the segmentation step will be applied to eliminate every pixel that exists outside the mask. To do that, each ridge pixel (P) in the thinned image will be scanned by checking its coordinates with the mask as follows:

$$P(x,y) = \begin{cases} Keep\ it, & I_{mask2}(x,y) = 1. \\ Delete, & otherwise. \end{cases} \quad (1)$$

2. Ridge Ending Extraction: Ridge ending will be extracted by scanning the eight-neighbourhood pixels around P, then by calculating the summation (N) of these eight pixels. The pixel P is considered to be a ridge ending (M) if $N = 1$. Figure 7 illustrates the notion derived from crossing number method [16].

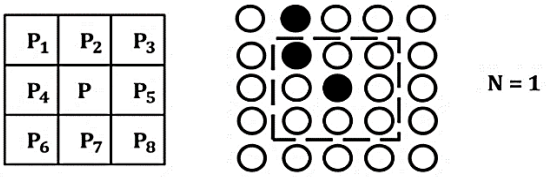

Figure 7. Extraction of ridge ending point.

3. Ridge Point Extraction: An extra feature was extracted to get more accurate information about minutiae and ridge line shape, defined as ridge point (R). Figure 8 illustrates the concept [20] with a sample.
4. In our system, extracting the ridge point R according to minutiae type is derived from the algorithm mentioned in [19], which is also used to validate the minutiae. This algorithm depends on the ridge shape, which means that if there is any deformation in the ridge shape, the minutiae will be ignored and deleted. Therefore, our focus was only on ridge ending point M, since it has only one branch which reduces the presence of ridge distortions.

The proposed ridge point extraction method is summarized as follows:

a. Create an initial image (L) with size ($W \times W$) and fill it with pixels from $I_{Thinned}$ which is centered by M and located in its ($W \times W$) neighbourhood.

b. Label each connected region in L.

c. Count the number (C) of similar pixel values on the border, which is equal to the value of the center pixel M.

d. In case (C=1), M is a true ridge ending and the pixel on the border is the ridge point (R). Otherwise, delete it as shown in Figure 9.

(a) The concept [20]. (b) An example of a fingerprint feature.

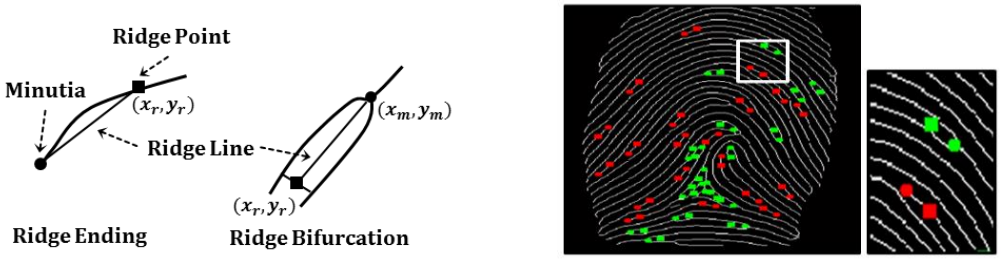

Figure 8. Illustration of minutiae and ridge points.
Ridge ending (red), bifurcation (green), minutiae point (●) and ridge point (■).

True Feature: C = 1 False Feature: C = 2

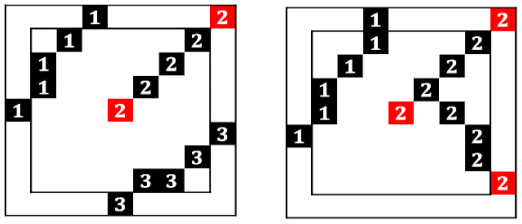

Figure 9. An example of ridge point extraction.

All true feature points extracted will be saved in a set called feature point set (*FPS*) in the following arrangement:

$$FPS = \begin{Bmatrix} \left(M_x^1, M_y^1, M_\theta^1, R_x^1, R_y^1, R_\theta^1\right), \\ \left(M_x^2, M_y^2, M_\theta^2, R_x^2, R_y^2, R_\theta^2\right), \\ \dots\ \dots\ \dots \\ \left(M_x^n, M_y^n, M_\theta^n, R_x^n, R_y^n, R_\theta^n\right) \end{Bmatrix}. \quad (2)$$

Here, *n* is the number of minutiae in the set, *M* represents minutiae point, *R* is ridge point and (*x*, *y*, θ) are the point components; the coordinates and the orientation, respectively.

Figure 10 displays an example of feature set *FPS* for ($I_{Thinned1}$) which consists of minutiae (ending and bifurcation) and their ridge points. *FPS* is a union between FPS_1 in ($I_{Thinned1}$) and FPS_2 in ($I_{Thinned2}$) of the same fingerprint.

3.1.3 Feature Matching

This step is used to determine whether two biometric traits are matched by comparing their extracted features and calculating the similarity score. Minutiae-based matching algorithms [21]-[22], [41]-[42] are commonly used in fingerprint systems. They basically depend on extracted minutiae information; the minutia coordinates (*x*, *y*) and the ridge orientation direction θ.

In our system, a weighted feature matching algorithm was proposed and derived from the classic minutiae-based matching algorithm. It has a simple changing with two features: minutiae and ridge points. These will be stored in a feature point set (*FPS*) as stated in the previous subsection. This suggested *FPS* allows the combination between the two classical matching algorithms; minutiae-based and ridge-based algorithms. In this way, more information about extracted minutiae can be gathered in the partial fingerprint. The steps of the proposed matching

algorithm are as follows.

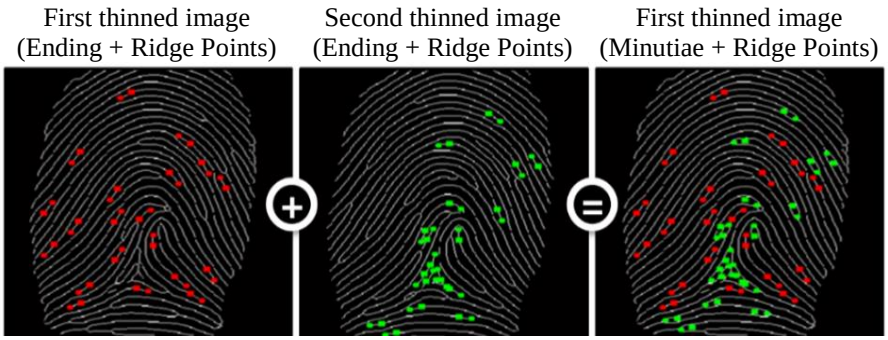

Figure 10. An example of feature extraction in the fingerprint system.

1. Registration: Registration is a process applied on two (*FPS*); template feature point set (FPS^T) and input feature point set (FPS^I). The goal is to align and register them with respect to each other using an affine transform [43]. The steps to estimate the appropriate transformation parameters for accepted registration are as follows:
   a. Make a list of reference points (*RP*) in the origin using each minutia in *FPS*.
   b. Transform the original coordinate system of each point in (*FP*) whether it is minutia point *M* or ridge point *R* in *FPS* to a new one and update its field values with:

$$\begin{pmatrix} new\_FP_x \\ new\_FP_y \\ new\_FP_\theta \end{pmatrix} = \begin{pmatrix} cos(RP_\theta) & -sin(RP_\theta) & 0 \\ sin(RP_\theta) & cos(RP_\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} FP_x - RP_x \\ FP_y - RP_y \\ FP_\theta - RP_\theta \end{pmatrix}. \quad (3)$$

   c. Finally, pair the features to calculate the maximum number of matching pairs.

   Since the system does not know whether the two fingerprints are belonging to the same person, it performs the registration process using all *RP* in order to find best transformation. Figure 11 shows a registration of two different samples for the same fingerprint.

2. Feature Pairing: After the registration process, two features are considered to be paired or matched if their point coordinates (x, y) and orientation (θ) are close enough to each other [43]-[44]. In our proposed algorithm, we just check the match of point coordinates without the orientation, since the direction minutiae will be specified by the ridge point. Moreover, the possibility of the existence of matching pair candidates is eliminated by using two tolerance boxes; one for each feature point. Figure 12 illustrates the idea.

Any pairing feature inside the tolerance box is supposed to take the similarity value 1, while in case of un-matched feature, it will be 0. However, the matching pairs in our system were classified according to their component level and took a similarity value in the range [0, 1]. To find the feature similarity value, we do the following:

a. Compute the Euclidean distance ($Dist_1$) between minutiae pairs (M^T, M^I) and ($Dist_1$) between ridge point pairs (R^T, R^I) using the general equation between two points (p, q):

$$Dist(p,q) = \sqrt{(p_x - q_x)^2 + (p_y - q_y)^2}\,. \quad (4)$$

b. Calculate the difference angles between (M^T, M^I) to get (Ang_1) and between (R^T, R^I) to get (Ang_2) using the general equation:

$$Ang(p,q) = |p_\theta - q_\theta|. \quad (5)$$

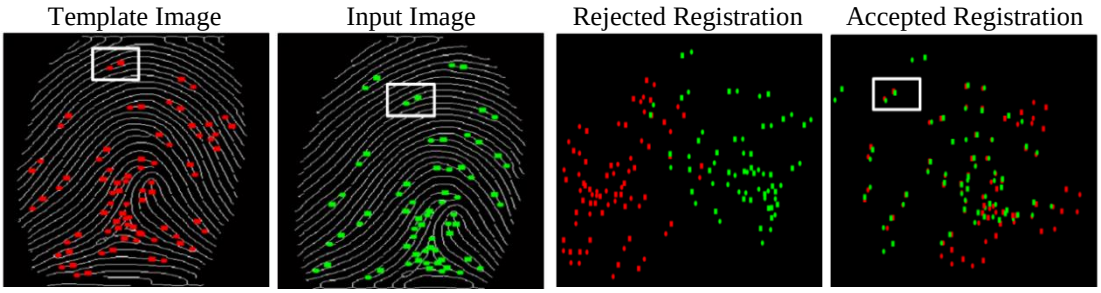

Figure 11. Registration of two samples for the same fingerprint.

c. Associate a level value for each *Dist* and *Ang* according to a threshold:

$$Level_{Dist} = \begin{cases} 2, & Dist \leq Th_1. \\ 1, & otherwise. \end{cases} \quad (6)$$

$$Level_{Ang} = \begin{cases} 2, & Ang \leq Th_2. \\ 1, & otherwise. \end{cases} \quad (7)$$

d. Calculate a feature pair similarity using Table 2. The table shows that the components ($Dist_1$, Ang_1, $Dist_2$ *and* Ang_2) have different weights (*CW*) from high (4) to low (1). The algorithm depends on the minutiae points; therefore, it will have the higher weight. Furthermore, the coordinates will be with higher weight, because the orientation value depends on the location of the point. The steps to create the table are as follows:

i. Find feature pair priority (*Pr*) through multiplying each component level by its weight, then sum the results.

$$Pr = \sum_{i=1}^{4} (CL_i \times CW_i). \quad (8)$$

ii. Arrange *Pr* in ascending manner to get Feature Weight (*FW*).

iii. Obtain feature pair similarity (*FS*) by normalizing *FW* to be in the range [0, 1].

$$FS_i = \frac{FW_i}{16}, \qquad i = 1, 2, \dots, 16. \quad (9)$$

iv. In case multiple points exist inside tolerance boxes, higher *FS* will be chosen. In addition, if multiple points exist with same highest *FS*, maximum *FW* will be taken with minimum minutiae pair distance ($Dist_1$).

In this way, we can decide whether the matching feature has been affected during the registration process or not. Therefore, a weight and different similarity value were given to the matching feature according to its quality: high value for high quality and low value for low quality.

Finally, we compute the matching similarity score of the two fingerprints (FP_{Score}) within the range [0, 1] by the next traditional equation.

$$FP_{Score} = \left(\sum_{i=1}^{m} FS_i\right) \Big/ min(n^T, n^I). \quad (10)$$

Here, *m* is the number of matching features, while n^T and n^I are the numbers of features in the template and input fingerprints, respectively.

3.2 The Second Modality: Hand Geometry System

In this subsection, the proposed hand geometry system will be described in detail with all methods and algorithms that were applied. The proposed system could work on any image, whether it contains pegs or not, but the hand image must be separate fingers as shown in Figure 4. The system passes through the following steps:

3.2.1 Pre-processing

The goal of pre-processing is to clean up any noise from the image and to make it appropriate and easy for the next step of extracting features as shown in Figure 13.

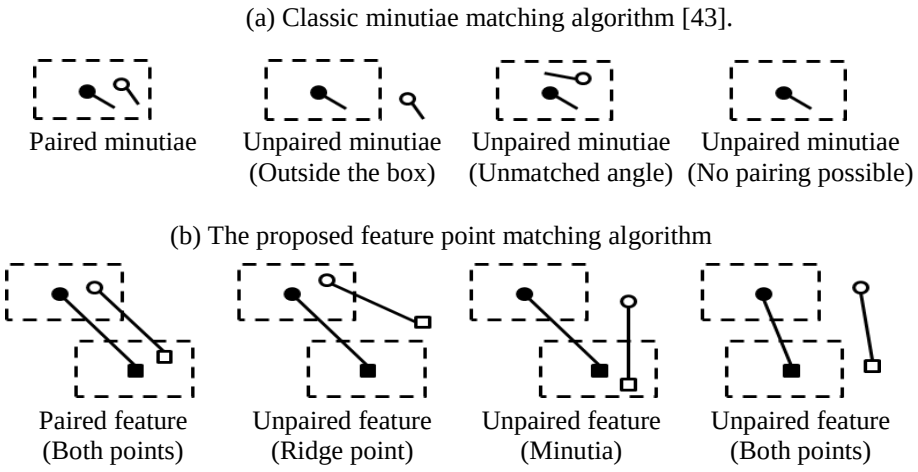

Figure 12. Possibilities of feature matching in the fingerprint system.
Minutiae point (●) and ridge point (■).

1. Convert to Grayscale Image: In our system, color images were used containing color pins to fix the suitable position of the hand. These pins are distorting hand borders and must be discarded. As is known, the color images are composed of three channels: red, green and blue. An arithmetic operation is performed with these different channels to transform the color image into a grayscale image. In our case, the green channel (I_{Green}) of the image is subtracted from the red one (I_{Red}) using equation **Error! Reference source not found.**. In this way, the pins are removed from the image given a clear grayscale image I_{Gray}) of the hand.

2. Binarization: This process is used to generate the binary image (I_{Binary}), which means making the image with only two colors; black and white. To do that, first an appropriate global threshold value (th_1) must be determined by applying Otsu method on the grayscale image. Then, each pixel in the grayscale image is compared with (th_1) as shown in the next equation.

$$I_{Binary}(x,y) = \begin{cases} 0, & I_{Gray}(x,y) < th_1. \\ 1, & otherwise. \end{cases} \quad (11)$$

3. Segmentation: As shown in Figure 13, the hand image shows a part of the arm. Hence, it was necessary to apply segmentation process to remove the arm part regardless of its length. The main idea is to determine the centroid point (C) of the hand. In literature [27], the binary image is used to locate C. In our system, we decided to thin the binary image and get the hand structure to do that. In this way, the extracted centroid point is more accurate. The following steps will be implemented:
   a. Removing noise from the binary image by applying 2-D median filter. This filter is used to replace each pixel (P) in I_{Binary} by the median of all pixel values inside a window with

size ($W \times W$). The window is a copy of I_{Binary}, so that its center pixel will be corresponding to P.

Table 2. Feature point weight table in the fingerprint system.
(Two feature levels with four point components; 24 = 16 possibilities)

	Minutia		**Ridge Point**		**Pr**	**FW**	**FS**
	$Dist_1$	**Ang_1**	**$Dist_2$**	**Ang_2**			
CW	**4**	**3**	**2**	**1**			
CL	1	1	1	1	10	1	0.063
	1	1	1	2	11	2	0.125
	1	1	2	1	12	3	0.188
	1	1	2	2	13	4	0.250
	1	2	1	1	13	5	0.313
	1	2	1	2	14	6	0.375
	1	2	2	1	15	8	0.500
	1	2	2	2	16	10	0.625
	2	1	1	1	14	7	0.438
	2	1	1	2	15	9	0.563
	2	1	2	1	16	11	0.688
	2	1	2	2	17	12	0.750
	2	2	1	1	17	13	0.813
	2	2	1	2	18	14	0.875
	2	2	2	1	19	15	0.938
	2	2	2	2	20	16	1.000

CW: Component weight, CL: Component level,
Pr: Feature pair priority, FW: Feature Weight, FS: Feature pair similarity.

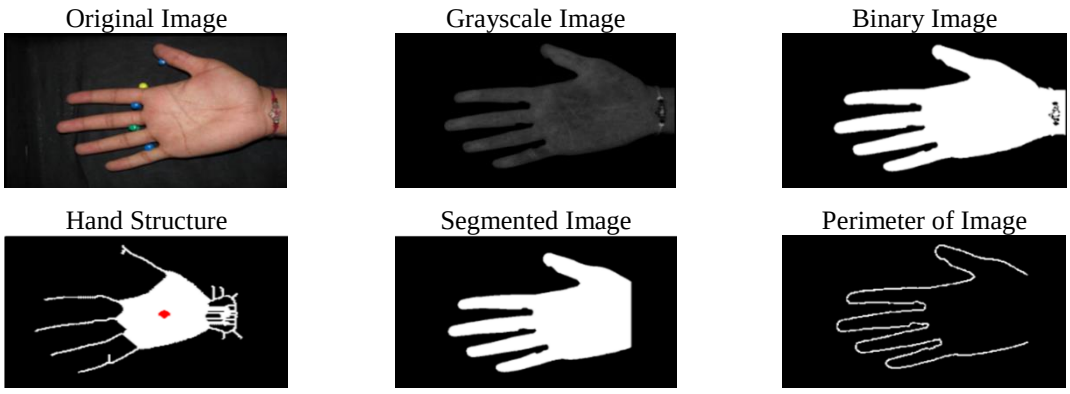

Figure 13. An example of pre-processing
steps in the hand geometry system.

b. Performing morphological operations for:
   i. Removing all small objects from the binary image using erosion operation.
   ii. Thinning the binary image to get hand structure ($I_{Thinned}$) using thin operation.

c. Determining the centroid point (C) of the thinned image:
   i. Count the total number of white pixels (n) in $I_{Thinned}$.
   ii. Find the (x, y) coordinates of all white pixels.
   iii. Locate the centroid point using:

$$C = I_{Thinned}(mean_x, mean_y). \quad (12)$$

$$mean_x = \frac{1}{n}\sum_{i=0}^{n} x_i. \quad (13)$$

$$mean_y = \frac{1}{n}\sum_{i=0}^{n} y_i. \quad (14)$$

d. Determining the hand length (*L*) according to hand acquisition; in our case, it will be in the x-direction. Therefore, a suitable threshold value (th_2) will be added to the *x* value of the centroid point (*C*).

$$L = C_x + th_2. \quad (15)$$

e. Finally, Remove arm part and estimate the segmented image ($I_{Segment}$) from the binary image.

$$I_{Segment}(x,y) = \begin{cases} I_{Binary}(x,y), & x < L. \\ 0, & otherwise. \end{cases} \quad (16)$$

4. Hand Perimeter: A morphological operation was applied on the segmented image to create the perimeter image with hand boundary that is exactly one pixel thick [45]. The method works by executing erode operation on $I_{Segment}$, then subtracting the resulting image from $I_{Segment}$ to get the perimeter image ($I_{Perimeter}$).

3.2.2 Feature Extraction

From previous studies, we found that the number of extracted features differs from a system to another. In general, those features are often represented in distances of hand width, finger width and finger length at different points along the finger [37]. The features in hand geometry depend on the landmark points of the hand, which are denoted by peaks and valleys between adjacent fingers. As well, additional points could be extracted from basic landmark points, represented in the finger middle points.

In our system, chain code algorithm [46] was applied to extract basic hand points from the perimeter image ($I_{Perimeter}$) as Figure 14 displays. Then, the extracted hand points were used to estimate five distance features of hand width (blue lines, a) and five distance features of finger length (yellow lines, b), while using ten distance features of finger width (red lines, c) and (green lines, d). The basic equations that we need to extract the distance features are as follows:

a. The Euclidean distance (*Dist*) between (*p*, *q*) by equation (4).

b. The center point (*C*) coordinates of a line exist between (*p*, *q*):

$$C_x = (p_x + q_x) / 2. \quad (17)$$

$$C_y = (p_y + q_y) / 2. \quad (18)$$

All twenty distance features extracted will be saved in a set known as the feature distance set (*FDS*):

$$FDS = \begin{Bmatrix} a_1, a_2, a_3, a_4, a_5, \\ b_1, b_2, b_3, b_4, b_5, \\ c_1, c_2, c_3, c_4, c_5, \\ d_1, d_2, d_3, d_4, d_5 \end{Bmatrix} = \{Dist_1, Dist_2, \dots, Dist_{20}\}. \quad (19)$$

3.2.3 Feature Matching

Feature matching determines the matching score of similarity between two hands; template

feature distance set (FDS^T) and input feature distance set (FDS^I). In a hand geometry system, there is no need to align the hands, since the algorithm directly compares between distances. The classic method [28] for feature matching is based on a distance measure. This is done through computing the summation of absolute difference (*diff_d*) between each distance feature in FDS^T and its corresponding feature in FDS^I:

$$diff\_d = \sum_{i=1}^{m} \left| Dist_i^T - Dist_i^I \right|. \tag{20}$$

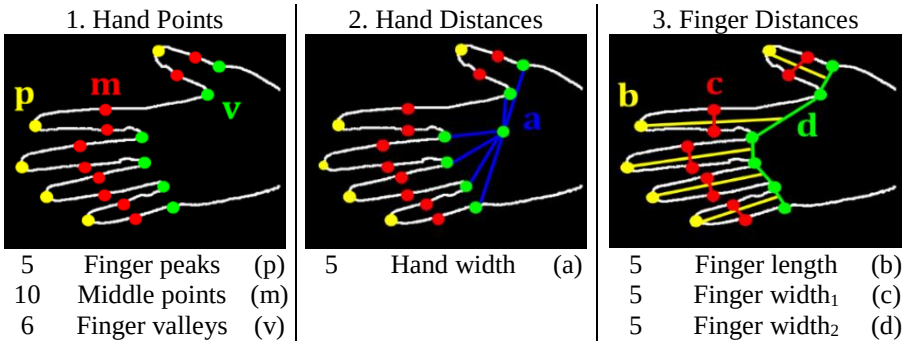

Figure 14. Feature extraction in hand geometry system.

Here, *m* is the number of features in the *FDS*.

The perfect feature match can be obtained if the value of *diff_d* is equal to zero. *diff_d* increases whenever it is heading towards mismatches. Therefore, a certain threshold value is determined to decide the outcome of matching. In our system, a simple change in the classic method [47] was proposed to get better results. The steps of the proposed feature matching are:

1. Classifying the features and taking the mean of the absolute difference for each of them:
   - Hand distance (a):

$$diff\_d_1 = \frac{1}{6} \sum_{i=1}^{6} |a_i^T - a_i^I|. \tag{21}$$

   - Finger distances (b, c and d):

$$diff\_d_{i+1} = \frac{1}{3} \left| \left(b_i^T + c_i^T + d_i^T\right) - \left(b_i^I + c_i^I + d_i^I\right) \right|, \qquad i = 1, 2 \ldots 5. \tag{22}$$

2. Determining an appropriate limit for feature pairing by applying two thresholds; th_1 and th_2, which are also used to give a weight for the feature pair called the feature level.

$$Level_{diff\_d} = \begin{cases} 2, & diff\_d \le Th_1. \\ 1, & Th_1 < diff\_d \le Th_2. \\ 0, & otherwise. \end{cases} \tag{23}$$

3. Computing the matching similarity score of the two hands geometry (HG_{Score}) in the range [0, 1] by obtaining distance feature pair similarity (*DS*) using the next two equations.

$$DS = \frac{Level_{diff\_d}}{2}. \tag{24}$$

$$HG_{Score} = \frac{1}{m} \sum_{i=1}^{m} DS_i. \tag{25}$$

Here, *m* is the total number of *DS* of the hand and fingers, which is equal to 6.
The goal of feature classification is to control the selected threshold values for each feature, which makes the classification more accurate. The mean of the features was taken in order to minimize the effect of any error which can occur when calculating distances.

3.3 Proposed Multi-modal Biometric System

Data fusion in multi-biometric systems is commonly applied to improve recognition performance. In our work, score level fusion was proposed to implement the proposed multi-biometric system based on two models; fingerprint and hand geometry, as shown in Figure 4.

3.3.1 Score Level Fusion

To calculate the final score of the proposed multi-modal system, the score resulting from each single mode will be fused using rules like minimum (min), maximum (max), sum, … etc. [48]. The weighted sum rule was the adopted rule for fusion in our system. The expression of weighted sum rule to compute matching similarity score (F_{Score}) of the multi-modal system is:

$$F_{Score} = \frac{1}{m}\sum_{i=1}^{n} w_i.Score_i = \frac{1}{m}(w_1.FP_{Score} + w_2.HG_{Score}). \quad (26)$$

$$F_{Score} = \frac{1}{m}(w_1.FP_{Score} + w_2.HG_{Score}). \quad (27)$$

Here, *m* is the value used to normalize the score within the range [0, 1], *n* is the number of single systems which will be fused, *w* and *Score* are the weight and matching score of each single system, respectively.

The value of *m* depends on the weight given to each module. In the traditional sum rule, the weight of all models will be equal, thus *m* will be equal to *n*. On the other hand, to estimate the weights in weighted sum rule, the classic method is used by obtaining the EER for each module and giving small weight to higher EER [31]. This method uses exhaustive search for all samples in the database and depends on the quality of stored samples. This leads to instability in the values assigned for each weight.

We suggested to assign weights according to the degree of distinctiveness factor of the biometric trait as illustrated in Table 1. According to the table, the degree of the fingerprint is high and hand geometry is medium. Thus, we proposed to assign a weight value of w_1=2 for fingerprint and w_2=1 for hand geometry; which led to assign *m* with 3 in order to normalize the score between [0, 1]. In this way, we overcame the system dependability on the EER.

3.3.2 Decision Making

Decision making is the final step in the proposed system, which is used to make the final decision. It checks whether the input and template samples belong to the same person by comparing the similarity score estimated from fusion module using a suitable threshold value (*Th*) as:

$$Decision = \begin{cases} Identified, & F_{Score} \geq Th. \\ Unidentified, & otherwise. \end{cases} \quad (28)$$

4. EXPERIMENTAL RESULTS

4.1 Performance Evaluation

To build a robust biometric system, the effectiveness of the system and its level of error rates

must be measured. Receiver Operating Characteristic (ROC) curve is a common approach to evaluate system performance and calculate error rates [49]. ROC can be used to calculate Error Equal Rate (EER) which describes the point at which genuine and imposter error rates are equal. It shows the correlation between: False Match Rate (FMR) which is known as False Acceptance Rate (FAR) and False Non Match Rate (FNMR) also called False Rejection Rate (FRR) [50].

Our proposed system was designed and programmed using MATLAB language. The experiments were performed using several databases to measure the system performance. Each database contains 800 samples; 100 different persons and 8 samples for each one. Therefore, according to the database, the total number of impostor tests will be: (100×99×0.5) = 4950, while the total number of genuine tests performed is: (8×7×0.5) × 100 = 2800 [50]. The EER was calculated through finding FMR and FNMR on different thresholds. In addition, the time cost of two stages was calculated; enrollment stage which consist of pre-processing and feature extraction and matching stage. It takes the average of (4950+2800) = 7750 comparisons.

4.2 Results of Fingerprint System

The system was tested using five public databases known as Fingerprint Verification Competition (FVC) [50]-[51]. The results of the proposed fingerprint identification system of the five databases are presented in Table 3. The minimum EER is 2.83% at Th of 0.23 for FVC2002-DB1. The maximum EER is 11.76% at Th of 0.22 for FVC2004-DB2. This is due to the deformation of some images in this database. The performance of the proposed system provides reliable results compared with previous studies with an average EER of 8.86% and Th of 0.22. Table 4 presents time cost of the proposed system. The minimum time cost the of enrollment step is 2.82 sec for FVC2004-DB1, while that of the matching step is 0.79 sec for FVC2002-DB1.

Table 3. Comparison between previous studies and the proposed fingerprint identification system on FVC databases.

EER%	Study [18]	Study [44]	Study [22]	Study [21]	Study [9]	Proposed	
						EER	Th
FVC2002-DB1	4.86	N/A	11.26	N/A	2.88	2.83	0.23
FVC2004-DB1	19.49	12.0	N/A	22.81	11.31	11.16	0.20
FVC2004-DB2	20.76	8.2	9.29	18.54	12.57	11.76	0.22
FVC2004-DB3	13.95	5.0	N/A	9.0	9.23	8.93	0.21
FVC2004-DB4	14.48	7.0	N/A	17.72	9.97	9.64	0.22
Average	14.71	8.05	10.28	17.02	9.19	8.86	0.22

Table 4. Time cost of the proposed fingerprint identification system on FVC databases.

Time in seconds	FVC2002-DB1	FVC2004-DB1	FVC2004-DB2	FVC2004-DB3	FVC2004-DB4	Average
Enrollment	4.60	2.82	4.51	5.70	3.96	4.32
Matching	0.79	2.58	3.62	13.23	1.08	4.26

4.3 Results of Hand Geometry System

The COEP palm print database was adopted to test the proposed hand geometry system. The database is a subsidiary of the College of Engineering, Pune-411005 (Autonomous Institute of Government of Maharashtra). It consists of a total of 1344 images pertaining to 168 persons with 8 different images of a single person. The images were captured using a digital camera with a quality resolution of 1600×1200 pixels [52].

In our system, only 100 persons were selected from the hand geometry database, since the fingerprint database contains 100 persons. We renamed the hand geometry files to correspond to the fingerprint images. In this way, we can claim that the fingerprint and hand geometry belong

to the same person and we can use them in the proposed multi-modal system. The results of the proposed hand geometry system are displayed in Table 5 and compared against those of previous studies. We note that the results are mixed, because there are many variables in each study, including the size of the database, the number of samples, the algorithm used … and so on. In general, the proposed system gives good and reliable results with EER of 8.89% at Th of 0.33. The time cost of the proposed system in the enrollment stage is 3.31 sec; while it takes about 2.5693e-004 sec in the matching stage.

Table 5. Comparison between previous studies and the proposed hand geometry identification system.

EER: FMR=FNMR	**Study [28]**	**Study [29]**	**Study [53]**	**Study [30]**	**Proposed**
Database Size	50	108	120	140	100
Samples	10	5	N/A	N/A	8
FMR%	1.24	2.4	11.4	15.0	8.89 at Th = 0.33
FNMR%			10.4		

4.4 Results of Multi-modal Biometric System

The proposed multi-modal biometric system depends on the single system databases. Since the fingerprint system used five databases and the hand geometry system used only one, the multi-modal system was tested five times by repeating the hand geometry database for each fingerprint database as shown in Table 6.

Table 6. Experimental results of multi-modal biometric system based on fingerprint and hand geometry at score level fusion using different rules.

EER%	**Min**	**Max**	**Sum Rule**	**Proposed Weighted Sum Rule**	
				EER	**Th**
FVC2002-DB1 with Coep	5.94	3.13	2.85	2.02	0.32
FVC2004-DB1 with Coep	9.25	5.15	5.23	3.74	0.25
FVC2004-DB2 with Coep	9.11	5.31	5.18	3.95	0.26
FVC2004-DB3 with Coep	7.49	5.29	5.14	3.59	0.25
FVC2004-DB4 with Coep	8.77	4.69	4.19	3.06	0.28
The Average	8.11	4.71	4.52	3.27	0.27

Table 7. Comparison between previous studies and the proposed multi-modal system.

EER: FMR=FNMR	**Study [32]**	**Study [33]**	**Study [34]**	**Study [54]**	**Proposed**
Traits used	Hand geometry Palmprint	Fingerprint Iris	Fingerprint Signature	Face Speech Signature	Fingerprint Hand geometry
Database Size	230	N/A	110	30	100
Samples	5	N/A	N/A	24	8
FMR%	1.8	2	3.80	~ 5%	2.02 ~ 3.95 = 3.27
FNMR%			3.55		

In order to illustrate the effect of the suggested rule on the EER results of the multi-modal system, different score level fusion rules were applied. Table 6 shows that the results were worst when the system used min rule with average EER of 8.11%, while the system gives best results when using the proposed weighted sum rule. It achieves an average EER of 3.27%.

Furthermore, Table 7 displays some results from previous studies to compare with the proposed multi-modal system. We found that the results are close and the EER values are low in all studies, despite the different characteristics used in each study and the different sizes of the

databases used. The minimum EER is 1.8% in [32] and the maximum EER is 5% in [54]. This indicates that the multimodal biometric system generally gives good results.

Finally, Figure 15 presents the three proposed biometric systems; fingerprint, hand geometry and multi-modal systems. The results illustrate that using the multi-modal system will improve the overall accuracy of the system and give more reliable results compared to the single systems. The experimental results showed that the average EER of the multi-modal system was 3.27%, while the fingerprint system and hand geometry system achieved an average EER of 8.86% and 8.89%, respectively.

5. CONCLUSION

In this work, we present a multi-modal biometric system framework that utilizes two biometric models; fingerprint and hand geometry model. These models are fused at the score level using weighted sum rule. Several algorithms and techniques were applied at each step in the system to improve recognition rates. The main contribution in the biometric field was through the implementation of the proposed matching algorithm which is based on the weighted feature for all three systems: fingerprint, hand geometry and multi-modal systems.

In addition to the matching algorithm, we enhanced the fingerprint by modifying Gabor filter algorithm and extracted extra points called ridge points beside the minutiae, in order to increase the information from each sample. In the hand geometry system, twenty distance features were extracted by chain code algorithm which were reduced to six features by classifying them into hand features and finger features. Finally, the multi-modal biometric system was experimentally tested using FVC and COEP databases that are not previously fused together according to the literature review. We tested the system using different traditional fusion rules, like min, max and sum rules, in addition to the proposed weighted sum rule to see which rule offers the best results.

In our experiments, it was observed that the use of the proposed matching algorithm with other proposed techniques resulted in improving the performance of each system. Especially, fingerprint module used five public FVC databases which have been created for international competition and contain a large number of low quality fingerprint samples. On the other hand, the COEP database of hand geometry included acceptable samples. Nevertheless, the EER in the hand geometry system is considered high compared to that in the fingerprint system.

Furthermore, the use of multiple biometrics and fusing them at the score level led to increase the performance compared to separate individual biometric systems. The experimental results showed that the average EER of the multi-modal system between fingerprint and hand geometry was 3.27%, while the fingerprint system achieved an average EER of 8.86% for all five FVC databases and the hand geometry system achieved an EER of 8.89% for COEP database. These results showed a significant improvement in the multi-modal biometric system by nearly 5.61%.

For future investigation, researchers can try to enhance the performance of multi-modal biometric systems by introducing the idea of matching algorithm based on weighted features with other enhanced techniques. Also, they can apply it including other biometric traits to create a multi-modal system with more than two models.

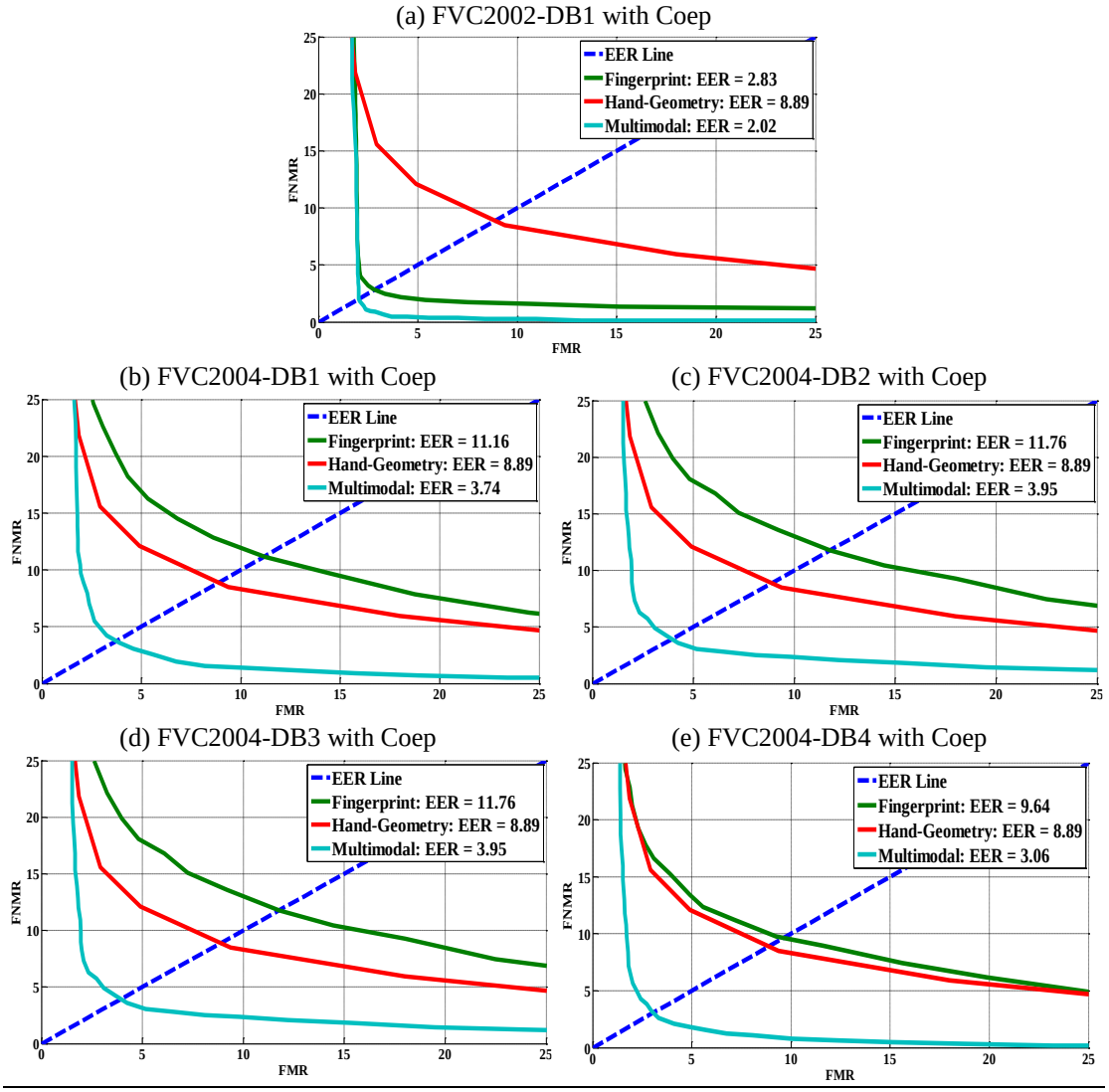

Figure 15. ROC curves of the comparison between the proposed systems: fingerprint, hand geometry and multi-modal systems.

REFERENCES

[1] S. Ram, Fingerprint Ridge Orientation Modeling, PhD Thesis, Technology Institute for Computer Graphics and Vision, Graz University, 2009.

[2] N. Saxena, V. Saxena, N. Dubey and P. Mishra, "HAND GEOMETRY: A New Method for Biometric Recognition," International Journal of Soft Computing and Engineering (IJSCE), vol. 2, pp. 192-196, January 2013.

[3] M. M. Monwar, A Multi-modal Biometric System Based on Rank Level Fusion, PhD Thesis, Department of Computer Science, University of Calgary, Calgary, Alberta, 2012.

[4] P. Kartik, S. Prasanna and R. Vara Prasad, "Multimodal Biometric Person Authentication System Using Speech and Signature Features," in TENCON-IEEE Region 10 Conference, pp. 1-6, 2008.

[5] U. Uludag, S. Pankanti, S. Prabhakar and A. K. Jain, "Biometric Cryptosystems: Issues and Challenges," Proceedings of IEEE, vol. 92, issue 6, pp. 948-960, June 2004.

[6] R. N. Rodrigues, L. L. Ling and V. Govindaraju, "Robustness of Multi-modal Biometric Fusion Methods against Spoof Attacks," Journal of Visual Languages & Computing, vol. 20, pp. 169-

179, 2009.

[7] A. Mishra, "Multi-modal Biometrics: Need for Future Systems," International Journal of Computer Applications, vol. 3, pp. 28-33, June 2010.

[8] S. Sheena and S. Mathew, "A Study of Multi-modal Biometric System," International Journal of Research in Engineering and Technology (IJRET), vol. 03, pp. 93-98, Dec 2014.

[9] E. N. Bifari and L. A. Elrefaei, "Automated Fingerprint Identification System Based on Weighted Feature Points Matching Algorithm," in International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 2212-2217, 2014.

[10] D. Peralta, I. Triguero, R. Sanchez-Reillo, F. Herrera and J. M. Benitez, "Fast Fingerprint Identification for Large Databases," Pattern Recognition, vol. 47, pp. 588-602, August 2013.

[11] L. Hong, W. Yifei and A. Jain, "Fingerprint Image Enhancement: Algorithm and Performance Evaluation," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 20, pp. 777-789, August 1998.

[12] K. Cao, X. Yang, X. Chen, X. Tao, Y. Zang, J. Liang *et al.*, "Minutia Handedness: A Novel Global Feature for Minutiae-based Fingerprint Matching," Pattern Recognition Letters, vol. 33, pp. 1411-1421, March 2012.

[13] M. U. Munir, M. Y. Javed and S. A. Khan, "A Hierarchical K-Means Clustering Based Fingerprint Quality Classification," Neurocomputing, vol. 85, pp. 62-67, January 2012.

[14] B. Popovic, M. Bandjur and A. Raicevic, "Robust Enhancement of Fingerprint Images Obtained by Ink Method," Electronics Letters, vol. 46, pp. 1379-1380, September 2010.

[15] S. Chikkerur, A. N. Cartwright and V. Govindaraju, "Fingerprint Enhancement Using STFT Analysis," Pattern Recognition, vol. 40, pp. 198-211, 2007.

[16] S. A. Sudiro, M. Paindavoine and T. M. Kusuma, "Simple Fingerprint Minutiae Extraction Algorithm Using Crossing Number on Valley Structure," in IEEE Workshop on Automatic Identification Advanced Technologies, pp. 41-44, 2007.

[17] I. G. Babatunde, O. C. Akinyokun, A. B. Kayode and O. Olatunbosun, "Adaptive and Faster Approach to Fingerprint Minutiae Extraction and Validation," International Journal of Computer Science and Security (IJCSS), vol. 5, pp. 414-424, 2011.

[18] G. Xin, C. Xiaoguang, C. Jia, D. Zirui, L. Chongjin and F. Jufu, "A Novel Method of Fingerprint Minutiae Extraction Based on Gabor Phase," in 17th IEEE International Conference on Image Processing (ICIP), pp. 3077-3080, 2010.

[19] M. Tico and P. Kuosmanen, "An Algorithm for Fingerprint Image Postprocessing," in Record of the 34th Asilomar Conference on Signals, Systems and Computers, vol. 2, pp. 1735-1739, 2000.

[20] H. Wei, Z. Ou and J. Zhang, "Fingerprint Identification Based on Ridge Lines and Graph Matching," Proceedings of the 6th World Congress on Intelligent Control and Automation, pp. 9965-9968, 21-23 June 2006.

[21] A. Bengueddoudj, S. Akrouf, F. Belhadj and D. Nada, "Improving Fingerprint Minutiae Matching Using Local and Global Structures," in the 8th International Workshop on Systems, Signal Processing and their Applications (WoSSPA), pp. 279-282, 2013.

[22] Z. Jie, Z. Bo, L. Xinjing and J. Xiaojun, "A Matching-improved Reparation Method for Incomplete Fingerprint," in IEEE International Conference on Cloud Computing and Intelligence Systems (CCIS), pp. 75-79, 2011.

[23] M. V. P. Do Nascimento, L. Vidal Batista and N. L. Cavalcanti, "Comparative Study of Learning Algorithms for Recognition by Hand Geometry," in Systems, Man and Cybernetics (SMC), IEEE International Conference, pp. 423-428, 2014.

[24] T. A. Budi Wirayuda, D. H. Kuswanto, H. A. Adhi and R. N. Dayawati, "Implementation of Feature Extraction Based Hand Geometry in Biometric Identification Systems," in International Conference of Information and Communication Technology (ICoICT), pp. 259-263, 2013.

[25] R. Sanchez-Reillo, C. Sanchez-Avila and A. Gonzalez-Marcos, "Biometric Identification through Hand Geometry Measurements," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, pp. 1168-1171, 2000.

[26] M. S. Al-Ani and M. Abd Rajab, "Biometrics Hand Geometry Using Discrete Cosine Transform (DCT)," Science and Technology, pp. 112-117, 2013.

[27] K. Wenxiong and W. Qiuxia, "Pose-Invariant Hand Shape Recognition Based on Finger Geometry," in IEEE Transactions on Systems, Man and Cybernetics: Systems, vol. 44, pp. 1510-1521, 2014.

[28] J. Fierrez, J. Ortega-Garcia, A. Esposito, A. Drygajlo, M. Faundez-Zanuy, J. Burgues *et al.*, "Comparison of Distance-Based Features for Hand Geometry Authentication," in Biometric ID Management and Multi-modal Communication. vol. 5707, Ed.: Springer, Berlin, Heidelberg, pp. 325-332, 2009.

[29] W. Xiong, K.-A. Toh, W.-Y. Yau and X. Jiang, "Model-guided Deformable Hand Shape Recognition without Positioning Aids," Pattern Recognition, vol. 38, pp. 1651-1664, 2005.

[30] O. Polat and T. Yıldırım, "Hand Geometry Identification without Feature Extraction by General Regression Neural Network," Expert Systems with Applications, vol. 34, pp. 845-849, 2008.

[31] A. K. Jain and A. Ross, "Learning User-specific Parameters in a Multi-biometric System," in International Conference on Image Processing, vol. 1, pp. I-57-I-60, 2002.

[32] N. Charfi, H. Trichili, A. M. Alimi and B. Solaiman, "Bimodal Biometric System Based on SIFT Descriptors of Hand Images," in IEEE International Conference on Systems, Man and Cybernetics (SMC), pp. 4141-4145, 2014.

[33] M. Abdolahi, M. Mohamadi and M. Jafari, "Multi-modal Biometric System Fusion Using Fingerprint and Iris with Fuzzy Logic," International Journal of Soft Computing and Engineering (IJSCE), vol. 2, pp. 504-510, January 2013.

[34] I. Dehache and L. Souici-Meslati, "A Multi-biometric System for Identity Verification Based on Fingerprints and Signatures," in International Conference on Complex Systems (ICCS), pp. 1-5, 2012.

[35] N. Meghanathan, D. Nagamalai, N. Chaki, S. M. Anzar and P. S. Sathidevi, "Optimization of Integration Weights for a Multi-biometric System with Score Level Fusion," in Advances in Computing and Information Technology, vol. 177, Ed.: Springer, Berlin, Heidelberg, pp. 833-842, 2013.

[36] S.-u.-d. Ghulam Mohi-ud-Din, A. Mansoor, H. Masood and M. Mumtaz, "Personal Identification Using Feature and Score Level Fusion of Palm- and Fingerprints," Signal, Image and Video Processing, vol. 5, pp. 477-483, 2011.

[37] S. Ashish, Design of a Hand Geometry Based Recognition System, MSc Thesis, Computer Science & Engineering, Indian Institute of Technology Kanpur, Jan. 2007.

[38] A. Batool and A. Tariq, "Computerized System for Fingerprint Identification for Biometric Security," in 14th International Multi-topic IEEE Conference (INMIC), pp. 102-106, 2011.

[39] M. Zahedi and O. Ghadi, "Combining Gabor Filter and FFT for Fingerprint Enhancement Based on a Regional Adaption Method and Automatic Segmentation," Signal, Image and Video Processing, pp. 1-9, April 2013.

[40] R. Thai, Fingerprint Image Enhancement and Minutiae Extraction, Honours Programme of the School of Computer Science and Software Engineering, The University of Western Australia, 2003.

[41] Z. Xiaolong, W. Yangsheng, Z. Xuying and W. Zheng, "A Scheme for Minutiae Scoring and Its Application to Fingerprint Matching," in 7th World Congress on Intelligent Control and Automation (WCICA), pp. 5917-5921, 2008.

[42] A. A. Paulino, F. Jianjiang and A. K. Jain, "Latent Fingerprint Matching Using Descriptor-Based Hough Transform," in IEEE Transactions on Information Forensics and Security, vol. 8, pp. 31-

45, Jan. 2013.

[43] N. K. Ratha, K. Karu, S. Chen and A. K. Jain, "A Real-time Matching System for Large Fingerprint Databases," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 18, pp. 799-813, 1996.

[44] H. Fronthaler, K. Kollreider and J. Bigun, "Local Features for Enhancement and Minutiae Extraction in Fingerprints," in IEEE Transactions on Image Processing, vol. 17, pp. 354-363, 2008.

[45] Morphological Operations, Available at: http://www.mathworks.com/help/images/ morphological-filtering.html

[46] I. K. G. D. Putra and M. A. Sentosa, "Hand Geometry Verification Based on Chain Code and Dynamic Time Warping," International Journal of Computer Applications, vol. 38, pp. 17-22, February 2012.

[47] J. Burgues, J. Fierrez, D. Ramos, M. Puertas and J. Ortega-Garcia, "Detecting Invalid Samples in Hand Geometry Verification through Geometric Measurements," in International Workshop on Emerging Techniques and Challenges for Hand-Based Biometrics (ETCHB), pp. 1-6, 2010.

[48] Y. M. Fouda, "Fusion of Face and Voice: An Improvement," IJCSNS International Journal of Computer Science and Network Security, vol. 12, pp. 37-43, April 2012.

[49] P. Nowak, "A Comparative Study on Biometric Hand Identification," Proceedings of the 21st International Conference in Mixed Design of Integrated Circuits & Systems (MIXDES), pp. 411-414, Lublin, 2014.

[50] Fingerprint Verification Competition 2002 (FVC2002), Available at: http://bias.csr.unibo.it/ fvc2002/

[51] Fingerprint Verification Competition 2004 (FVC2004), Available at: http://bias.csr.unibo.it/ fvc2004/

[52] COEP Palm Print Database, Available at: http://www.coep.org.in/index.php?pid=367

[53] O. O. V. Villegas, H. M. Orozco, H. de Jesus Ochoa Dominguez, L. O. Maynez and V. G. C. Sanchez, Biometric Human Identification of Hand Geometry Features Using Discrete Wavelet Transform: INTECH Open Access Publisher, 2011.

[54] P. Kartik, R. Vara Prasad and S. Prasanna, "Noise Robust Multi-modal Biometric Person Authentication System Using Face, Speech and Signature Features," in Annual IEEE India Conference, pp. 23-27, 11-13 Dec. 2008.

ملخص البحث:

في هذه الورقة، اقترحنا تصميم نظام بيومتري متعدد النماذج (Multi-modal Biometric System) باستخدام لغة MATLAB يدمج بين صفتين بيومتريتين: بصمة الإصبع و هندسة اليد. النظام المقترح سوف يُدمج في مرحلة درجة المطابقة (Matching Score Level) من خلال تطبيق قاعدة الجمع الموزون (Weighted Sum Rule) المعدّلة. تم اختبار نظام البصمة باستخدام خمس قواعد بيانات معروفة بـ (FVC) واختبار نظام هندسة اليد باستخدام قاعدة البيانات المعروفة بـ (COEP). وبالتالي، لاختبار النظام البيومتري متعدد النماذج قمنا بدمج كل قاعدة من قواعد بيانات البصمة مع قاعدة بيانات اليد. وقد أظهرت النتائج التجريبية تحسناً كبيراً في النظام البيومتري متعدد النماذج مع متوسط (EER) يساوي ٣,٢٧%، في حين بلغ متوسط (EER) ٨,٨٦% و ٨,٨٩% لنظام البصمة ونظام هندسة اليد، على التوالي.

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 3, No. 1, April 2017.

DEVELOPING A COURSE TIMETABLE SYSTEM FOR ACADEMIC DEPARTMENTS USING GENETIC ALGORITHM

Mohammad A. Al-Jarrah[1], Ahmad A. Al-Sawalqah[2] and Sami F. Al-Hamdan[3]

(Received: 13-Jun.-2016, Revised: 10-Nov.-2016, Accepted: 15-Jan.-2017)

ABSTRACT

Preparing course timetables for universities is a search problem with many constraints. Exhaustive search techniques in theory can be used to develop course timetables for academic departments, but unfortunately these techniques are computation intensive, since the search space is very large and therefore are impractical. In this paper, Genetic Algorithms (GA's) are utilized to build an automated course timetable system. The system is designed for any academic department. The proposed timetabling system requires minimal effort from the administration staff to prepare the course timetable. Moreover, the prepared course timetable considers faculties' desires, students' needs and available resources, such as classrooms and laboratories with optimal utilization.

The proposed timetabling process was divided into three stages. The first stage is the data collection stage. In this stage, the administrative staff; usually the head of the department, is responsible for preparing the required data, such as the names of the faculty personnel and their desires of courses and laboratories ordered with some priority scheme. Number and type of theoretical and practical courses are also fed to the system based on some statistics about student numbers and previous course timetable history. The system is also fed with number of lecture rooms allocated for the department and number of labs with information about theoretical courses they are able to serve. In the second stage, the program generates an initial set of suggested schedules (chromosomes). Each chromosome represents a solution to the problem, but usually is not satisfactory. Finally, the proposed timetabling system starts the search for a good solution that satisfies best interests of the department according to a cost function. GA is applied in search for a satisfactory course timetable based on a pre-defined criterion. The system has been developed and tested utilizing benchmarked datasets developed by an international timetabling competition (ITC2007) and for the Computer Engineering Department at Yarmouk University. In both cases, the algorithm showed very satisfactory results.

KEYWORDS

Courses timetable problem, Courses timetable generation, Genetic algorithm, Chromosome generation, Parents selection algorithm, Crossover, Mutation.

1. INTRODUCTION

The head of an academic department in any university is usually responsible for preparing a course timetable every semester. Preparing course timetables is a time-consuming task which academic colleges face. Course timetabling is not only formulating a timetable for courses, but also has to be performed based on many constraints, such as classroom availability and capacity, interference between rooms and courses and conflicts between courses and instructors. Very early in 1999, class timetabling at Sirindhon International Institute of Technology has been

1. M. Al-Jarrah is with the Computer Engineering Department, Yarmouk University, Irbid, Jordan. E-mail: jarrah@yu.edu.jo.
2. A. Al-Sawalqah is with the Communications & Electronics & Computer Engineering Department, Tafila Technical University, Tafila, Jordan. E-mail: ahmadsw@ttu.edu.jo.
3. S. Al-Hamdan is with the Computer Engineering Department, Yarmouk University, Irbid, Jordan. E-mail: shamdan@yu.edu.jo.

tackled based on the mentioned constraints. A cost function was defined for each one to utilize the existing facilities and resources effectively [1].

Generally, course timetabling in many universities is prepared manually based on administration experience. The administrator should consider all available facilities and resources, such as courses, instructors, rooms and laboratories. Moreover, the instructors' time and time of course sections are important constraints to handle. Therefore, based on all the mentioned constraints, course timetabling is a very exhaustive and time-consuming task.

Course timetabling is one of the Nondeterministic Polynomial-time (NP) hard problems [1], [12], [23]-[25]. Heuristic search algorithms are usually used in most of these problems to find a near-optimal solution. Nevertheless, this only works for simple cases. For more complex inputs and requirements, obtaining a considerably good solution can take long time and effort [2]-[3], [26]-[27].

Genetic algorithms (GA's) are a class of non-traditional techniques which can handle complex and large search spaces to a certain extent in problem solving [4]-[6], [14]-[16]. In some GA approaches, the search space is divided into subspaces to increase the probability of having good chromosomes in the initial population. This technique increases the chances of finding a near-optimal solution resulting usually in a reduction of computation time [17]-[20]. In the following sections, we propose an automated course timetabling system using genetic algorithms.

Preparing course timetables for academic departments usually utilizes different approaches, such as tabu search, simulated annealing and fuzzy logic [7]-[8], [21]. A combined approach including GA and Hill Climbing has been also introduced to prepare course timetables for academic departments [9]. Another combined approach has also been presented for solving timetabling problem using honey-bee mating optimization algorithm [10]. In [11], an evolutionary approach to solve university course timetabling problems has been introduced.

Wren [22] defines timetabling as follows: "Timetabling is the allocation, subject to constraints, of given resources to objects being placed in space time, in such a way as to satisfy as nearly as possible a set of desirable objectives''. Lü *et al.* [28] developed a timetabling algorithm based on tabu search. Bonutti *et al.* developed a curriculum-based course timetabling for universities. The best known solutions have been discussed and compared [29]. Muller, the winner of the International Timetabling Competition 2007, introduced his algorithm based on hill climbing and great deluge technique.

In the course timetabling problem, an optimal solution is defined as the best utilization of available expertise, resources and time for best delivered services. The best utilization is defined under a set of constraints. The constraints are divided into hard and soft constraints. In this paper, the soft and hard constraints used are those related to time, room, instructors and courses. Also, some of the constraints are defined based on type of room (availability of a data show for example), instructor rank and type of course (theoretical or practical). All these constraints have been considered in the proposed algorithm to develop a satisfactory course timetable with minimal effort. Moreover, the proposed algorithm generates a course timetable that satisfies all hard constraints.

The rest of this paper is organized as follows: In section 2, course timetabling using genetic algorithms is discussed. In section 3, automated timetabling using GA's is described. In Section 4, the implementation and experimental results are illustrated for the proposed course timetabling using GA's. Finally, conclusions are presented in section 5.

2. COURSE TIMETABLING USING GENETIC ALGORITHMS

Course timetable is a table which contains information about courses offered for students to register in. Course timetabling is one of the major time-consuming tasks performed frequently

by heads of academic departments. The inputs for the process for preparing a course timetable can be formulated as follows:

1- Set of courses (*Co*) and number of offered sessions for each course, such that $Co = \{(c_1, s_1, ct1, sz_1), (c_2, s_2, ct2, sz_2) \ldots, (c_n, s_n, ctn, sz_n)\}$, where c_i is the i^{th} course, s_i is the number of sessions for course c_i, ct_i is the course type and sz_i is the number of students. Course type has two values; representing theoretical or practical.

2- Set of instructors (Ins), such that Ins= {(I1, L1), (I2, L2),…..,(Im, Lm)}, where Ii is instructor i and Li is the load of instructor i.

3- Set of lecture rooms and laboratories (R), such that R = {(r1, t1), …. , (rs, ts)}, where ri is room number I and ti is the room type. Room type has two values; representing lecture room or laboratory room.

4- Set of time slots (TS), such that TS= {(d1, s1), ….. (dn,Sn)}, where di is the day group selection and Si is the i^{th} slot of time from the selected day group. In universities, the week days are divided into two groups. For example, the first group includes Sunday, Tuesday and Thursday, whereas the second group includes Monday and Wednesday. Moreover, the first group is divided into time slots, such that each slot is 60 minutes, and the second group is divided into time slots, where the slot is 90 minutes. University departments offer their courses on a daily basis. For example, the course may be offered on Sunday, Tuesday and Thursday from 8:00AM to 9:00AM or on Monday and Wednesday from 8:00AM to 9:30AM for the entire semester.

Based on the inputs, the task is to prepare course timetabling that satisfies all hard constraints and most possible soft constraints. The hard constraints are:

H1- Scheduling all courses listed in *Co* list.
H2- Assigning an instructor to each course and satisfying instructors load according to *Ins* set.
H3- Assigning a lecture room to each course and satisfying room type according to *Co* list, where practical courses must be in a laboratory room.
H4- Sessions of the same course should not be in the same time slot.
H5- The lecture room size must be greater than the number of students in the *Co* set.
H6- An instructor must not have more than one lecture in the same time slot.

The soft constraints are:

S1- Room capacity to expected attending students. To satisfy this constraint, the ratio of the number of students in *Co* list to room size must be more than 0.85.
S2- Room utilization. It is expected to utilize lecture room all the time. To satisfy this constraint, the ratio of the utilization time to the total room time must be more than 0.90.
S3- Instructor preferable courses. This constraint is satisfied if the assigned course to an instructor is one of the preferred courses of this instructor.
S4- Instructor preferable lecturing time slots and not preferable time slots. This constraint is satisfied if the assigned course time to an instructor is one of the preferred time slots of this instructor.
S5- Instructor teaching experience. It is preferred to assign a course to an instructor who taught it. This constraint is satisfied if the course assigned to an instructor has been previously

taught by this instructor.
S6- Instructor timetable compactness. This constraint measures the distribution of instructor timetable. To satisfy this constraint, first, instructor timetable should not include more than two adjacent lectures and the waiting time between lectures should not be more than one time slot.

The process of preparing a course timetable considers the inputs, such that the resulting course timetable must satisfy all hard constraints and all possible soft constraints. Therefore, the process of preparing a course timetable is not a straight forward process and the solution is not unique. Moreover, preparing a course timetable is very costly in terms of time and effort if it's prepared manually. Preparing a course timetable automatically using intelligent algorithms has been investigated as mentioned earlier in the introduction section.

Genetic algorithms are one of the methods that reduce computation cost in comparison with other artificial intelligence approaches, such as heuristic search tree, particularly for hard or complex problems like preparing courses timetables. GA's clone biological evolution and consist of simple steps that go through several iterations (generations) until a satisfactory solution is reached. For course timetabling system, a satisfactory solution is one that satisfies all hard constraints and as much as possible of soft constraints. For example, the load for an instructor of the rank assistant or associated professor should not exceed 12 credit hours, while it is 9 credit hours for a full professor. This is a hard constraint; however, the desire of a certain instructor of having his courses after 10 O'clock is a soft constraint that is good to satisfy, but is not necessary. Assuming that the problem has an optimal solution, achieving 100% satisfactions for all constraints (optimal) is computationally very costly.

In GA's, a solution is represented as a chromosome. Population is a set of chromosomes that represent the solution subspace. New generations of chromosomes are generated through two basic genetic operations: crossover and mutation. In crossover, two genes, where a gene is a part of a chromosome, containing a part of the solution, are swapped between the mated pair of chromosomes. The new chromosome would be considered as an accepted, not accepted or satisfactory solution [9]. In mutation, a gene is completely changed randomly for the selected chromosomes. Crossover occurs much more frequently than mutation. GA's assume that the new generations of chromosomes will have better qualities than their ancestors. The process of crossover and mutation is repeated until a near-optimal solution for the problem in hand is reached [10].

GA's have many features that make them in some occasions the best problem solving method. GA's can be used in problems where the search space is usually large and complex when traditional methods need long time and high computation systems [13]. Furthermore, GA's are considered as intelligent systems, where the knowledge is built based on last best knowledge until a satisfactory solution is reached. This means that GA's do not require generating the full search space.

GA's have a parallel nature in which parallelism can be explored and used making it possible to speed up the calculations. To identify a near-optimal solution, each chromosome in the population has a fitness value. The fitness value for a chromosome defines how much it is close to the optimal solution. Genetic processing is applied to improve the fitness value of new generations of chromosomes. If near-optimal solution is not found after a reasonable number of search iterations, then the process is started all over again with a new set of chromosomes. Hopefully, the new running chromosomes will lead to the required near-optimal solution.

3. AUTOMATED COURSE TIMETABLING SYSTEM USING GENETIC ALGORITHM

The proposed algorithm for course timetabling utilizes genetic algorithms to generate a near-optimal course timetabling in a relatively short period of time and with a little effort from the administrator side. Figure 1 illustrates the proposed algorithm for course timetabling system utilizing GA's. The final schedule is achieved if the solution satisfies all predefined hard constraints. The main loop of search includes genetic operations, which include mutation and crossover.

According to Figure 1, the timetabling is divided into two stages. The first stage includes chromosome generation and population initialization. This stage starts after collecting data and constraints from administrators who are responsible to prepare the course timetable. The second stage includes the process of applying genetic operations, which are crossover and mutation of selected chromosomes. Each chromosome represents a solution to the problem (course timetable), but usually is not optimal. In this stage, we select chromosomes called parents to apply genetic operations. Operations will generate new chromosomes that have properties better than their parents. The generated chromosomes will be added to the population. The new population is evaluated to check if one of the chromosomes achieved the targeted goal.

3.1 Chromosome Generation and Population Initialization

In this subsection, we discuss the first stage that handles the generation of chromosomes and adding them to the population. The generated chromosomes are generated randomly. Each gene is selected randomly from the gene pool. The following steps explain how to generate a chromosome.

a. The course sessions that must be offered by the department are determined by the department administration and considered as an input list for the population initialization process. The set of courses is included in the Co set.
b. The chromosome genes are course genes, instructors' genes, days and time of lectures and lecture room or laboratory room.
c. Select session gene from the input list randomly and assign this value to a chromosome; if the input list is empty, go to step h.
d. Complete the remaining genes for this chromosome part. For example, select randomly an instructor from the instructor pool, a time from the time pool and a lecture room from the halls' pool. The process is accomplished for every gene in the chromosome.
e. For the generated chromosome part, check if the course is a laboratory one, then reselect the hall from the laboratory halls' pool.
f. After considering all items of the input list, the obtained chromosome represents an initial solution. Add this chromosome to the initial population and reload the input list.
g. Repeat steps c to f *N* times, where *N* is the number of chromosome in the population.
h. End of population initialization.

The result of population initialization is a set of solutions. The near-optimal solution which we are searching for is most probably not one of the initial generations. Therefore, the fitness value is computed using the proposed fitness function for each chromosome generated during the population initialization process.

3.2 Fitness Functions

The fitness function is computed and evaluated for every chromosome in the population. The fitness is measured through quantifying constraints. The constraints are divided into two groups; hard constraints and soft constraints. A hard constraint is a constraint that makes the course timetable invalid if not satisfied by the solution. For example, an instructor is assigned to teach two courses at the same time. Thus, for a valid course timetable, all hard constraints must be achieved. A soft constraint is one that is strongly recommended to satisfy, but does not make the solution invalid if not satisfied. For example, some instructors prefer to have their lectures early, while others prefer to have them at late times. A satisfied constraint is a constraint that will have a positive value (+2) on the fitness function and an unsatisfied one will have a negative value (-2). However, a satisfied soft constraint will have also a positive contribution (+1) on the fitness, while an unsatisfied one will have a zero effect on the fitness function (see Equations 1 and 2).

For hard constraints:

$$HC_i = \begin{cases} 0. \ if\ hard\ constraint\ i\ is\ accheived \\ 2. \ if\ hard\ onstraint\ i\ is\ not\ accheived \end{cases} \quad (1)$$

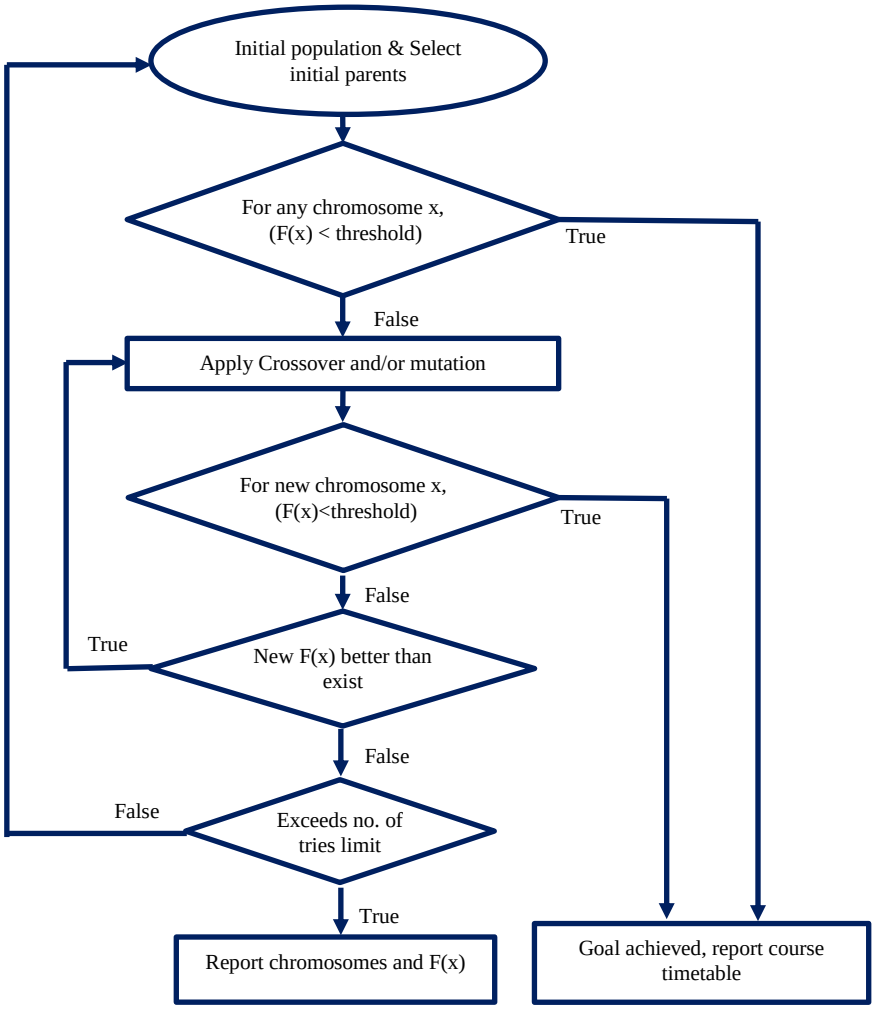

Figure 1. The proposed system for preparing a course timetable using GA's.

For soft constraints, (SCi) is calculated as follows:

$$SC_i = \begin{cases} 0. \ if\ soft\ constraint\ \ i\ is\ accheived \\ 1. \ if\ soft\ constraint\ \ i\ is\ not\ accheived \end{cases} \quad (2)$$

The fitness function will include all constraints. Thus, the fitness function is computed for a certain chromosome *x* as follows:

$$F(x) = \sum_{i=1}^{L} HC_i + \sum_{i=1}^{M} SC_i \quad (3)$$

where *L* is the number of hard constraints and *M* is the number of soft constraints. This fitness function ranges from 0 to (2N+M). According to Equation (3), the chromosome with a fitness value equal to zero is a chromosome that stratified all constraints. Thus, the ultimate goal for our algorithm is to obtain a chromosome with a fitness value equal to zero. However, finding

such a chromosome is not guaranteed by GA's and if possible will usually be computationally very costly. Thus, we defined the following condition, which we named *GOAL* in the proposed algorithm, to stop the search loop as follows:

$$GOAL = \begin{cases} True. \quad for\ any\ x. if\ F(x) < threshold. and\ HC(x) = 2L \\ False. \quad otherwize \end{cases} \tag{4}$$

The value of *HC(x) = 2L* if all constraints are achieved.

3.3 Chromosome Selection

After *N* chromosomes are added to the initial population, the fitness function is calculated for each candidate in the population. Most likely, there is no chromosome in the initial population which satisfies the *GOAL*; therefore, a process of genetic operations will be applied on the set of solutions called parents. Two chromosomes will be mated together to produce new chromosomes. The selection of these two chromosomes is dynamic and depends on the fitness value for each chromosome. The probability of selecting a chromosome to be a parent depends on its fitness value. The probability value for selecting a chromosome (x) is defined as:

$$p(x) = 1 - \frac{F(x)}{\sum_{i=1}^{N} F(i)} \tag{5}$$

where *N* is the number of chromosomes in the population and *F(x)* is the fitness of the chromosome. According to Equation 5, the chromosome with a higher fitness value has a better chance to be chosen for mating operations.

After selecting parents, the genetic operators are applied. From each operation, new chromosomes are generated and the fitness values for these chromosomes are computed for the next iteration.

3.4 Applying Genetic Algorithms Operation (Mutation and Crossover)

Genetic algorithms have two types of operation: crossover (mating) and mutation. Chromosome mating operation is usually accomplished utilizing a pair of chromosomes that are selected as parents by chromosome selection process. The crossover operation includes the exchanges of instructor, classroom, course or time genes in a random manner. The result of the mating operation is two new chromosomes that would have the good properties of their parents. The fitness function is computed for the new chromosomes. If the fitness values for the new chromosomes are greater than those of their parents, these new chromosomes are added to the population. Otherwise, they are discarded and the process is repeated.

The mutation is the process of changing a gene in parent chromosome with a new gene. In the proposed algorithm, the mutation is accomplished as follows:

1- Select a chromosome from the population as a parent one.
2- Randomly, select which gene is to be replaced. The targeted genes are course, instructor, classroom or time slot.
3- Select randomly a new gene from the gene pool. The selected gene type must be similar to that of the one selected in step 2. For example, if we decided to replace an instructor gene, then the pool will be the instructors' pool in the department.
4- Replace the new gene with the old one.
5- Calculate the fitness function for the new chromosome. If the fitness value for the new one is better than that of the parent, then add the new chromosome to the population. Otherwise, discard the new chromosome and repeat the process.

The mutation process is applied every while or when crossover operation does not achieve an improvement. Thus, the probability of applying mutation is increased with respect to number of

unsuccessful crossover operations and with respect to the difference between the best fitness values for the population before crossover and the fitness values for the population after crossover operation. The implementation of randomly applying mutation is achieved through defining a mutation counter. The value of the counter is initially zero and every time the crossover failed to improve the fitness value, this counter is incremented by one. Then, at the end of every loop, we randomly generate an integer between 0 and 10000. If the generated number is less than the mutation counter, we apply the mutation process and reset the mutation counter.

4. PROPOSED ALGORITHM IMPLEMENTATION AND RESULTS

The proposed algorithm for developing course timetables for academic departments using genetic algorithms is implemented using Microsoft Visual Studio. We utilized C# language to develop the application which generates the course timetable. The users for this application include the head of the department and the department instructors. The head of the department (HOD) interacts with the application using a Windows Application. HOD inserts the list of courses and their attributes, list of instructors, as well as list of resources such as lecture rooms and laboratories, list of hard constraints and list of soft constraints. Then, instructors interact with the application through a web application. The instructors insert their soft constraints. Finally, the HOD approves the constraints of the instructors and initiates the process of generating the course timetable.

The developed application was tested and verified for the Department of Computer Engineering at Yarmouk University. HOD inserted lists of 45 courses, 13 laboratories and number of section to be offered for every course and laboratory. Then, every instructor inserted preferred courses as well as preferred and not preferred time slots. Also, each instructor inserted a set of previously taught courses.

The second phase of generating the course timetable was started through initiating the second stage of the algorithm. The system generates 30 chromosomes. It took 33 seconds to generate the initial population. This experiment was conducted utilizing Intel Pentium i3 CPU and 4 GB memory. The next step was applying genetic operations according to the proposed algorithm. Table 1 shows a sample output of applying mutation and crossover operations. The first row of Table 1 shows that the processed crossover operation at iteration 10079 has two parents with 18 and 35 fitness values. The new chromosome has a fitness value of 42, which is better than the parents' value. Thus, this chromosome is added to the population. The second row of Table 1 shows the process of crossover at iteration number 10210. The new chromosome has a fitness value equal to 22, which is less than its parents' values. Thus, the new chromosome is ignored and the process is restarted. The mutation operation is applied in iteration number 1209. The new chromosome fitness value is equal to 41, which is better than its parents' fitness values. Therefore, this chromosome is added to the population.

The implemented algorithm generated a course timetable for the Department of Computer Engineering at Yarmouk University. The best course timetable was generated after more than 32000 iterations. The generated course timetable was distributed to the instructors and they were asked to evaluate it. The evaluation was 9.56 out of 10. This evaluation shows a high degree of satisfaction among the instructors and the department administration. For the purpose of validating the proposed system with other timetabling approaches, we modified our system to read the input from text file. The file contains a dataset for curriculum-based course timetabling prepared by international timetabling competition [29]-[30]. The dataset has information about offered courses, rooms, curricula and unavailability constraints. Moreover, we have modified the scheduling process in our proposed system to consider courses have minimum working days. In our original proposed system, all offered courses span over one semester. Moreover, in the original proposed system, the instructor can be selected from a pool. Thus, the preference of

lecturers to teach a course is considered as soft constraint, while the selection of a lecturer from the pool is a hard constraint.

Table 1. Sample output for crossover and mutation operation.

GA's Operation	Chromosome Situation	Operation Fit Value
Crossover	Chromosome accepted	Iteration = 10079 F(1)= 18, F(11)= 35 F(Pc)= 42
Crossover	Chromosome not accepted	Iteration = 10210 F(1)= 17 , F(11)= 26 F(Pc)= 22
Mutation	Chromosome accepted	Iteration = 1209 F(8)= 24 , F(Pc) = 41

Table 2. Obtained minimum number of satisfied soft constraints of the proposed system for five datasets and comparison with other algorithms.

Dataset	Proposed system	Lü *et al.* [28]	Bonutti *et al.* [29]	Muller [30]
comp01	4	4	5	5
comp02	20	22	75	43
comp03	34	41	93	72
comp04	21	19	45	35
comp05	203	224	326	298
comp06	18	25	62	41
comp07	6	4	38	14

After the modification of the proposed system to comply with all requirements of the international timetabling competition, we downloaded 5 datasets from the international timetabling competition website and computed the number of violated hard constraints and soft constraints for each dataset. The scheduling process is stopped based on timeout condition according to ITC-2007 competition [30]. Because the scheduling algorithm is stopped based on timeout condition, the process may have different results for different runs. Thus, we applied the proposed algorithm fifty times and considered the best results. Table 2 summarizes the obtained results and the results of the other scheduling algorithms that utilized the same dataset. According to Table 2, the proposed algorithm outperforms other approaches on most listed datasets.

5. CONCLUSIONS

In this paper, we proposed a new approach to generate course timetables for academic departments at universities. The proposed approach utilizes Genetic Algorithms to develop course timetables. The proposed approach is divided into three stages. The first stage is inputting the course list and number of sections in each one, instructors and their constraints, rooms available, hard and soft constraints for rooms and lab resources. The second stage generates a set of initial solutions (chromosomes) randomly called initial population. In the third stage, the initial solutions are subjected to GA operations (crossover and mutation) repeatedly

until a near-optimal solution is reached that achieves a predefined (threshold) fitness function. The closer to zero, the better is the solution.

The fitness function was computed for each solution. The value of the fitness function indicates the goodness of the solution. After that, genetic operators which are crossover and mutation, were applied. The goal of applying these operators is to obtain new solutions with better fitness values. As a feature in genetic algorithm, crossover is applied more frequently than mutation. Applying genetic operators was repeated until we obtained a satisfying solution according to an instructors' satisfaction questionnaire. The proposed algorithm was implemented and utilized to generate a course timetable for the Department of Computer Engineering at Yarmouk University. An evaluation process of the generated course timetable among instructors was found to be very satisfactory. Moreover, we modified our code to read inputs from text file formatted according to benchmarked datasets. Then, we tested the proposed algorithm utilizing seven benchmarked datasets. The results of our algorithm outperformed other methods in most datasets.

REFERENCES

[1] J. Nakasuwan, P. Srithip and S. O. Komolavanij, "Class Timetabling Optimization," Thammasat International Journal of Science and Technology, vol. 4, no. 2, pp. 88-98, 1999.

[2] P. Chang, S. Chen and K. Lin, "Two-Phase Sub-Population Genetic Algorithm for Parallel Machine-timetabling Problem," Expert Systems with Applications, vol. 29, no. 3, pp. 705-712, 2005.

[3] H. Babaei, J. Karimpour and A. Hadidi, "A Survey of Approaches for University Course Timetabling Problem," Journal of Computers & Industrial Engineering, vol. 86, pp. 43-59, 2015.

[4] P. Guo, J. Chen and L. Zhu, "The Design and Implementation of Timetable System Based on GA's," IEEE International Conference on Mechatronics Science, Electric Engineering and Computer (MEC), pp. 1497-1500, 2011.

[5] A. George, T. Marwala and F. Nelwamondo, "Use of Data Mining in Scheduler Optimization," arXiv preprint arXiv:1011.1735, 2010.

[6] K. Deb, "Multi-objective Genetic Algorithms: Problem Difficulties and Construction of Test Problems," Evolutionary Computation, vol. 7, no. 3, pp. 205-230, 1999.

[7] C. Teoh, A. Wibowo and M. Ngadiman, "Review of State of the Art for Metaheuristic Techniques in Academic Timetabling Problems," Artificial Intelligence Review, vol. 44, no. 1, pp. 1-21, 2015.

[8] S. Mir Hassani and F. Habibi, "Solution Approaches to the Course Timetabling Problem," Artificial Intelligence Review, vol. 39, no. 2, pp. 133-149, 2015.

[9] M. El-Sherbiny, R. Zeineldin and A. El-Dhshan, "Genetic Algorithm for Solving Course Timetable Problems," International Journal of Computer Applications, vol. 124, no. 10, 2015.

[10] N. Sabar, M. Ayob, G. Kendall and R. Qu, "A Honey-bee Mating Optimization Algorithm for Educational Timetabling Problems," European Journal of Operational Research, vol. 216, no. 3, pp. 533-554, 2012.

[11] J. Obit, D. Ouelhadj, D. Landa-Silva and R. Alfred, "An Evolutionary Non-Linear Great Deluge Approach for Solving Course Timetabling Problems," IJCSI International Journal of Computer Science Issues, vol. 9, no. 4, pp. 1-13, 2012.

[12] V. Sapru, K. Reddy and B. Sivaselvan, "Timetabling Using Genetic Algorithms Employing Guided Mutation," IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), pp. 1-4, 2010.

[13] A. Page and J. Naughton, "Dynamic Task Timetabling Using Genetic Algorithms for Heterogeneous Distributed Computing," Proceedings of 19th IEEE International Symposium in Parallel and Distributed Processing, pp. 189a-189a, April 2005.

[14] S. Wu, H. Yu, S. Jin, K. Lin and G. Schiavone, "An Incremental Genetic Algorithm Approach to Multiprocessor Timetabling," IEEE Transactions on Parallel and Distributed Systems, vol. 15, no. 9, pp. 824-834, 2004.

[15] E. Vallada and R. Ruiz, "A Genetic Algorithm for the Unrelated Parallel Machine Timetabling Problem with Sequence-Dependent Setup Times," European Journal of Operational Research, vol. 3, pp. 612-622, 2011.

[16] T. Dereli and H. Filiz, "Allocating Optimal Index Positions on Tool Magazines Using Genetic Algorithms," Robotics and Autonomous Systems, vol. 33, no. 2, pp. 155-167, 2000.

[17] J. Chen, D. Goldberg, S. Ho and K. Sastry, "Fitness Inheritance in Multi-Objective Optimization," Proceedings of the Genetic and Evolutionary Computation Conference, pp. 319-326, 2002.

[18] F. Pezzella, G. Morganti and G. Ciaschetti, "A Genetic Algorithm for the Flexible Job-shop Timetabling Problem," Computers & Operations Research, vol. 35, no. 10, pp. 3202-3212, 2008.

[19] M. Carvalho, A. Laender, M. Gonçalves and A. Da Silva, "A Genetic Programming Approach to Record Deduplication," IEEE Transactions on Knowledge and Data Engineering, vol. 24, no. 3, pp. 399-412, 2012.

[20] G. Kim and C. Lee, "Genetic Reinforcement Learning Approach to the Machine Timetabling Problem," Proceedings of IEEE International Conference on Robotics and Automation, vol. 1, pp. 196-201, 1995.

[21] M. Ghaith and A. Masri, "Scatter Search for Solving the Course Timetabling Problem," 3rd IEEE Conference on Data Mining and Optimization (DMO), Selangor, Malaysia, 28-29 June 2011.

[22] A. Wren, "Scheduling, Timetabling and Rostering – A Special Relationship?," International Conference on the Practice and Theory of Automated Timetabling, pp. 46-75, 1996.

[23] A. Bettinelli, V. Cacchiani, R. Roberti and P. Toth, "An Overview of Curriculum-based Course Timetabling," TOP, vol. 23, no. 2, pp. 313-349, 2015.

[24] V. Cacchiani, A. Caprara, R. Roberti and P. Toth, "A New Lower Bound for Curriculum-based Course Timetabling," Computers & Operations Research, vol. 40, no. 10, pp. 2466-2477, 2013.

[25] E. Özcan, A. Parkes and A. Alkan, "The Interleaved Constructive Memetic Algorithm and Its Application to Timetabling," Computers & Operations Research, vol. 39, no. 10, pp. 2310-2322, 2012.

[26] H. Rudová, T. Müller and K. Murray, "Complex University Course Timetabling," Journal of Scheduling, vol. 14, no. 2, pp. 187-207, 2011.

[27] S. Jat and S. Yang, "A Hybrid Genetic Algorithm and Tabu Search Approach for Post Enrolment Course Timetabling," Journal of Scheduling, vol. 14, no. 6, pp. 617-637, 2011.

[28] Z. Lü and J. Hao, "Solving the Course Timetabling Problem with a Hybrid Heuristic Algorithm," International Conference on Artificial Intelligence: Methodology, Systems and Applications, pp. 262-273, 2008.

[29] A. Bonutti, F. De Cesco, L. Di Gaspero and A. Schaerf, "Benchmarking Curriculum-based Course Timetabling: Formulations, Data Formats, Instances, Validation, Visualization and Results," Annals of Operations Research, vol. 194, no. 1, pp. 59-70, 2012.

[30] T. Muller, ITC2007: Solver Description, Technical Report, Purdue University, 2008.

"Developing a Course Timetable System for Academic Departments Using Genetic Algorithm", Mohammad A. Al-Jarrah, Ahmad A. Al-Sawalqah and Sami F. Al-Hamdan.

ملخص البحث:

إنّ إعداد جداول المحاضرات في الجامعات مسألة بحثية تنطوي على العديد من المحدّدات. وهناك تقنيات يمكن استخدامها من الناحية النظرية من أجل إعداد جداول المحاضرات الخاصة بالأقسام الأكاديمية، إلا أنها لسوء الحظ تحتاج الى الكثير من الحسابات لأن حيز البحث واسع جداً، وهذا يجعل تلك التقنيات غير عملية. في هذه الورقة، يتم استخدام الخوارزميات الوراثية لبناء نظام آلي لإعداد جداول المحاضرات. وهذا النظام مصمّم بحيث يُلائم أيّ قسم أكاديمي. ولا يتطلب النظام المقترح سوى القليل من الجهد من جانب الطاقم الإداري لإعداد الجدول المطلوب. من جهة أخرى، يراعي الجدول المعدّ بهذه الطريقة رغبات أعضاء هيئة التدريس واحتياجات الطلبة والموارد المتاحة مثل الغرف الصفية والمختبرات بفاعلية مثالية.

وقد تم تطوير النظام واختباره باستخدام مجموعات بيانات مرجعية عالمية لإعداد جداول المحاضرات، وطُبّق النظام على قسم هندسة الحاسوب في جامعة اليرموك. وقد أسفر اختبار النظام وتطبيقه عن نتائج مُرضية جداً.

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 3, No. 1, April 2017.

BIG DATA IN HEALTHCARE: REVIEW AND OPEN RESEARCH ISSUES

Mohammad Ashraf Ottom

(Received: 18-Oct.-2016, Revised: 29-Dec.-2016, Accepted: 21-Jan.-2017)

ABSTRACT

The globe is generating a high volume of data in all domains, such as social media, industries, stock markets and healthcare systems. Most of data volume has been generated in the past two years. This massive amount of data can bring benefits and draw knowledge to individuals, governments and industries and assist in decision making. In healthcare, an enormous volume of data is generated from healthcare providers and stored in digital systems. Hence, data are more accessible for reference and future use. The ultimate vision for working with health big data is to support the process of improving the quality of service in healthcare providers, reducing medical mistakes and providing a promoting consultation in addition to providing answers when needed. This paper provides a critical review of some applications of big data in healthcare, such as the flu-prediction project by the Institute of Cognitive Sciences, which combines social media data with governmental data. The project aim is to provide swift response about flu-related questions. The project should study human multi-modal representations, such as text, voice and images. Moreover, integrating social media data with governmental health data could create some challenges, because governmental health data are considered as more accurate than subjective opinions on social media. Another attempt to utilize big data in healthcare is Google Flu Trends GFT. GFT collects search queries from users to predict flu activity and outbreak. GFT performed well for the first two to three years; however, it started to perform worse since 2011 due to people behaviour changes. GFT did not update the prediction model based on new data released by the Centre for Disease Control and Prevention-US (CDC). On the other hand, ARGO (Auto Regression with Google) performed better than all previously available influenza models, because it adjusts people behaviour changes and relies on current publicly available data from google-search and CDC. This research also describes, analayzes and reflects the value of big data in healthcare. Big data has been introduced and defined based on the most agreed terms. The paper also explains big data revenue forecast for the year 2017 and historical revenue in three main domains: services, hardware and software. Big data management cycle has been reviewed and the main aspects of big data in healthcare (volume, velocity, variety and veracity) have been discussed. Finally, a discussion has been made of some challenges that face individuals and organizations in the process of utilizing big data in healthcare, such as data ownership, privacy, security, clinical data linkage, storage and processing issues and skills requirements.

KEYWORDS

Big data, Healthcare, Hadoop, Google flu trends, Big data challenges, Cycle of big data management.

1. INTRODUCTION

Some people could be amazed about the fact that 90% of the current volume of data have been generated in the past two years and that the amount of data is expected to grow 40% per annum according to Aureus Analytics [1]. Big data is a vast and vague terminology that carries many meanings depending on time and technology generation. For example, one megabyte of data in the 1950s was considered as a big amount of data when IBM manufactured the disk file IBM-350 with a total capacity of a couple of megabytes [2], where the US presidential debate between Barack Obama and Mitt Romney in 2012 has produced about 10.3 million tweets in ninety minutes [3]. In addition, other social media and business transactions produce an

A. Ottom is with the Computer Information Systems Department, Yarmouk University, Irbid, Jordan.
E-mail: ottom.ma@yu.edu.jo.

enormous amount of data in every minute. Therefore, the term big data is a changing term depending on time, since what we consider as "big" in the present may not be big in the future.

The main objective of this manuscript is to describe a current issue in Information Technology research, which is big data in healthcare field. This manuscript is to provide insight into values and opportunities of big data in healthcare industry. It also attempts to draw some attention to market revenues and economic impacts of big data. Some big data applications in healthcare are reported and commented. Some related research issues are identified and discussed. In addition, the paper criticizes some applications of big data in healthcare, such as the flu-prediction project by the Institute of Cognitive Sciences, Google Flu Trends and Auto Regression with Google (ARGO).

A study [4] investigated the term big data to establish a solid definition that describes what is actually meant by big data. The study found that big data is more frequent in four domains: information, technologies, methods and impacts. The study proposed the most prominent definition: “Big data is the information assets characterized by such high volume, velocity and variety to require specific technology and analytical methods for their transformation into value”. In information technology, big data could be a huge amount of data that is beyond the traditional database capabilities [5]-[6].

Several reports claim that every day the globe produces 2.2 terabytes of data in which only 10% are structured data, where 90% are unstructured data. Further, 90% of data have been generated in the past two years [1]. Nowadays, organizations are holding and collecting a huge amount of data, but do not know what they have and how to utilize the collected data properly. This is like a person who has a big library at home or in his office, but never read the books. However, recently the term big data is used to represent the huge data that people generate around the globe with a little debate about its importance. Big data has created some controversy among researchers about the data significance and future importance. Some supporters claim that big data could be the new trend for innovation and discovering knowledge from outsized amounts of data in the near future and will lead to the evolvement of a new the science in parallel with computer machine learning and data mining advancement [7]-[8]. On the other hand, some disputed big data importance and stated that bigger data is not always better [9]. Michael Jordan in an interview with IEEE spectrum stated about big data: “It is like having billions of monkeys typing. One of them will write Shakespeare”. However, he admitted that data analysis can produce inferences about a certain problem, but with a certain level of quality and with an error margin [10]. Big data also raises an argument about the future of data mining in big data era and the possibility of big data to replace data mining completely [7]. This argument needs in-depth future research to let see what the coming days carry for data mining.

Despite the debate created around the term big data, a recent study [11] showed the importance and market revenue of big data vendors, such as HP, IBM, DELL, SAP, ORACLE and SAS in three major domains: hardware, software and services. The study found that the total revenue reached US$ 18.6 billion in 2013, where 40% of the revenues were in the service domain, 38% in the hardware domain and 22% in the software domain. Figure 1 shows big data revenues by domain for the year 2013.

The study [11] also forecasted the market revenues for the year 2017 and compared the revenues for the years from 2011 until 2017. The market revenue for 2011 was US$ 7.3 billion and is expected to reach $US50.1 billion in 2017. Figure 2 shows the market revenues for the years from 2011 to 2017.

Another study [12] by the International Data Corporation (IDC) about big data storage forecast for the period from 2014 to 2019 found that hardware platforms and systems are estimated to grow at a Compound Annual Growth Rate (CAGR) of 20.4% from 2014 to 2019.

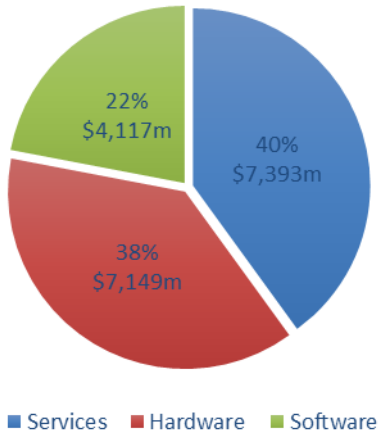

Figure 1. Big data revenue by domain in 2013 in US$ million.

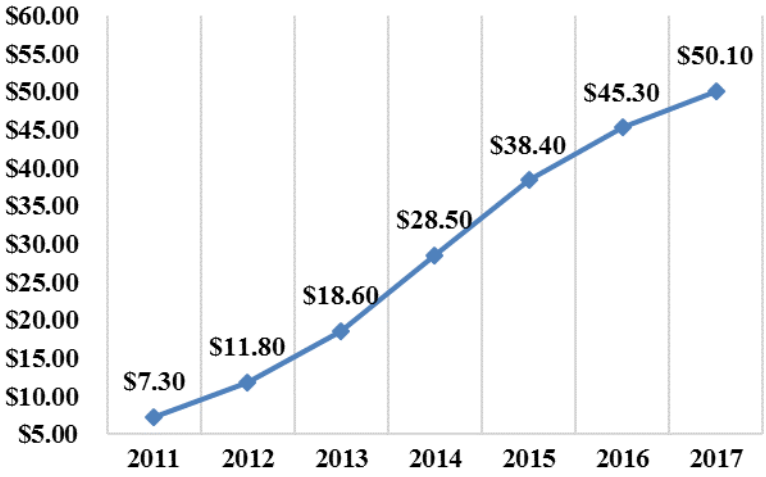

Figure 2. Big data market forecast between 2011 and 2017 in US$ billion.

In healthcare domain, the advancement in information technology and the capability of storing more data in the digital age have driven countries and governmental organizations to computerize health records and produced what we call Electronic Health Record (EHR) or Electronic Medical Record (EMR). EHR/EMR is the electronic form of a patient's medical history that is equivalent to the traditional hard copy of the patient's medical record. EHR system enables patient's records to be shared across healthcare providers in a certain state or globally. EHR records may include a range of data, including general medical records, family medical history, patient examinations, patient treatments, allergies, immunization status, laboratory results and radiology images [13]. The adoption of EHR in many countries has encouraged healthcare providers to store patients' information in an electronic form which produced a large amount of structured and unstructured data. For instance, reports about the United States of America stated that U.S EHR reached 150 exabytes by the end of 2011 and may soon reach the zettabyte and yottabyte era [14]. Another report [15] analyzed the future of EHR market. In terms of geographical analysis, the report determined the market trends for five regions (Europe, Asia and the Pacific, North America, Latin America and the rest of the world). The report concluded that EHR market was valued at US$ 18.9321 billion in 2014 and predicted a compound annual growth rate of 5.4% for the period from 2014 to 2023 that may reach the peak of US$ 30.280 billion by 2023.

2. BIG DATA ASPECTS IN HEALTHCARE

The literature shows that the initial aspect for big data was introduced by Doug Laney in 2001 [1]. Laney argued that e-business and e-commerce system transactions are generating a high volume of information, rapid growth of data (velocity) and different types of data from several domains (variety). In addition, enterprises started to consider data as an asset they became keener to store data for future use. Volume, velocity and variety formed the basic aspects of big data called the 3Vs. Figure 3 shows the 3Vs big data model.

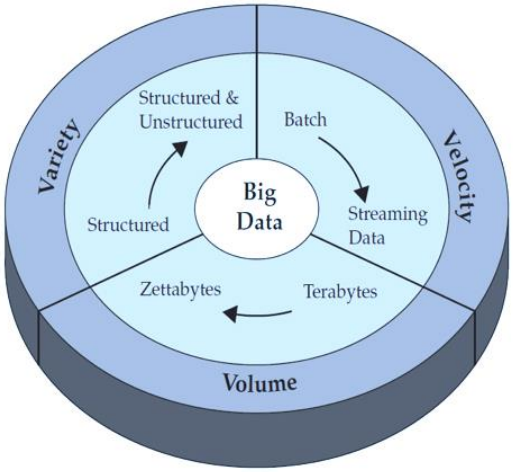

Figure 3. IBM big data model characterized by volume, velocity and variety [16].

Volume refers to dimension, quantity or magnitude of data. This aspect of big data is changing with time. For example, one terabyte of data could be a large amount of data today. However, due to rapid development of computer storage, this couldn't be considered as large data in the future. IBM conducted a survey to determine how big the database should be to be considered as big data. It was found that one terabyte dataset can be considered as big [17], but one terabyte of data may not be considered as big data in the future as the world is shifting to zettabyte era. Variety refers to data diversity from different sources, where data can be structured or unstructured. Velocity refers to data transfer in terms of speed and contents change [18].

In healthcare, big data aspects can be categorized into 4Vs: volume, variety, velocity and veracity [19], as shown in Figure 4. Volume, variety and velocity are identical to the general aspect of big data that shown in Figure 3. Though, an additional aspect has been added, which is veracity. Due to large volume of data and variety of sources, healthcare data varies in quality and complexity. Usually, healthcare data contains biases, missing feature values and noise, which could affect the decision-making process. Additionally, reliable data could reduce the cost of data processing [19]. Mainly, there are two types of data quality problems. The first type is attributed to technical matters, whereas the second type is the truthfulness of data and data sources [20].

Big data problem solving procedure consists of several phases. First comes the process of capturing and collecting data from one source or from different sources. Healthcare data volume is huge and derived from different sources, such as hospitals, medical centres, pharmacies, pathologies …etc. The process of capturing healthcare data can generate some challenges, such as data characteristics, heterogeneity, storage capacity, storage medium, cloud storage, compression tools and plans for backup and disaster recovery. Thus, IT industries, such as Oracle, have developed solutions to handle a very high transaction volume in a distributed environment and provide support to the research community in the field [21]. Similarly, HDFS (Hadoop File System) can handle a huge volume of data across multiple machines or distributed

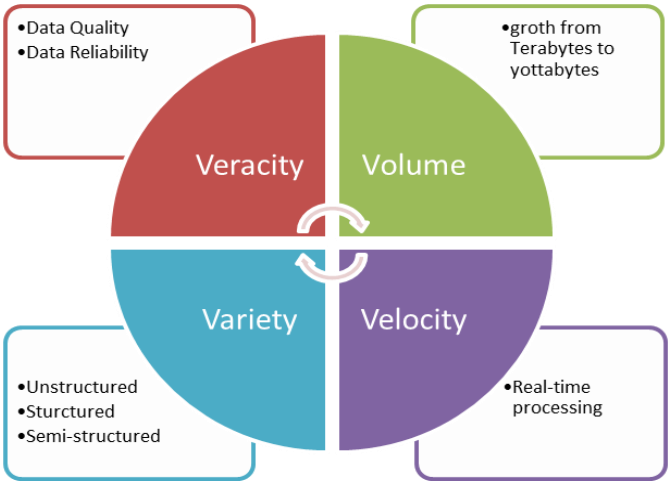

Figure 4. Big data in healthcare (4Vs).

systems [22]. Second comes the process of organizing data. In some cases, data need to be transformed from its original state into a new state to prepare data for analysis using big data solutions. In healthcare, migrating data from legacy database management systems to Hadoop system, for example, require to reformat data into a more beneficial structure, such as hierarchical structure [23]. The third phase is data integration from different sources. In healthcare, Hadoop and NoSQL solutions support data integration from data warehouses, hospitals and social platforms like Facebook …etc. [24]. Fourthly, the purposes of analysis should be identified. For example these purposes could be, obtaining a valuable insight from huge data, predicting healthcare fraud, identifying a useful pattern, predicting patients' behaviour, detecting diseases at earlier stages or tracking disease outbreaks and transmission …etc. [25]. Big data can be analyzed with advanced software tools used in the analytic field, such as online analytical processing, data mining, statistical analysis, machine learning …etc. Finally, decision makers can act based on the findings of the analysis process. Figure 5 shows the cycle for big data management [26].

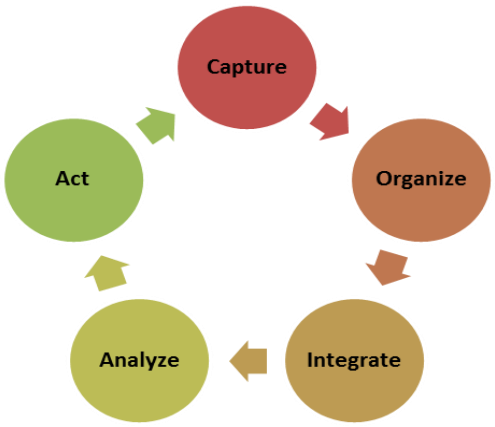

Figure 5. The cycle for big data management.

3. BIG DATA IN HEALTHCARE

The main objective of big data is to gain knowledge from a high volume of unstructured data that comes from numerous sources. Health data and computational techniques can be used to answer clinical questions. For instance, flu-prediction project by the Institute of Cognitive Sciences at Osnabrück University is combining social media data, e.g. Twitter, with CDC data

(Centre for Disease Control and Prevention in the USA). Flu-prediction project [27] gives the users the opportunity to ask questions about flu-related subjects and provides answers in seconds. However, the precision of flu-prediction project answers needs more enhancement and development. Besides, combining social media data with governmental health data will possibly create some challenges, because governmental health data is considered as more accurate than subjective opinions on social media.

Another attempt on big data in healthcare is Google-Flu Trends. Google Flu Trends (GFT) is a web service operated by Google. The main aim is to provide some information on influenza activity for selected countries (more than 25 countries). Google Flu Trends collects search queries from users, then it tries to make accurate predictions about flu activity. This project was first launched in 2008 by Google.org to help predict outbreaks of flu [28]. GFT performed well for the first two to three years. Unfortunately, Google Flu Trends started to perform worse since 2011 and missed the peak in year 2013 by 140 percent according to a research team from Northeastern University, the University of Houston and Harvard University [29]. However, GFT failure doesn't reduce the importance and value of big data. Further, it indicated the possibility of using big data in the healthcare sector and opened the door for more future research such as Auto Regression with GOogle search data (ARGO).

A research team from the Department of Statistics at Harvard University investigated the failure of Google Flu Trends and proposed a new model for influenza tracking called ARGO (Auto Regression with GOogle search data) [30]. They found the main causes of Google Flu Trends failure. First comes, the lack of ability to adapt with people behaviour changes online. For example, people usually search for influenza symptoms and diagnostics, but lately, people behaviour shifted to search for news about the influenza season. In addition, different keywords are used for searching. Secondly, Google Flu Trends didn't utilize the new data released by CDC (Centre for Disease Control and Prevention) which may allow them to adjust and enhance the model using more reliable data [31].

ARGO performed better than all previously available Google-search–based tracking models for influenza, as well as the Google Flu Trends. ARGO utilizes the current publicly available data from google-search, data collected from Google Flu Trends and other publicly available data from other vendors, such as CDC. The main features of ARGO as described by the research team [30] were the ability to integrate influenza data with other data related to epidemics and monitor and record the changes of people search online behaviour. Most important, ARGO can be used for real-time tracking of other social events and diseases. Despite the decent features of ARGO and its performance against previous models, AGRO is not guaranteed to work for ever or to success next year for example, because of public data and people search behaviour shift. Furthermore, any changes to the inner-works of the search engine or any changes in the way information is displayed to users will affect the accuracy of ARGO.

From Google Flu Trends GFT and ARGO, we may comprehend the significance of the work accomplished by GFT and the importance of the idea. Despite the failure of GFT, it draws a road map for others to investigate problems and find solutions and that what AGRO team has been doing. Further, we may expect other research teams to focus on all versions of flu-season-trends to find a better version that may in future become a product and to emphasize the importance of big data in healthcare and other sectors. A study on big data analysis [32] confirmed the importance of big data in modelling disease spread and real-time identification of emergencies. Another study [33] demonstrated that big data analysis can be used to discover disease patterns and record disease outbreaks over the world, especially when a swift information is needed. The study added that public health issues can be improved with the analytical approach, where a large amount of data can help determine the needs, offer required services and predict and prevent the future crises to benefit the people. Further, detecting fraud in healthcare becomes more efficiently. Not to forget is the most important role of big data analysis in healthcare, which is to enhance the quality of care and services for patients. In

addition, big data can help with knowledge distribution across healthcare providers in one country or across the world. Some countries have a lack in medical expertise. So, knowledge and guidance driven by big data may provide useful and rapid information for practitioners in regional and third world countries. For example, Swine Flu (H1N1) was first reported in April 2009 infecting millions of people and estimated deaths worldwide, due to H1N1, were about 18500 people according to WHO report [34]. When a certain country or organization has huge volume of health data about the generic influenza viruses, then big data tools could provide swift knowledge, such as initial diagnosis and treatment procedures. These outcomes and knowledge can be distributed and shared across healthcare providers worldwide to save lives and reduce the impacts before it is too late.

4. BIG DATA TOOLS IN HEALTHCARE

Mike Cafarella and Doug Cutting can be considered as the founders of Hadoop, which is an evolution from the open source web search engine (Apache Nutch). Apache Nutch creators understood the limitation of nutch and the difficulty to reach the very huge number of webpages in the internet. At the meantime, Google published a research paper about its own storage system architecture called Google's Distributed File System (GFS) which solved nutch storage needs and drew a map to implement Nutch Distributed File System (NDFS). It also opened the space for other research issues and development opportunities. Google also introduced MapReduce in another research paper and nutch team implemented MapReduce to nutch which later ran on MapReduce and NDFS. In 2006, nutch developers formed an independent project based on MapReduce and NDFS, called Hadoop, which is the elephant toy name for Doug Cutting's son [35].

Hadoop is an open source infrastructure software or a framework to store and process a huge dataset. It is an open source project under apache. Basically, Hadoop performs two fundamental operations; storing data and processing data. Hadoop stores information by using Hadoop Distributed File System (HDFS). HDFS uses the cluster architecture to store data across different machines (nodes), such as PCs and servers, which gives the ability to store a huge volume of data on thousands of nodes similar to what Yahoo is doing nowadays. The second operation that Hadoop preforms is data processing. Data processing is accomplished by another component called MapReduce. Processing data includes counting keywords in the dataset, aggregating data and searching in the data. The traditional architecture of processing data stored on different clusters was to move the data from the cluster to the software. This operation may cause bandwidth problems and consume a long time, especially with big data. However, MapReduce performs opposite to traditional architecture by moving the software process to the node (map) instead of moving the data to the software, and then collecting the answers or the process output (Reduce). This is where the name MapReduce comes from. Hadoop ecosystem contains different sub-projects to help give Hadoop more flexibility and ease of use, such as Pig and Hive [22], [36].

Pig is a high-level programming language for expressing a data analysis program that is designed to ease and utilize Hadoop programming and tasks for users with little programming skills, without having to know MapReduce coding to achieve a certain task. Further, Pig is an extendable language, where users can define their own functions. In addition, tasks in Pig are encoded to permit the system to optimize task execution automatically, which gives the users more room to focus on semantics rather than on efficiency [37]. Hive is a data warehouse infrastructure used for several operations on distributed storage and managing of large data using HiveQL (pretty similar to Structure Query Language SQL). Hive was developed by Facebook, then Apache Software Foundation used the Facebook version and enhanced it to form an open source tool under the name Apache Hive. The main idea behind Hive is to solve the complexity of MapReduce programming; therefore, instead of writing MapReduce program in Java, users are able to code a query for MapReduce job and process it to produce the same

result as in MapReduce [38]. HBase is a part of Hadoop ecosystem that runs on the top of Hadoop Distributed File System (HDFS). The main reason for developing HBase is that HDFS read/write of data is a sequential matter, where HBase allows random real-time access and stores data in a similar way as in the conventional RDBMS (columns and rows), where HDFS stores data as a collection of files. In addition, HBase provides faster access and operation on a huge volume of data. HBase has its own Java client API and tables in HBase can be used both as input/output for MapReduce tasks [36]. Figure 6 shows a common Hadoop ecosystem.

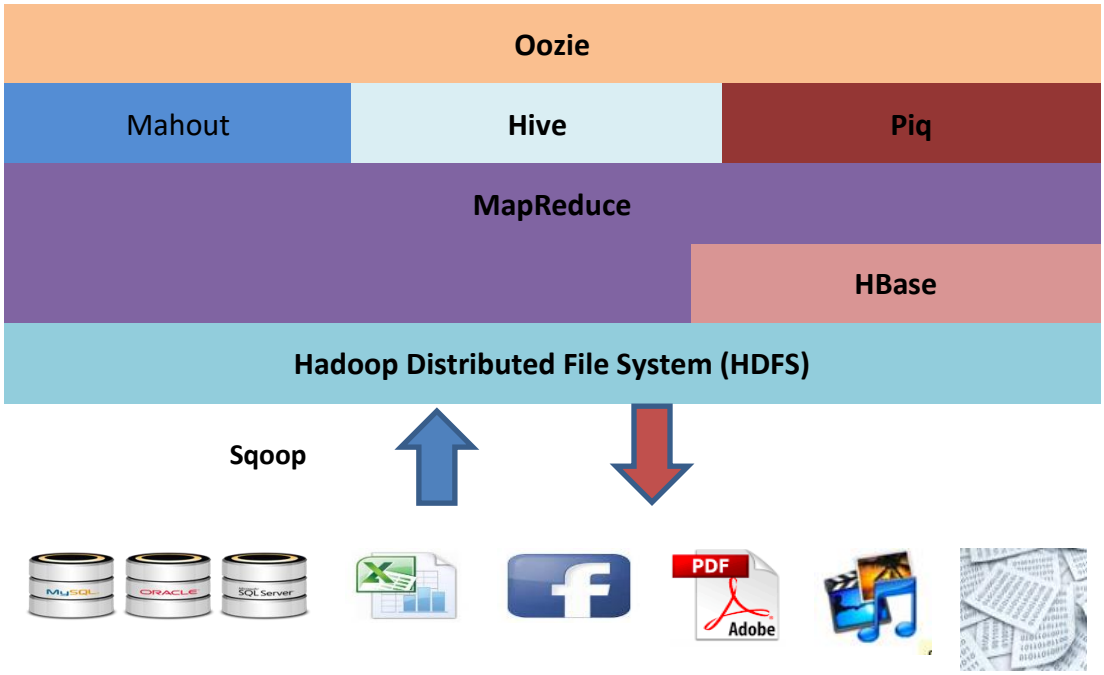

Figure 6. Hadoop ecosystem.

In big data analytics, data loading into clusters and processing occur in parallel with other data that resides on one location or is distributed over different locations. Loading a huge volume of data into Hadoop can be a difficult task, where accessing data directly by mapReduce may cause some complications, where Sqoop can facilitate the process of importing and exporting data, generate Java classes to allow interaction with data and allow users to import data directly into Hive data warehouse [39]-[40]. Mahout is an implementation of scalable machine learning algorithms by Apache Software Foundation. Mahout would be a good choice when data is distributed on several machines and when data volume is huge. In machine learning applications, larger data size could produce better accuracy. As the number of training data increases, Mahout's performance also increases [41].

Hadoop is considered as the best technology for big data in healthcare industry because of its ability to handle a huge volume of data in different formats, leaving no data behind, as stated by Charles Boicey; an information solutions architect at the University of California Irvine. Hadoop has been used widely in healthcare industry for cancer treatment and genomics, monitoring patients, clinical support network, healthcare intelligence, fraud prevention and detection …etc. [42].

The main reason why cancer has not been cured so far is that cancer transforms into diverse patterns based on every individual's genomics, and there are over three billion base pairs that constitute human DNA. In the future, big data tools such as Hadoop could provide the opportunity to store and map the three billion base pairs for every cancer patient. In monitoring

patients, Hadoop technology has been successfully used to monitor about 6200 children patients in Children's Healthcare of Atlanta (CHOA) in the United States. CHOA used sensors beside each child to track different signs, such as blood pressure, heartbeat and respiratory rate. Sensors collect a huge volume of data which is beyond legacy systems, then use the collected data successfully to produce patterns that alert medical staff to provide medical attention to patients when it is needed. Explorys is a company holding the largest database in healthcare. Explorys used Hadoop technology to analyze the huge volume of data and produced an analytical tool which assists medical staff to provide the most suitable treatment for individual patients or patient populations [42].

5. BIG DATA RESEARCH CHALLENGES IN HEALTHCARE

Big data can bring huge benefits to individuals, organizations, countries and to the world. However, benefits can bring risks as well, such as the lack of privacy and security. Further, many big data tools are open source and free to use tools, which may cause back-doors for intrusions, hackers and data theft. Hence, confidentiality, security, integrity and availability should be measured.

Privacy and Security: The first concern when working with big data is privacy and security. Privacy and security are a key concern for individuals and organizations that hold information/data about people, products, transactions …etc. Health data obtained by healthcare providers and medical practitioners from individuals' may contain private and confidential data. Individuals data must be handled with enormous care to protect people's privacy and confidentiality. Some approaches used to enhance the security level and obtain some confidentiality are [43]: First, individual identification is deleted during data collection (anonymous data). Second, individual identification is recorded initially during data collection and then removed. In this type of identification, there is a chance to re-identify the patient because patient information has been recorded at some stage (anonymized data). However, the removal of personal health data requires the removal of data elements in Table 1. Removing these data to meet the de-identification act [44] can affect the outcome of data analysis [45]. Third, encoding and encrypting data, however, give some chances to identify the encryption key using the advancement of computer technology which still exists. Privacy advocates and data regulators are gradually complaining about data collection and data usage in big data era, calling for a sophisticated protocol that achieves balance between individual privacy and research benefits [46].

Data Ownership: Data ownership describes a critical and ongoing challenge in big data applications in healthcare and other fields. Though petabytes of health data reside in healthcare providers' premises and governmental healthcare systems, these are not really owned by them. On the other hand, patients believe that they own the data. This controversy maybe ended in the legal system to resolve the ownership issues unless healthcare providers receive a written consent from patients prior to utilizing data for commercial or research purposes [47]-[48].

Clinical Data Linkage (CDL): CDL defines one of big data issues, where there are two or more tuples or instances relating to the same individual or entity. CDL occurs in the healthcare data when one individual or entity is stored on multiple healthcare providers such as two different hospitals. In addition, CDL can be quite challenging if the infrastructure of healthcare system is heterogeneous. In Australia, for example, clinical data linkage amongst patients with Spinal Cord Injury (SCI) is a challenge, because the Australian healthcare system is distributed over different states, and there is lack of coordination between healthcare providers in the states and in the federation. Hence, duplication exists and patient information that exists on many different sources is inconsistently updated [49].

Data Characteristics: Big data are usually collected from different sources and in different formats, which make them heterogeneous data. This may cause a limited value for big data,

since data may be incomplete, poorly described, improperly collected or outdated [50]. Further, the main purpose of data collection in healthcare systems is not for data analytics. Usually, healthcare data analytics is a secondary purpose of data collection. For instance, patients' data is collected primarily for payment, as well as tracking patient progress, treatment and clinical status. When the data is used for knowledge discovery, it may compromise the reliability and validity of any resulting models, because data has been collected for a different purpose. Hence, this creates a challenge for big data analysis and needs more effort to ensure data reliability [45].

Storage and Processing Issues: Doubtlessly, there is a significant advancement in computer storage technology. However, data grow significantly whenever a new storage technology is invented due to the huge amount of data collected/transferred by social media, healthcare providers, business transactions, stock markets …etc. The enormous amount of data created and generated around the globe put great pressure on data processing. The processing issue could be solved by bringing the software to data instead of sending data to the software or by transmitting only data which is important to the analysis process. However, this may create problems on data integrity and data source [47]. Health data movement is another related challenge; for instance, how huge volumes of data sized as giant (petabytes) move from data centres to the cloud. In this case, huge volumes of data are transported physically using physical media, such as FedEx and Amazon Snowball [51], but this could create another challenge when data needs an update. Storage capacity is another challenge in healthcare big data. For instance, one single human genome sequencing requires about two terabytes of storage capacity [52].

Skills Requirement: Big data researchers have been engaged with plenty of studies and researches that describe big data applications and technology development in data storage and analysis. However, there has been little attention to skills required for individuals to works in the big data field. A recent study [53] investigated the required skills to deal with big data and concluded that a big data specialist should be a combination of computer scientist and statistician with significant industry knowledge. Further, experts cite major shortages in big data specialists [51]. Further, educational institutions are responsible for making students aware of the new trend in data analytics and the required skills and technology for the industry.

6. SUMMARY AND CONCLUSION

This paper criticizes some applications of big data in healthcare, such as flu-prediction project by the Institute of Cognitive Sciences, which combines social media data with governmental data to provide swift response about flu-related subjects. Regardless of great efforts made by the Institute of Cognitive Sciences, the project should study human multi-modal representations (verbal communication, emotions, text and images). Moreover, integrating social media data with governmental health data will possibly create some challenges, because governmental health data is considered as more accurate than subjective opinions on social media. Particularly, integration between different authoritative sources of data could enable the composition of two complementary points of view of the same problem, which could affect the outcome of knowledge acquisition. Google Flu Trends GFT collects search queries from users to predict flu activity and outbreak. GFT performed well for the first two to three years; however, it started to perform worse since 2011 due to people behaviour changes. GFT did not update the prediction model based on new data released by the Center for Disease Control and Prevention-US (CDC). On the other hand, ARGO performed better than all previously available influenza models, because it adjusts people behaviour changes and relies on current publicly available data from google-search and CDC.

This research paper also describes, analyzes and reflects the value of big data in healthcare. Big data has been introduced and defined based on the most agreed terms. The paper explains big data revenue forecast for the year 2017 and historical revenue in three main domains: services, hardware and software. Big data management cycle has been reviewed and the main aspects of

big data in healthcare (volume, velocity, variety and veracity) have been discussed. Finally, a discussion has been made of some challenges that face individuals and organizations in the process of utilizing big data in healthcare, such as data ownership, privacy and security, clinical data linkage, storage and processing issues and skills requirement. The paper found that each issue mentioned in the current work requires further research because of their importance, where this will be the future research, especially on privacy and security issue. The paper also concludes that the world is moving toward data-driven approach which relies on data to perform business decisions. It is also important to understand that failure can be the first step toward success. Hence, the failure of Google Flue Trends might be the first step toward a successful model to utilize big data successfully and efficiently in healthcare.

Table 1. De-identification of protected health information in accordance with the Health Insurance Portability and Accountability Act (HIPAA) privacy rule, Washington, DC: Department of Health and Human Services [cited January 16, 2017].

(A) Names
(B) All geographic subdivisions smaller than a state, including street address, city, county, precinct, ZIP code and their equivalent geocodes, except for the initial three digits of the ZIP code if, according to the current publicly available data from the Bureau of the Census: (1) The geographic unit formed by combining all ZIP codes with the same three initial digits contains more than 20,000 people; and (2) The initial three digits of a ZIP code for all such geographic units containing 20,000 or fewer people is changed to 000
(C) All elements of dates (except year) for dates that are directly related to an individual, including birth date, admission date, discharge date, death date, and all ages over 89 and all elements of dates (including year) indicative of such age, except that such ages and elements may be aggregated into a single category of age 90 or older
(D) Telephone numbers
(L) Vehicle identifiers and serial numbers, including license plate numbers
(E) Fax numbers
(M) Device identifiers and serial numbers
(F) Email addresses
(N) Web Universal Resource Locators (URLs)
(G) Social security numbers
(O) Internet Protocol (IP) addresses
(H) Medical record numbers
(P) Biometric identifiers, including finger and voice prints
(I) Health plan beneficiary numbers
(Q) Full-face photographs and any comparable images
(J) Account numbers
(R) Any other unique identifying number, characteristic or code, except as permitted by paragraph (c) of this section; and
(K) Certificate/license numbers
(ii) The covered entity does not have actual knowledge that the information could be used alone or in combination with other information to identify an individual who is subject of the information.

REFERENCES

[1] R. Jayanthi, "Big Data Applications in Healthcare," in: Impact of Emerging Digital Technologies on Leadership in Global Business, USA: IGI Global, p. 202, 2016.

[2] K. W. Oh, P. Lee and Y. W. Choi, "Enhanced Unlatch Operation of Disk Drive for Low Temperature Environment," Procedia Eng., vol. 131, pp. 906–913, 2015.

[3] A. Li, "Presidential Debate Most-Tweeted Event in U.S. Political History," 2012, [Online], Available at: http://mashable.com/2012/10/04/presidential-debate-twitter/#p54DATw_Wuqz.

[4] A. De Mauro, M. Greco and M. Grimaldi, "What is Big Data? A Consensual Definition and a Review of Key Research Topics," AIP Conference Proceedings, vol. 1644, pp. 97–104, 2015.

[5] E. Dumbill, "Making sense of big data," Big Data, vol. 1, no. 1, pp. 1–2, 2013.

[6] D. Fisher, R. De Line, M. Czerwinski and S. Drucker, "Interactions With Big Data Analytics," Interactions, vol. 19, no. 3, pp. 50–59, 2012.

[7] C. Wu, R. Buyya and K. Ramamohanarao, "Big Data Analytics = Machine Learning + Cloud Computing," p. 27, Jan. 2016.

[8] N. Council, Frontiers in Massive Data Analysis, The National Academies Press Washington, DC, 2013.

[9] D. Boyd and K. Crawford, "Critical Questions for Big Data: Provocations for a Cultural, Technological and Scholarly Phenomenon," Information, Commun. Soc., vol. 15, no. 5, pp. 662–679, 2012.

[10] L. Gomes, "Machine-learning Maestro Michael Jordan on the Delusions of Big Data and other Huge Engineering Efforts," IEEE Spectrum, vol. 20, Oct. 2014.

[11] J. Kelly, "Big Data Vendor Revenue and Market Forecast," Wikibon Artic. Febrero, 2014.

[12] Ashish Nadkarni, Iris Feng and Laura DuBois, "Worldwide Storage in Big Data Forecast, 2015–2019," IDC, Market Forecast, Doc # 259205, Oct. 2015.

[13] M. A. B. Ahmad, Mining Health Data for Breast Cancer Diagnosis Using Machine Learning, PHD Thesis, University of Canberra, Australia, Dec. 2013.

[14] W. Raghupathi and V. Raghupathi, "Big Data Analytics in Healthcare: Promise and Potential," Heal. Inf. Sci. Syst., vol. 2, no. 1, p. 3, 2014.

[15] Transparency Market Research, "Electronic Health Records Solution Market (Web Based, Client Server Based, Software as Services) for Applications in Hospitals, Physicians Office, Ambulatory Centers - Global Industry Analysis, Size, Share, Growth, Trends, and Forecast 2015 - 2023," MarketersMedia, USA, 2016.

[16] P. Zikopoulos and C. Eaton, Understanding big data: Analytics for Enterprise Class Hadoop and Streaming Data, McGraw-Hill Osborne Media, 2011.

[17] A. Gandomi and M. Haider, "Beyond the Hype: Big Data Concepts, Methods and Analytics," Int. J. Inf. Manage., vol. 35, no. 2, pp. 137–144, 2015.

[18] I. A. T. Hashem, I. Yaqoob, N. B. Anuar, S. Mokhtar, A. Gani and S. U. Khan, "The Rise of 'Big Data' on Cloud Computing: Review and Open Research Issues," Inf. Syst., vol. 47, pp. 98–115, 2015.

[19] R. Fang, S. Pouyanfar, Y. Yang, S.-C. Chen and S. Iyengar, "Computational Health Informatics in the Big Data Age: A Survey," ACM Comput. Surv., vol. 49, no. 1, pp. 1–36, 2016.

[20] B. Feldman, E. M. Martin and T. Skotnes, "Big Data in Healthcare - Hype and Hope," Dr. Bonnie 360 Degree (Bus. Dev. Digit. Heal., vol. 2013, no. 1, pp. 122–125, 2012.

[21] Oracle, "Oracle Big Data Products," [Online], Available at: https://www.oracle.com/big-data/products.html.

[22] Hadoop, "Apache Hadoop," [Online], Available at: https://wiki.apache.org/hadoop.

[23] D. Miner and A. Shook, MapReduce Design Patterns: Building Effective Algorithms and Analytics for Hadoop and Other Systems, O'Reilly Media, Inc., 2012.

[24] N. Yuhanna and T. F. Wave™, "Market Overview: Big Data Integration," Forrester

Research, Inc. Reproduction Prohibited, 2014.

[25] A. Trnka, "Big Data Analysis," Eur. J. Sci. Theol., vol. 10, no. 1, pp. 143–148, 2014.

[26] J. Hurwitz, A. Nugent, F. Halper and M. Kaufman, Big data for dummies, John Wiley & Sons, 2013.

[27] Osnabrück University and IBM WATSON, "Flu-prediction Project," 2016, [Online], Available at: http://www.flu-prediction.com.

[28] Google, "Google Flu Trends," 2016, [Online], Available at: http://www.google.org/flutrends/about/

[29] D. Lazer and R. Kennedy, "What We Can Learn From The Epic Failure of Google Flu Trends," in: Wired, 2015.

[30] S. Yang, M. Santillana, and S. C. Kou, “Accurate estimation of influenza epidemics using Google search data via ARGO,” Proc. Natl. Acad. Sci., vol. 112, no. 47, pp. 14473–14478, Nov. 2015.

[31] B. Mole, "New Flu Tracker Uses Google Search Data Better Than Google," Scientific Method, USA, 2015.

[32] D. Lazer, R. Kennedy, G. King and A. Vespignani, "The Parable of Google Flu: Traps in Big Data Analysis," Science (80-.)., vol. 343, no. 6176, pp. 1203–1205, 14 March 2014.

[33] Builtinla, "Significant Benefits of Big Data Analytics In Healthcare Industry," Builtinla, 2016, [Online], Availavle at: http://www.builtinla.com/blog/significant-benefits-big-data-analytics-healthcare-industry.

[34] WHO, "Report of the Review Committee on the Functioning of the International Health Regulations (2005) in relation to Pandemic (H1N1) 2009," World Health Orgnaization, 2011.

[35] T. White, Hadoop: The definitive guide, O’Reilly Media, Inc., 2012.

[36] R. C. Taylor, "An Overview of the Hadoop/MapReduce/HBase Framework and Its Current Applications in Bioinformatics," BMC Bioinformatics, vol. 11, no. Suppl. 12, p. S1, 2010.

[37] The Apache Software Foundation, "Apache Pig!," 2016, [Online], Available at: https://pig.apache.org/. [Accessed: 05-Aug-2016].

[38] The Apache Software Foundation, "Apache Hive," 2016, [Online], Available at: https://hive.apache.org/. [Accessed: 05-Aug.-2016].

[39] "Apache Sqoop - Overview : Apache Sqoop," [Online], Available at: https://blogs.apache.org/sqoop/entry/apache_sqoop_overview. [Accessed: 24-Dec.-2016].

[40] "Introducing Sqoop - Cloudera Engineering Blog," [Online], Available at: http://blog.cloudera.com/blog/2009/06/introducing-sqoop/, Accessed: 24-Dec.-2016.

[41] A. Gupta, Learning Apache Mahout Classification, Packt Publishing Ltd., 2015.

[42] "5 Healthcare applications of Hadoop and Big data," DeZyre, 16 March 2015.

[43] K. J. Cios and G. William Moore, "Uniqueness of Medical Data Mining," Artif. Intell. Med., vol. 26, no. 1, pp. 1–24, 2002.

[44] "Methods for De-identification of PHI | HHS.gov," Health Information Privacy, [Online], Available at: https://www.hhs.gov/hipaa/for-professionals/privacy/special-topics/de-dentification/index.html#guidancedetermination, Accessed: 16 Jan. 2017.

[45] S. White, "A Review of Big Data in Health Care: Challenges and Opportunities," Open Access Bioinformatics, vol. 6, pp. 13–18, 2014.

[46] O. Tene and J. Polonetsky, "Privacy in The Age of Big Data: A Time for Big Decisions,"

Stanford Law Review, 2012, [Online], Available at: https://www.stanfordlawreview.org /online/privacy-paradox-privacy-and-big-data/.

[47] S. Kaisler, F. Armour, J. A. Espinosa and W. Money, "Big Data: Issues and Challenges Moving Forward," The 46th Hawaii International Conference on System Sciences (HICSS), pp. 995–1004, 2013.

[48] S. Kaisler, W. H. Money and S. J. Cohen, "A Decision Framework for Cloud Computing," The 45th Hawaii International Conference on System Sciences, pp. 1553–1562, 2012.

[49] J. Moon *et al.*, "Clinical Data Linkages in Spinal Cord Injuries (SCI) in Australia:," in: Big Data Analytics in Bioinformatics and Healthcare, vol. 43, no. 10, IGI Global, 1AD, pp. 392–405.

[50] A. W. Toga and I. D. Dinov, "Sharing Big Biomedical Data," J. Big Data, vol. 2, no. 1, p. 7, Dec. 2015.

[51] P. Moghe, "6 Hidden Challenges of Using the Cloud for Big Data and How to Overcome Them," Insider, 2016, [Online], Available at: http://thenextweb.com/insider/2016/04/12/6-challenges-cloud-overcome/.

[52] S. Robinson, "The Storage and Transfer Challenges of Big Data," MIT Sloan Management Review, 2012, [Online], Available at: http://sloanreview.mit.edu/article/the-storage-and-transfer-challenges-of-big-data/. [Accessed: 16-Jan-2017].

[53] S. Debortoli, O. Müller and J. vom Brocke, "Comparing Business Intelligence and Big Data Skills," Bus. Inf. Syst. Eng., vol. 6, no. 5, pp. 289–300, Oct. 2014.

ملخص البحث:

يتولّد في كوكبنا حجم هائل من البيانات في شتى المجالات، مثل: وسائل التواصل الاجتماعي، والصناعات، وأسواق المال، وأنظمة الرعاية الصحية. ويمكن لهذا الكم الهائل من البيانات أن يجلب الفوائد والمعرفة للأفراد والحكومات والصناعات وأن يساعد في اتخاذ القرارات. في مجال الرعاية الصحية، يتم توليد حجم هائل من البيانات من مقدمي الرعاية الصحية وتخزينها في أنظمة رقمية. وبذلك تكون البيانات أكثر قابلية للوصول من أجل الرجوع إليها أو استخدامها مستقبلاً. وتتلخص الرؤية النهائية لاستخدام البيانات الضخمة في مجال الرعاية الصحية في دعم عملية تحسين الخدمة المقدمة من موفّري الرعاية الصحية، والتقليل من الأخطاء الطبية، وتقديم المشورة عند الحاجة. تقدم هذه الورقة مراجعة نقدية لبعض تطبيقات البيانات الضخمة في الرعاية الصحية؛ مثل مشروع توقُّع الزكام الذي تبنّاه معهد العلوم الإدراكية. من جهةٍ أخرى، تصف هذه الورقة قيمة البيانات الضخمة في مجال الرعاية الصحية وتحللها وتعكسها. وفيها تم تعريف البيانات الضخمة بناءً على المصطلحات المتفق عليها لدى الباحثين والمهتمين في هذا المجال. كذلك، تبين هذه الورقة التوقعات المتعلقة بالعائد الاقتصادي للبيانات الضخمة لعام ٢٠١٧، إلى جانب عائداتها التاريخية في ثلاثة مجالات أساسية هي: الخدمات، والمعدّات، والبرمجيات. وقد تم استعراض دورة إدارة البيانات الضخمة ومناقشة الجوانب الأساسية للبيانات الضخمة في مجال الرعاية الصحية (الحجم، والسرعة، والتنوع، والصدق). وفي الختام، تم إجراء مناقشة لبعض التحديات التي تواجه الأفراد والمنظمات في عملية الاستفادة من البيانات الضخمة في الرعاية الصحية؛ مثل: ملكية البيانات، وخصوصيتها وأمنها، والربط السريري للبيانات، والمسائل المتعلقة بتخزين البيانات ومعالجتها، والمتطلبات التي يجب توافرها بشأن مهارات العاملين في هذا المجال.

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 3, No. 1, April 2017.

BAT Q-LEARNING ALGORITHM

Bilal H. Abed-alguni

(Received: 30-Nov.-2016, Revised: 01-Feb.-2017, Accepted: 23-Feb.-2017)

ABSTRACT

Cooperative Q-learning approach allows multiple learners to learn independently and then share their Q-values among each other using a Q-value sharing strategy. A main problem with this approach is that the solutions of the learners may not converge to optimality, because the optimal Q-values may not be found. Another problem is that some cooperative algorithms perform very well with single-task problems, but quite poorly with multi-task problems. This paper proposes a new cooperative Q-learning algorithm called the Bat Q-learning algorithm (BQ-learning) that implements a Q-value sharing strategy based on the Bat algorithm. The Bat algorithm is a powerful optimization algorithm that increases the possibility of finding the optimal Q-values by balancing between the exploration and exploitation of actions by tuning the parameters of the algorithm. The BQ-learning algorithm was tested using two problems: the shortest path problem (single-task problem) and the taxi problem (multi-task problem). The experimental results suggest that BQ-learning performs better than single-agent Q-learning and some well-known cooperative Q-learning algorithms.

KEYWORDS

Q-learning, Bat algorithm, Optimization, Cooperative reinforcement learning.

1. INTRODUCTION

Q-learning is a well known reinforcement learning (RL) algorithm that allows machines and software agents to develop an ideal behavior within a specific environment based on trial and error [1]-[3]. A Q-learning agent learns how to behave by trying actions to determine how to maximize some reward. This is usually accomplished using temporal difference learning to find mapping from state-action pairs into quality values (Q-values). A Q-value of a state-action pair (s, a) represents the expected utility of taking action a in state s and following a fixed policy thereafter. The Q-values are normally calculated using a utility function known as a Q-function. These values are usually stored in a data structure known as a Q-table.

Cooperation among several reinforcement learners in the same multi-agent environment provides an opportunity for the learners to cooperatively solve a learning problem. Such an approach to RL, which is called cooperative RL, is increasingly used by research labs around the world to solve real world problems, such as robot control and autonomous navigation [4], [5]. This is because cooperative reinforcement learners can learn and converge faster than independent reinforcement learners via sharing of information (e.g., Q-values, Episodes, Policies) [3], [6]-[8]. One such example is cooperative Q-learning, in which several learners share their Q-values among each other in order to accelerate their convergence to optimal solutions [9], [10]. Cooperative Q-learning is normally broken into two stages. The first stage is known as the independent learning stage, in which each reinforcement learner individually applies Q-learning to enhance its own solution. In the second stage, the learning by interaction stage, the learners share their Q-values based on a sharing strategy. A Q-value sharing strategy defines how the independent learners can share their Q-values among each other to obtain new Q-tables. This strategy can only be applied when the agents have Q-tables with a similar structure.

B. Abed-alguni is with the Computer Science Department, Yarmouk University, Irbid, Jordan.
E-mail: Bilal.h@yu.edu.jo.

Current cooperative Q-learning algorithms, such as AVE-Q, BEST-Q, PSO-Q [6], [11]-[15] and WSS [7], [16]-[19], may not find the optimal Q-values for different reasons (Section 3). As a result, the policies of the learners might not converge to optimality. In addition, some cooperative Q-learning algorithms perform well with single-task problems, but very poorly with multi-task problems [9]. This issue causes uncertainty about the benefit of choosing one cooperative algorithm over the other cooperative algorithms.

The bat algorithm (BA) is a metaheuristic method that can be used to solve optimization problems by simulating the echolocation behavior of bats [20]. An advantage of BA is that it tries to balance between exploration and exploitation of actions by using tuning techniques that control its parameters (frequencies, pulse emission rates and loudness of the potential solutions) [20]. Consequently, the possibility of finding the optimal solutions increases. Therefore, in order to solve the problems of current cooperative Q-learning algorithms, this paper presents a new cooperative Q-learning algorithm called the Bat Q-learning algorithm (BQ-learning) that is based on the BA algorithm. BQ-learning is distinguished from the other cooperative algorithms by the use of a Q-value sharing strategy based on the BA algorithm. This paper argues that the proposed BQ-learning algorithm increases the possibility of finding the optimal Q-values for different types of learning problems.

The remainder of the paper is organized as follows: Section 2 presents background information, Section 3 discusses related work, Section 4 discusses the BQ-learning algorithm, Section 5 discusses simulation results using the shortest path problem and the taxi domain problem and Section 6 presents the conclusion and future work of this paper.

2. BACKGROUND INFORMATION

This section briefly summarizes some of the underlying concepts of Q-learning and Bat algorithms.

2.1 Q-learning

The problem model of Q-learning is commonly represented as a Markov Decision Process (MDP) [1]. An MDP comprises a set of states $S = \{s_0, s_1, \ldots, s_n\}$, a set of actions $A = \{a_0, a_1, \ldots, a_m\}$, a reward function $R : S \times A \rightarrow \mathbb{R}$ and a transition model $T : S \times A \times S \rightarrow [0,1]$. As specified by the transition model, all the transition probabilities are deterministic, meaning that they can only equal 1 or 0. For example, a transition probability $T(s_x, a_z, s_y) = 1$ means that transitioning from state s_x to state s_y upon executing action a_z is possible. On the other hand, a transition probability $T(s_x, a_z, s_y) = 0$ indicates that the transition is invalid. The immediate expected reward for executing this transition is the deterministic reward $R(s_x, a_z)$ [3]. It is important to note that the implementation of Q-learning to stochastic MDPs is beyond the scope of this paper.

A learner is normally required to apply Q-learning to an MDP for a number of learning episodes in order to learn which action is optimal for each state. A learning episode is the time the agent takes to reach the goal state starting from an initial selected state. Reaching the goal state requires the learner to apply a simple value iteration procedure during each learning episode. This procedure starts when the learner uses its selection policy to select an action a from the set of possible actions A of current state s. The learner then receives a reward $R(s,a)$ and bserves a new state s' of the environment. Subsequently, the agent uses these information to update its Q-table using the following Q-function:

$$Q(s,a) \leftarrow (1-\alpha)Q(s,a) + \alpha\,[R(s,a) + \gamma\ max_{a' \in A} Q(s',a')] \qquad (1)$$

where $s \in S$, $a \in A$, $\alpha \in [0,1]$ is the learning rate and $\gamma \in [0,1]$ is the discount factor. Upon successful convergence to a solution, the output of Q-learning is the optimal Q-function from which an optimal policy $\pi^* : S \rightarrow A$ (i.e., mapping from states to actions that maximizes the total discounted reward ($R = r_0 + \gamma^1 r_1 + \mathrm{K} + \gamma^n r_n$)) can be derived using a greedy selection method.

2.2 Bat Algorithm

Microbats are small bats that usually eat insects. An amazing feature of this species is that they rely on a special type of sonar called echolocation to locate their prey. Microbats make loud sound pulses as they fly. When these pulses hit an object, they produce echoes that return to the ears of the bats. The time required for the sound waves to return back to the microbat is used to calculate the distance of an object.

The Bat algorithm (BA) is a metaheuristic method that is inspired from the echolocation behavior of microbats [20]. This algorithm combines the advantages of existing metaheuristic algorithms, such as particle swarm optimization (PSO) and intensive local search in one algorithm. The research works of Yang and Gandomi [21] and Yang [20] suggest that BA performs better than many existing metaheuristic algorithms, such as PSO, intensive local search, harmony search and genetic algorithm.

The following simplifications of the main characteristics of the echolocation process were followed in order to simulate BA as a problem solver [22]:

- Microbats know the difference between prey and other objects and use echolocation to calculate the distance of their prey.
- Each bat i flies randomly at position x_i with velocity v_i, frequency f_i, varying wavelength λ and loudness A to hunt a prey.
- Loudness varies in the $[A_{min}, A_0]$ interval.
- Each bat i can adjust the frequency f_i and the pulse rate $r_i \in [0,1]$ of its emitted pulse.
- Frequencies of the bats are in the range $[f_{min}, f_{max}]$. These frequencies correspond to wave lengths in the range $[\lambda_{min}, \lambda_{max}]$ that can be calculated as follows:

$$\lambda = \frac{v}{f}; \text{where} \, v = 340 \, m/s \, \text{which is the speed of sound in the air.} \tag{2}$$

Based on the above equation, either λ or f can be used in the BA algorithm, because the relationship between these variables is constant ($v = \lambda f$). The choice between λ and f depends on the type of the problem of interest.

At the beginning of BA (Figure 1), each bat is assigned a random frequency in the range $[f_{min}, f_{max}]$. This range is normally chosen based on the size and complexity of the implemented problem. There are several rules that control the movement of a virtual bat.

The following rules show how a virtual bat i changes its position x_i (solution) and velocity v_i at instant t:

$$f_i = f_{min} + (f_{max} - f_{min})\beta, \tag{3}$$

$$v_i^t = v_i^{t-1} + (x_i^{t-1} - x_*) f_i, \tag{4}$$

1: Objective function $fun(x)$, $x = (x_1,..., x_d)^T$
2: Initialize the bat population $x_i (i = 1, 2, ..., n)$ and v_i
3: Define pulse frequency f_i at x_i
4: Initialize pulse rates r_i to positive values around zero and loudness A_i to positive values around 1.
5: **while** *t< Max number of iterations* **do**
6: Generate new solutions by adjusting frequency and updating velocities and
7: locations/solutions [Equations (3) to (5)].
8: **if** *rand* > r_i **then**
9: Select a solution among the best solutions
10: Generate a local solution around the selected best solution
11: **endif**
12: Generate a new solution by flying randomly
13: Calculate the average loudness A^* of all solutions
14: **if** *rand* < A^* *and fun*(x_i) < *fun*(x_*) **Then**
15: Accept the new solutions
16: Increase r_i and reduce A_i
17: **endif**
18: Rank the bats and find the current best x_*
19: $t = t + 1$
20: **endwhile**
21: Postprocess results and visualization

Figure 1. The Bat algorithm (BA) [20].

$$x_i^t = x_i^{t-1} + v_i^t, \tag{5}$$

where $\beta \in [0,1]$ is a random parameter extracted from a uniform distribution and x_* is the current best position among the positions of all bats.

After calculating x_*, a local solution can be generated randomly for each bat based on the following equation:

$$x_{new} = x_{old} + \eta\, A^* \tag{6}$$

where $\eta \in [-1,1]$ is a tuning random parameter and A^* is the average loudness of all bats at instant t.

The update equations of the velocities, positions and frequencies of the bats are similar to the update equations of the velocities and positions of the particles in PSO (Section 3). Actually, BA can be considered as a combination of PSO and intensive local search that aims to balance between the exploration and exploitation of solutions.

In the nature, when a microbat finds a prey, it usually decreases the loudness and increases the pulse emission of its sound. This aspect is simplified in the BA algorithm by assuming that $A_{min}=0$ and $A0=1$. The assumption that $A_{min}=0$ indicates that a bat has located a prey and temporarily has stopped emitting any sound. In the beginning of the simulation process of BA, positive random values around zero are generated and assigned to the pulse emission of each bat, while positive random values around 1 are generated and assigned to the loudness of each bat.

At each iteration of the BA algorithm, a local search for a new best solution around one of the best solutions x_* (line 8) is triggered when the pulse rate is less than a randomly generated number $rand \in [0,1]$. Then, each time x* is improved (line 14) the pulse rate r_i is increased and the loudness A_i is decreased as follows:

$$A_i^{t+1} = \sigma\, A_i^t \tag{7}$$

$$r_i^{t+1} = r_i[1 - e^{-\delta t}] \tag{8}$$

where σ and δ are constant parameters that can be determined experimentally, however, as a general rule $0<\sigma<1$ and $\delta>0$ to guarantee that the loudness will decrease and the pulse rate will increase as new best solutions are discovered.

A new solution x_* is accepted if it satisfies two conditions. First, the estimation *fun(x_*)* of the new x_* must be better than the estimation of a randomly selected bat's solution. Second, the value of *rand* should be less than the average loudness of all the solutions.

The purpose of setting the loudness to a value near to one and the pulse rate to a value near to zero is to encourage the exploration of new solutions around the current best solutions. This is because a pulse rate near to zero is expected with a high probability to be less than the randomly generated number $rand \in [0,1]$ (line 8). Consequently, there is a high probability that a new solution would be generated around one of the best solutions (lines 8 to 11). As the values of pulse rates are increased each time a better solution is accepted (line 14), the probability of generating a new solution around one of the best solutions decreases (line 8).

3. RELATED WORK

This section provides an overview of well known cooperative Q-learning algorithms with special focus on the second learning stage of these algorithms.

Iima and Kuroe [11]-[13] proposed three cooperative Q-learning algorithms (BEST-Q, AVE-Q and PSO-Q) that allow multiple learners to share their Q-values after each round of independent learning. Each one of these algorithms evaluates its Q-values during the independent learning stage using an evaluation method that approximates the rewards [6], [13], [14]. This method evaluates each state-action pair (s,a) by calculating the sum of its discounted rewards $E(s,a)$ used to update its Q-value during an episode (independent learning stage). Discounting the reward is important to increase the weight of rewards while approaching the end of the episode. This is because the Q-values are in continuous change during the episode. At the end of the independent learning stage, each learner i calculates $E_i(s,a)$ for each $(s,a) \in S \times A$ as follows:

$$E_i(s,a) = \sum_{i=1}^{n} \gamma^{n-i}\, R_i(s,a) \tag{9}$$

where n denotes the number of times the state-action pair (s,a) has been updated by agent i during the episode, $R_i(s,a)$ is the reward received for performing action a at state s and γ is the discount parameter. The parameter γ is the same discount factor used in Equation 1. This parameter is used in Equation 9 to balance between the rewards received in the beginning of the episode with rewards received in the end of it.

In BEST-Q, the superior Q-values are extracted from the Q-tables of all of the learners, then copied to each Q-table of each agent. According to this description, an agent i updates $Q_i(s,a)$ for all $(s,a) \in S \times A$ as follows:

$$Q_i(s,a) \leftarrow Q^{best}(s,a). \quad (10)$$

In the above equation, $Q^{best}(s,a)$ of state-action pair (s,a) is the Q-value with the highest $E(s,a)$ for all agents. The main disadvantage of BEST-Q is that it might not find the optimal Q-values, because the Q-tables of all of the learners become the same after each update. As a result, the diversity of the Q-values is affected negatively [11].

AVE-Q is a modification of BEST-Q that retains the diversity of each agent's Q-values after the learning by interaction stage. In this algorithm, the Q-values of learner i are updated by averaging each Q-value in the learners' Q-table and its corresponding best Q-value for all $(s,a) \in S \times A$ as follows:

$$Q_i(s,a) \leftarrow \frac{Q^{best}(s,a) + Q_i(s,a)}{2}. \quad (11)$$

Actually, AVE-Q moves at the interaction stage into the middle of the agent's Q-values and their corresponding best values without investigating the quality of the agent's Q-values. As a consequence, AVE-Q may produce an incorrect policy, because it does not remove the bad Q-values at the interaction stage [3].

The Particle Swarm Optimization (PSO) algorithm is a powerful metaheuristic method that attempts to iteratively optimize a solution with respect to a particular measure [23]. An optimization problem can be solved with PSO by moving the candidate solutions (particles) in the search space based on their positions and velocities. The movement of a particle is controlled by the particle's local best position and directed in the direction of the global best positions in the search-space. The global best positions are the best positions found by all of the particles after each iteration of the algorithm.

PSO-Q uses PSO at its second learning stage as a Q-value sharing method. In this method, the particles are the Q-values and the qualitative measurer is the Q-function. The Q-table of each learner is updated based on the velocities and positions of the Q-values as follows [12]:

$$V_i(s,a) \leftarrow W V_i(s,a) + C_1 R_1 [P_i(s,a) - Q_i(s,a)] + C_2 R_2 [G(s,a) - Q_i(s,a)], \quad (12)$$

$$Q_i(s,a) \leftarrow Q_i(s,a) + V_i(s,a), \quad (13)$$

where V_i is the velocity of learner i for state-action pair (s,a), W, C_1 and C_2 are weight parameters and R_1 and $R_2 \in [0,1]$ are random numbers. In the above equation, the best Q-value found by agent i for (s,a) is denoted as $P_i(s,a)$ and the best Q-value found by all of the agents for (s,a) is denoted as $G(s,a)$. Normally, the value of $G(s,a)$ is estimated using Equation 10.

Two issues should be taken into consideration when implementing PSO-Q to a specific problem. First, determining suitable values for the parameters of PSO-Q usually requires multiple simulations to insure that PSO-Q will perform well. Second, there is no guarantee that PSO-Q will search outside the surroundings of the best Q-value for each possible combination of states and actions for all agents.

Ahmadabadi and Asadpour [18] proposed a cooperative Q-learning algorithm called Weighted Strategy Sharing (WSS). In this algorithm, each learner learns from its peers by following a two-step learning process. First, each learner assigns expertness values to the Q-tables of the other learners according to their relative expertness. Second, each learner updates its own Q-table by calculating the weighted average of the Q-values of the learners' Q-tables as follows:

$$Q_i(s,a) \leftarrow \sum_{j=1}^{n}(W_{ij}\ Q_j(s,a)) \tag{14}$$

where W_{ij} is the expertness value assigned by learner *i* to learner *j*'s expertness.

An expertness value can be evaluated using one of many expertness measures that have similar outcomes [18]. One such measure is the Normal measure (Nrm) which calculates the expertness of a learner (xr) by finding the sum of rewards that the learner has obtained during the previous independent learning stage:

$$xr_i^{Nrm} = \sum_{t=1}^{now} r_i(t)\ ; \tag{15}$$

where $r_i(t)$ is the reward that learner i obtains at instant t .

Based on the output of the above formula, learner i can assign a weight to the knowledge of learner j by taking into account the expertness of all learners as follows:

$$W_{ij} \leftarrow \frac{xr_i}{\sum_{k=1}^{n} xr_k}\ ; \tag{16}$$

where n is the number of learners and xr_k is the expertness of learner k for $k = 1, ..., n$.

A problem with WSS is that it might not converge to optimality when the shared Q-values are so extreme, because these values will deform the average Q-value [9].

Abed-alguni et al. [9] suggested a new cooperative Q-learning algorithm called average aggregation Q-learning which combines WSS, AVE-Q, BEST-Q and PSO-Q into one algorithm in order to reduce the instability in the performance of these algorithms for different problems. In this algorithm, each agent improves its Q-values by averaging the Q-values that resulted after implementing WSS, BEST-Q, AVE-Q and PSO-Q algorithms. Respectively, each agent i calculates $Q_i(s,a)$ for each $(s,a) \in S \times A$ as follows:

$$Q_i(s,a) \leftarrow \frac{Q^{BEST\text{–}Q}(s,a) + Q^{AVE\text{–}Q}(s,a) + Q^{WSS}(s,a) + Q^{PSO\text{–}Q}(s,a)}{4}; \tag{17}$$

where i is the learner's identification number and the denominator is the number of the combined algorithms.

Although average aggregation Q-learning solves the variability in performance for four famous cooperative Q-learning algorithms, it requires heavy computations to do so, because it mainly depends on the results of the other cooperative Q-learning algorithms.

In conclusion, there is no guarantee that the algorithms discussed in this section will converge to optimal solutions. Moreover, none of these algorithms has a stable performance when implemented to various learning problems [9]. The next section will present the BQ-learning algorithm that attempts to solve these problems.

4. BQ-LEARNING

The BQ-learning algorithm comprises two repetitive sequential learning stages.

- First Learning Stage: each learner tries independently to enhance its policy by applying Q-learning. Then, the Q-values of all the agents are evaluated by the evaluation method described in Section 3 - Equation 9.

- Second Learning Stage: the Q-values of all of the learners are updated through sharing of Q-values among the learners according to the evaluation results of the Q-values and the bat Q-value sharing strategy .

4.1 First Stage of BQ-learning

Figure 2 shows the BQ-learning algorithm. In the beginning of BQ-learning, the number of learners n and the total number of episodes of BQ-learning p should be specified. Also, the number of learning episodes m_i that each learner i performs during the first learning stage of BQ-learning should be specified. In addition, the Q-values and Q-value evaluations of each learner are initialized to zero (lines 8 to 10). That is, $Q_i(s,a) = 0$ and $E_i(s,a) = 0$ for all $(s,a) \in S \times A$ of each learner i.

```
 1: Q_i : Q -table of learner i .
 2: Q_i(s,a) : Q-value for state-action pair (s,a) of learner i .
 3: E_i(s,a) : evaluated value for Q_i(s,a) .
 4: n : number of learners.
 5: m_i : number of learning episodes performed by learner i during the first
        learning stage of BQ-learning.
 6: p : total number of episodes of BQ-learning.
 7: Begin
 8: for i = 1 to n do
 9:   Initialize Q_i(s,a) and E_i(s,a) for all (s,a) ∈ S × A .
10: end for
11: Set counter = 0 .
12: while counter < p do
13:   for i = 1 to n do
14:     Allow learner i to apply Q-learning as described in Section 2.1 for m_i
        episodes.
15:     Find E_i(s,a) for all (s,a) ∈ S × A based on Equation (9).
16:   end for
17:   Update Q_i(s,a) for all (s,a) ∈ S × A for all of the learners by interaction
        between the learners based on the bat Q-value sharing strategy described in
        Figure 3.
18:   counter = counter + n * m .
19: end while
20: End
```

Figure 2. The BQ-learning algorithm.

Lines 13 to 16 in Figure 2 represent the first learning stage of BQ-learning, where each learner i applies Q-learning for m_i episodes and then calculates $E_i(s,a)$ for all $(s,a) \in S \times A$ as described in Section 2.1. Allowing each learner to learn for the same number of episodes ($m_0 = m_1 = ... = m_{n-1} = m_n$) indicates that all of the n learners have equal levels of knowledge at the end of the first learning stage. On the other hand, allowing each learner to learn for a different number of episodes means that the learners have different levels of knowledge at the end of the first learning stage.

4.2 Second Stage of BQ-learning

1: Q_i: Q -table of learner i.
2: $Q_i(s,a)$: Q-value for state-action pair (s,a) for learner i.
3: $Q_*(s,a)$: best Q-value for state-action pair (s,a) for all agents.
4: $F_i(s,a)$: frequency for state-action pair (s,a) for learner i.
5: $V_i(s,a)$: difference of $Q_i(s,a)$ before and after its update.
6: $r_i(s,a)$: pulse rate for state-action pair (s,a) for learner i.
7: $A_i(s,a)$: loudness for state-action pair (s,a) for learner i.
8: $E_i(s,a)$: evaluated value for state-action pair (s,a) for learner i.
9: $E_*(s,a)$: evaluated value for $Q_*(s,a)$.
10: $A^*(s,a)$: average loudness of all $A_i(s,a)$.
11: **Begin**
12: Set the objective function as the evaluation function of Q-values (Equation (9)).
13: **for** $i = 1$ to n **do**
14: Initialize $F_i(s,a)$, $r_i(s,a)$, $A_i(s,a)$ and $V_i(s,a)$ for all $(s,a) \in S \times A$.
15: **endfor**
16: Find $Q_*(s,a)$ by applying the update function of BEST-Q algorithm (Equation (10)).
17: **while** $t <$ *Max number of iterations* **do**
18: **for** $i = 1$ to n **do**
19: Update $V_i(s,a)$, $F_i(s,a)$ and $Q_i(s,a)$ for all $(s,a) \in S \times A$. [Equations 18 to 20].
20: **endfor**
21: Generate a random number ($rand \in [0,1]$).
22: **if** $(rand > r_*(s,a))$ **then**
23: Allow learner i^* to apply Q-learning for few times starting from state s of $Q_*(s,a)$ using Equation 21.
24: **endif**
25: Randomly select $Q_i(s,a)$.
26: **if** $(rand < A^*(s,a)$ **and** $E_i(s,a) < E_*(s,a))$ **then**
27: Accept the new Q-values.
28: Increase $r_i(s,a)$ and reduce $A_i(s,a)$ [Equations 22 and 23] .
29: **endif**
30: Find $Q_*(s,a)$ by applying the update function of BEST-Q algorithm (Equation (10)).
31: $t = t + 1$
32: **endwhile**
33: **End**

Figure 3. Bat Q-value sharing strategy.

Line 17 in Figure 2 represents the second learning stage of BQ-learning that is described in details in Figure 3. It is important to keep in mind that the second learning stage of BQ-learning is what really distinguishes it from the other cooperative Q-learning algorithms described in Section 3.

Figure 3 shows the flow of the proposed Q-value sharing strategy that is based on the Bat algorithm. In Figure 3, the Q-values represent the locations of the bats (line 2), the velocity of a Q-value *V(s,a)* is the rate at which it changes (line 5) and the objective function is the evaluation

function *E(s,a)* of the Q-values (line 12). Line 14 in Figure 3 shows that the frequency, loudness and pulse rate for all $(s,a) \in S \times A$ of each learner i are initialized to zero. Then, in line 16, the best Q-value of each $(s,a) \in S \times A$ is calculated.

Line 19 of this algorithm shows that the Q-values and their frequencies and velocities are updated iteratively. The new Q-value $Q_i(s,a)$, velocity $V_i(s,a)$ and frequency $F_i(s,a)$ of learner i are given by:

$$F_i(s,a) = F_{min} + (F_{max} - F_{min})\beta, \tag{18}$$

$$V_i(s,a) = V_i(s,a) + (Q_i(s,a) - Q_*(s,a))F_i(s,a), \tag{19}$$

$$Q_i(s,a) = Q_i(s,a) + V_i(s,a), \tag{20}$$

where $\beta \in [0,1]$ is a random tuning parameter and $Q_*(s,a)$ is the current best Q-value among all n Q-values for the state-action pair (s,a).

A local Q-value can be generated around $Q_*(s,a)$ for each learner by allowing one of the learners to enhance $Q_*(s,a)$ by applying Equation 21 to $Q_*(s,a)$ for few times:

$$Q_*(s,a) \leftarrow (1-\alpha)\, Q_*(s,a) + \alpha\, [R(s,a) + A^* * max_{a \in' A} Q_*(s',a')]; \tag{21}$$

where $\alpha \in [0,1]$ is the same learning rate used in Equation 1 and A^* is the average loudness of all Q-values at iteration t. In the above equation, A^* is used to control the influence of future rewards instead of γ in Equation 1.

At each iteration of the algorithm, a local search for a new best Q-value (line 22) around the current best Q-value $Q_*(s,a)$ for each (s,a) is triggered when the pulse rate $r_*(s,a)$ is less than a randomly generated number $rand \in [0,1]$.

The new Q-value of $Q_*(s,a)$ is accepted if it satisfies two conditions. First, the estimation $E_*(s,a)$ of the new $Q_*(s,a)$ must be better than the estimation of a randomly selected Q-value for the same (s,a). Second, the value of *rand* should be less than the average loudness of (s,a) of all the learners. Fulfilling these conditions also implies that the pulse rate $r_i(s,a)$ should be increased and the loudness $A_i(s,a)$ should be decreased as follows:

$$A_i(s,a) = \sigma\, A_i(s,a), \tag{22}$$

$$r_i(s,a) = r_i(s,a)[1 - e^{-\gamma t}]; \tag{23}$$

where σ and δ are constant parameters. As a general rule, $0 < \sigma < 1$ to decrease the loudness and $\delta > 0$ to increase the pulse rate each time the Q-values are improved.

Assigning a low pulse rate $r_i(s,a)$ for each (s,a) in the beginning of the optimization process (line 17) and then increasing it (line 28) is an essential factor for the success of the algorithm. This is because it reduces the rate of local search around $Q_*(s,a)$ as BQ-learning is approaching the best Q-value.

The local search for the best Q-values can be performed simultaneously by multiple agents in BQ-learning as well as in other optimization-based cooperative Q-learning algorithms, such as PSO-Q and average aggregation Q-learning. BQ-learning is expected to perform better than the cooperative Q-learning algorithms discussed in Section 3, because it attempts to balance between the exploration and exploitation of the nominated best Q-values for sharing using tuning techniques that control its parameters (frequencies, pulse emission rates and loudness of

the potential solutions). Neither BEST-Q nor AVE-Q attempts to search around the best Q-values before sharing them. Consequently, BEST-Q might not find the optimal Q-values [11], while AVE-Q may produce an incorrect policy [11]-[12].

5. EXPERIMENTS

In this section, the performance of BQ-learning was compared with the performance of single-agent Q-learning, AVE-Q, BEST-Q, PSO-Q, WSS and average-aggregation Q-learning (Section 3) using two problems: the shortest path problem [12] and the taxi problem [24]. These problems have been widely used in the literature to evaluate the performance of cooperative Q-learning algorithms [12]-[13], [24]-[26].

5.1 Test Problems

RL can be applied to two types of learning problems [24]. First, single-task problems (e.g., shortest path problem), in which the learner is required to learn a single task. Second, multi-task problems (e.g., taxi domain problem), in which the learner is required to learn multiple tasks.

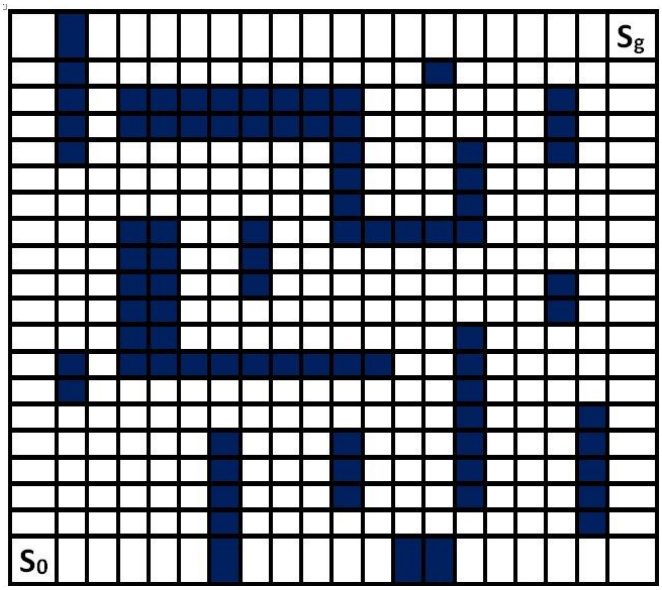

Figure 4. An example of shortest path problem on a grid size of 20×20.

The shortest path problem is a single-task problem that has been used in many research studies to evaluate the efficiency of cooperative Q-learning algorithms [11]-[13]. In this problem, an agent is required to learn one task which is finding the shortest path from one cell to another in a grid, such that the number of visited cells is minimized. The grid in this problem is usually represented as a two-dimensional array that is indexed by two subscripts, one for the row and one for the column. In the shortest path problem, the target cell is usually specified prior to learning and the start cell is randomly selected before the beginning of each learning episode. The learner can move during each episode in four directions (up, down, right and left) as long there are no obstacles or barriers obstructing its way. Figure 4 shows an example of shortest path problem on a grid size of 20×20. Filled squares represent obstacles that the agent cannot pass, s_0 is the start cell and s_g is the target cell.

The taxi domain problem is an episodic multi-task problem that has been used in many research studies to evaluate the performance of hierarchical Q-learning algorithms [24]-[26]. In each episode, a taxi agent in a grid world of size 5×5 is required to perform multiple tasks: finding a customer, picking up the customer, driving the customer to a destination location and dropping

down the customer in the destination location. The taxi agent can accomplish these goals by choosing actions from a set of six actions: move one cell (left, right, top or bottom), pickup action and drop off action. If any of these actions that leads the taxi agent to a barrier or a wall cell, the location of taxi agent remains unchanged. In the grid, there are four source and destination locations. Figure 5 shows an example of taxi domain problem. In the figure, a taxi is located on a 5×5 grid. There are four pre-determined locations in the grid, marked as Red (R), Blue (B), Green (G) and Yellow (Y). In the beginning of the simulation process, one of these locations is selected as a pick-up point and another location is selected as a drop-off point.

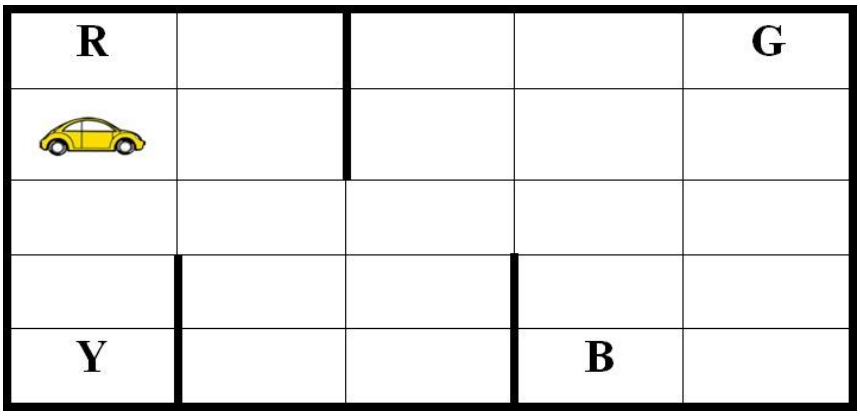

Figure 5. Taxi domain problem.

5.2 Setup

The shortest path problem in Figure 4 was modeled as an MDP as follows:

- The cells in the 20×20 grid represent the states of the MDP:

$$S = (grid[0][0], grid[0][1], \ldots, grid[19][19]).$$

- The target cell is specified prior to learning and the start cell is randomly selected before the beginning of each learning episode.
- There are four primitive actions in the shortest path problem:

$$A = (\text{moveup}, \text{movedown}, \text{moveleft}, \text{moveright}).$$

- The reward model for the learners is defined as follows:

$$R(s,a) = \begin{cases} +10.0 & \text{if it reached the target cell} \\ 0 & \text{otherwise} \end{cases}$$

- The transition model for the learners is:

$$T(s,a,s') = \begin{cases} 1 & \text{if } s' \text{ is next to } s \text{ in the direction of } a \text{ and is} \\ & \text{not a barrier or wall} \\ 0 & \text{otherwise} \end{cases}$$

The taxi domain problem in Figure 5 was modeled as an MDP as follows:

- The cells in the 5×5 grid represent the states of the MDP:

$$S = (grid[0][0], grid[0][1], \ldots, grid[4][4]).$$

- The location of the taxi is specified prior to learning.

- The location of the passenger (source) and the dropping point of the passenger (destination) are randomly chosen in the beginning of each learning episode.
- There are six primitive actions in the taxi domain problem:

$$A = (\text{move up}, \text{move down}, \text{move left}, \text{move right}, \text{pick up}, \text{put down}).$$

- The reward model for the learners is defined as follows:

$$R(s,a) = \begin{cases} +10.0 & \text{If the passenger was delivered to its selected drop off point} \\ -10 & \text{illegal pickup or putdown} \\ 0 & \text{otherwise} \end{cases}$$

- The transition model for the learners is:

$$T(s,a,s') = \begin{cases} 1 & \text{if } s' \text{ is next to } s \text{ in the direction of } a \text{ and is not a barrier or wall} \\ 0 & \text{otherwise} \end{cases}$$

The experiments were implemented using two models of knowledge [19]. In the first model, the learners were assumed to have equal levels of knowledge. This was simulated by allowing the learners to learn for the same number of episodes before sharing of their Q-values. In the second model, the learners were assumed to have different levels of knowledge, which was achieved by allowing each learner to learn for a different number of episodes each time it is learning independently. For example, a learner that has learned for 25 episodes has more practical knowledge than a learner that has learned only for 10 episodes.

The action selection policy was the ε-soft policy, in which a random action is uniformly selected with probability ε and the action with the highest expected reward is chosen the rest of the time [22].

The learning parameters for the experiments were set as follows:

- In Q-learning, the learning rate α was tuned dynamically, so that low Q-values have larger learning rates than high Q-values as recommended by Ray and Oates [27]. The discount factor $\gamma = 1$ [28].
- In all the cooperative Q-learning algorithms, the learning rate $\alpha = 0.01$ and the discount factor $\gamma = 0.9$. These values ensure that each cooperative learner learns adequately and make, the best use of its current knowledge at each learning episode as recommended by Abed-alguni *et al.* [9].
- In each episode, a learner starts learning from a randomly selected state and finishes learning when a goal state is reached. Otherwise, the learner finishes learning after 5,000 moves without meeting its goal.
- In order to ensure an adequate exploration/exploitation ratio, the probability of selecting a random action was $\varepsilon = 0.05$ in the ε-soft selection policy.
- The Nrm measure was selected as the expertness measure of WSS. This measure has a similar performance to the performance of all other tested expertness measures.
- As in Abed-alguni et al. [9], the weight parameters in PSO-Q were $W = 0$, $C1 = C2 = 1$.
- In BQ-learning, the frequency, the loudness and the pulse rate were in the range [0,1] for each solution. The discount parameter of the frequencies $\beta=0.5$. Initially, the loudness A was set to 1 and the pulse rate r was set to 0 for each Q-value.

Three agents are involved in the experiments. The total number of learning episodes is 2,000 for the shortest path problem, while the total number of learning episodes for the taxi problem is

12,000 episodes. Each algorithm was executed 100 times in order to provide meaningful statistical analysis of the experiments.

In the experiments, an algorithm is considered to have converged to a good policy when the average number of moves in its policy enhances by less than one move over 100 successive episodes.

5.3 Experimental Results

5.3.1 Shortest Path Problem

Figure 6 shows the average number of moves per 10 episodes to find the shortest path to the goal state in a 20×20 grid. The second learning stage of the cooperative Q-learning algorithms takes place after each 10 episodes of individual learning. We can see from the figure that BQ-learning converges after 420 episodes to a solution. On the other hand, single agent Q-learning, AVE-Q, WSS, average aggregation Q-learning and PSO-Q converge after around 520 episodes to solutions, while BEST-Q requires 60 additional episodes to converge to a solution. These results suggest that the performance of BQ-learning is better than those of the other algorithms in single-task problems when the agents have similar levels of knowledge before sharing.

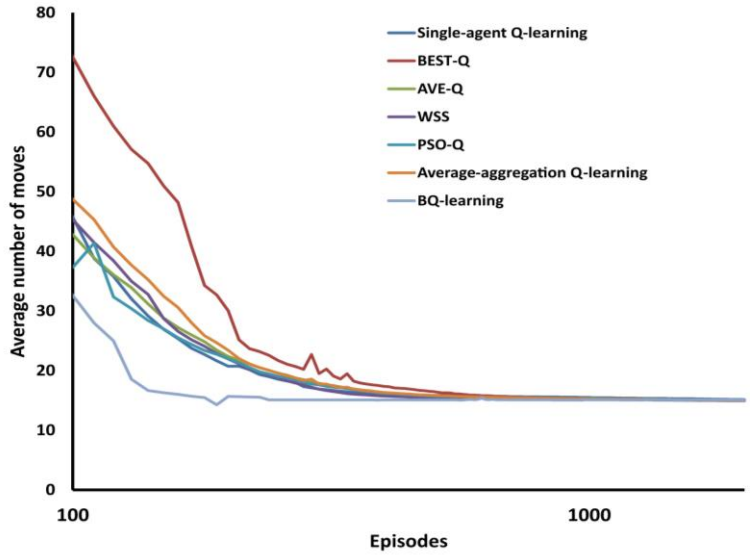

Figure 6. Experiment 1: Average number of moves per 10 episodes in a shortest path problem of a grid size of 20×20. Each curve is the average of 100 runs. Sharing of Q-values takes place after each ten episodes of individual learning.

Figure 7 shows the average number of moves per 10 episodes to find the shortest path to the goal state in a 20×20 grid. Respectively, in Figure 7, the first, the second and the third agents learn for 10, 5 and 1 episodes before sharing of their Q-values among each other. In this experiment, BQ-learning requires 300 episodes of learning to converge to a solution, which is only 14.9% of the number of episodes required for single-agent Q-learning (2020), 54% of BEST-Q (550), 53.6% of WSS (560), 62.5% of PSO-Q (480), 61.2% of AVE-Q (490) and 29.4% of average aggregation Q-learning (1,020). These results mean that BQ-learning outperforms the other algorithms in single-task problems when the agents have different levels of knowledge before sharing.

5.3.2 Taxi Problem

Figures 8 and 9 show the average number of steps per 10 episodes to deliver a passenger in a

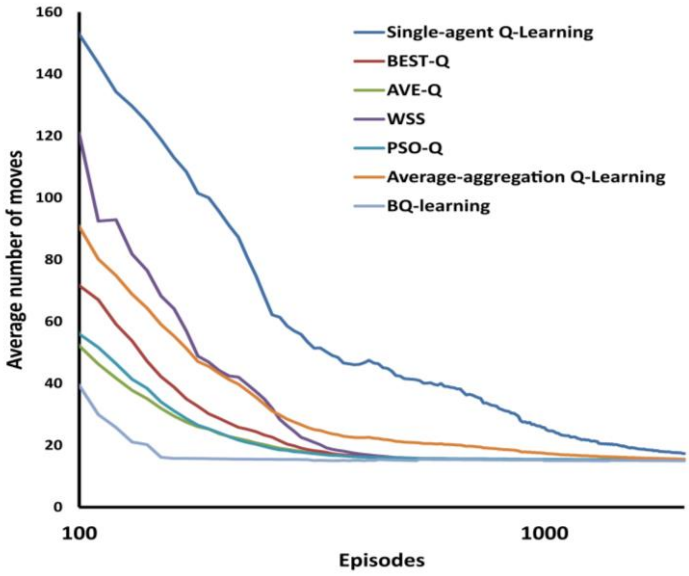

Figure 7. Experiment 2: Average number of moves per 10 episodes in a shortest path problem of a grid size of 20×20. Each curve is the average of 100 runs. One, five and ten episodes of learning occur before implementing a Q-value sharing strategy.

5×5 grid. Sharing of Q-values occurs in Figure 8 after each 10 episodes of independent learning, while in Figure 9, the 1st learner, the 2nd learner and the 3rd learner respectively learn for 1, 5 and 10 episodes before sharing of their Q-values.

Figure 8 shows that BQ-learning requires 7,180 episodes to converge to a solution, followed by PSO-Q that requires 7,560 episodes (5% more episodes than BQ-learning) to converge to a solution. On the other hand, the other algorithms failed to converge to a solution at the end of the simulation process. These results suggest that BQ-learning converges to a solution faster than the other algorithms in multi-task problems when the agents have similar experiences.

Figure 9 shows that all of the cooperative Q-learning algorithms failed to converge to a solution except BQ-learning and AVE-Q. As expected, BQ-learning has the fastest convergence speed among all algorithms. From Figure 9, we can also see that WSS has the worst performance among all the algorithms including single-agent Q-learning. These results indicate that BQ-learning outperforms the other algorithms in multi-task problems when the agents have different levels of experience.

5.4 Performance Analysis

Two statistical measures were used in Table 1 to compare the performance of the tested algorithms over 100 runs. The results are in the format: average number of iterations $\pm$ standard deviation of iterations. The last row of the table shows that BQ-learning requires less number of iterations to converge to a solution. In addition, the standard deviations of the number of iterations of BQ-learning are the lowest among all the standard deviations of the other algorithms that converge to a solution. This means that the performance of BQ-learning is more stable than the performance of the other tested algorithms.

Figures 10 and 11 show how the performance of three instances of BQ-learning is affected as the number of learning episodes of the agents is varied: (1-5-10), (15-30-45) and (25-50-100) learning episodes before sharing. The results in Figure 10 show that the convergence points of all instances of BQ-learning in the shortest path problem are not far from each other: BQ-learning (1-5-10) converges after 303 episodes, BQ-learning (15-30-45) converges after 333 episodes and BQ-learning (25-50-100) converges after 342 episodes.

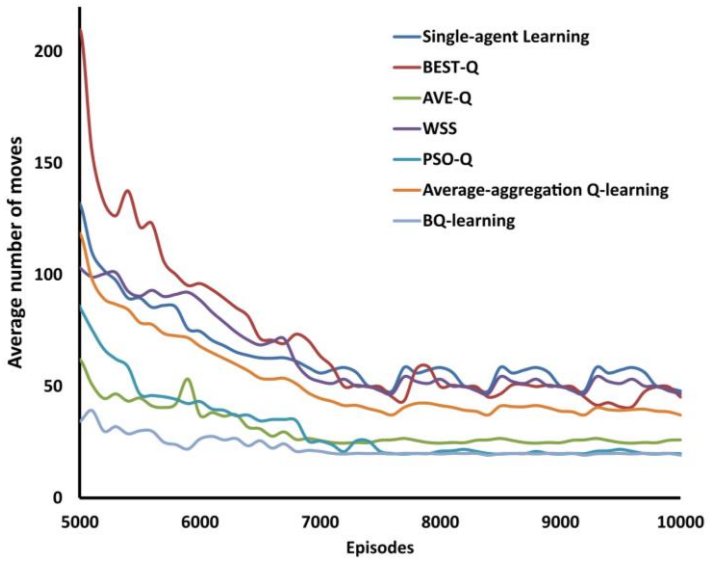

Figure 8. Experiment 3: Average number of moves per 10 episodes in a taxi problem of a grid size of 5×5. Each curve is the average of 100 runs. Sharing of Q-values takes place after each ten episodes of individual learning.

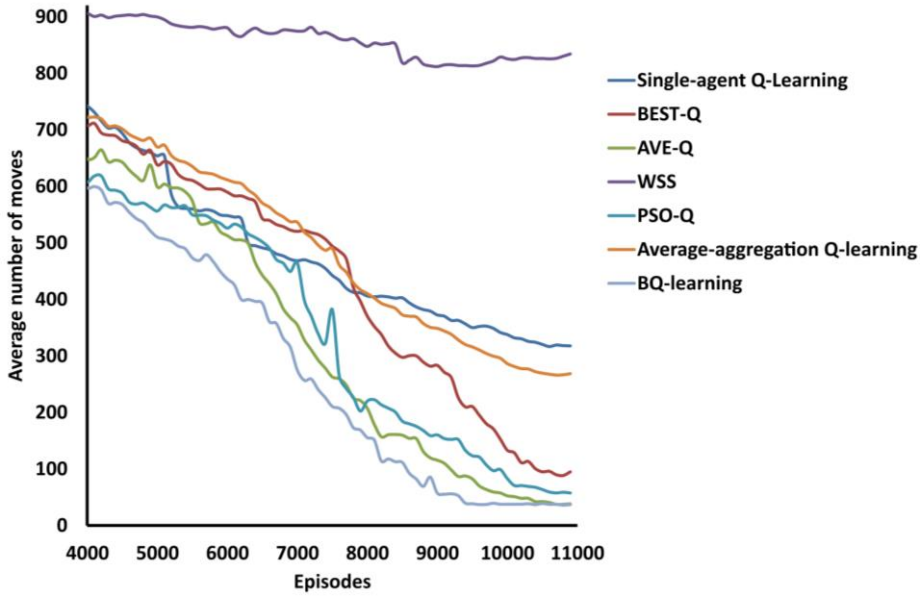

Figure 9. Experiment 4: Average number of moves per 10 episodes in a taxi problem of a grid size of 5×5. Each curve is the average of 100 runs. One, five and ten episodes of learning occur before implementing a Q-value sharing strategy.

In Figure 11 (taxi domain), BQ-learning (1-5-10) converges after 9,136 episodes, BQ-learning (15-30-45) converges after 9,233 episodes and BQ-learning (25-50-100) converges after 9,417 episodes. To sum up, the results in both figures indicate that the convergence speed of BQ-learning is not highly sensitive to the number of episodes that each agent learns before sharing of Q-values.

The overall results of the experiments suggest that BQ-learning performs better than conventional Q-learning and the other cooperative Q-learning algorithms, regardless of the

levels of experience of the agents (similar experiences vs. different experiences) and the types of the learning problems (single-task vs. multiple-task problems).

Table 1. Average and standard deviation of number of iterations over 100 runs. The star symbol * indicates that the algorithm did not converge to a solution at the end of the simulation process.

Algorithm	Experiment 1	Experiment 2	Experiment 3	Experiment 4
Single-agent Q-learning	520 ± 36	$2{,}000 \pm 0^{*}$	$12{,}000 \pm 0^{*}$	$12{,}000 \pm 0^{*}$
BEST-Q	583 ± 44	550 ± 66	$12{,}000 \pm 0^{*}$	$12{,}000 \pm 0^{*}$
AVE-Q	521 ± 31	480 ± 35	$12{,}000 \pm 0^{*}$	$11{,}003 \pm 356$
WSS	524 ± 32	560 ± 44	$12{,}000 \pm 0^{*}$	$12{,}000 \pm 0^{*}$
PSO-Q	523 ± 34	480 ± 61	$7{,}560 \pm 701$	$12{,}000 \pm 0^{*}$
Average aggregation	521 ± 31	$1{,}020 \pm 52$	$12{,}000 \pm 0^{*}$	$12{,}000 \pm 0^{*}$
BQ-learning	413 ± 17	300 ± 32	$7{,}180 \pm 425$	$9{,}136 \pm 256$

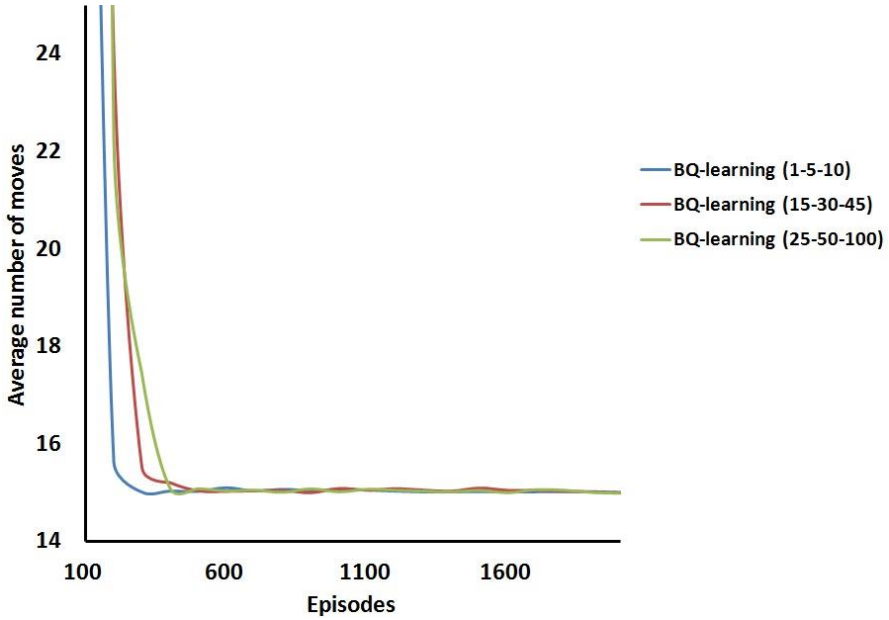

Figure 10. Experiment 5: Performance of different instances of BQ-learning in a shortest path problem of a grid size of 20×20. Each curve is the average of 100 runs.

6. Conclusion and Future Work

Cooperative Q-learning approach is an efficient learning approach that accelerates the learning process of individual learners in homogeneous multi-agent systems. This paper presented the BQ-learning algorithm which is a new cooperative Q-learning that is inspired from the bat algorithm. The learning process of BQ-learning comprises two stages. First, the individual learning stage, where each agent learns or improves its own policy by implementing the standard Q-learning algorithm. Second, the learning by interaction stage, where the learners share their Q-values among each other using a Q-value sharing strategy based on the bat

algorithm.The BQ-learning algorithm has many advantages. First, compared to current cooperative Q-learning algorithms, the BQ-learning algorithm can be implemented to single-task and multi-task problems, because optimizing the tasks of a learning problem using the bat algorithm improves the overall solution for the problem. Second, the bat sharing strategy in BQ-learning increases the possibility of finding the optimal Q-values, because it attempts to balance between the exploration and exploitation of actions using tuning techniques that control its parameters (frequencies, pulse emission rates and loudness of the potential solutions).

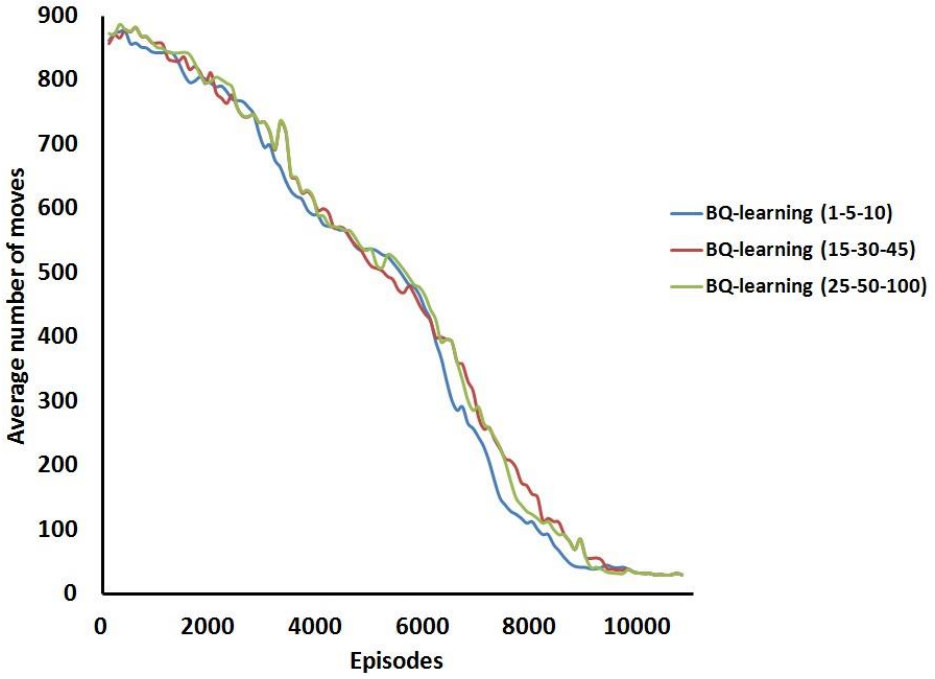

Figure 11. Experiment 6: Performance of different instances of BQ-learning in a taxi problem of a grid size of 5×5. Each curve is the average of 100 runs.

Finally, the results of the pilot experiments suggest that BQ-learning performs faster than single-agent Q-learning and other famous cooperative Q-learning algorithms, whether the agents have similar or different levels of experience and regardless of the type of the learning problems (single-task vs. multiple-task problems).

Future work includes implementing the BQ-learning algorithm to continuous space learning problems and developing a new cooperative Q-learning algorithm based on a combination of the firefly and monkey algorithms.

REFERENCES

[1] C. Watkins, Learning from Delayed Rewards, PhD thesis, Cambridge University, Cambridge, England, 1989.

[2] C. Watkins and P. Dayan, "Technical Note: Q-learning," Machine Learning, vol. 8, no. 3, pp. 279-292, 1992.

[3] B. H. Abed-alguni, S. K. Chalup, F. A. Henskens and D. J. Paul, "A Multi-agent Cooperative Reinforcement Learning Model Using a Hierarchy of Consultants, Tutors and Workers," Vietnam Journal of Computer Science, vol. 2, no. 4, pp. 213-226, 2015.

[4] P. Kormushev, S. Calinon and D. G. Caldwell, "Reinforcement Learning in Robotics: Applications and Real-world Challenges," Robotics, vol. 2, no. 3, pp. 122-148, 2013.

[5] J. Kober, J. A. Bagnell and J. Peters, "Reinforcement Learning in Robotics: A Survey," The International Journal of Robotics Research, vol. 32, no. 11, pp. 1238-1274, 2013.

[6] H. Iima, Y. Kuroe and S. Matsuda, "Swarm Reinforcement Learning Method Based on Ant Colony Optimization," Proc. of 2010 IEEE International Conference on Systems, Man and Cybernetics (SMC), (O. Kaynak and G. Dimirovski, eds.), pp. 1726-1733, 2010.

[7] B. Cunningham and Y. Cao, "Non-reciprocating Sharing Methods in Cooperative Q-learning Environments," Proc. of the 2012 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology, vol. 2, pp. 212-219, IEEE Computer Society, 2012.

[8] M. Tan, "Multi-agent Reinforcement Learning: Independent vs. Cooperative Agents," Proc. of 10th International Conference on Machine Learning, vol. 337, Amherst, MA, 1993.

[9] B. H. Abed-alguni, D. J. Paul, S. K. Chalup and F. A. Henskens, "A Comparison Study of Cooperative Q-learning Algorithms for Independent Learners," International Journal of Artificial Intelligence, vol. 14, no. 1, pp. 71-93, 2016.

[10] H. Iima, Y. Kuroe and K. Emoto, "Swarm Reinforcement Learning Methods for Problems with Continuous State-action Space," Proc. of 2011 IEEE International Conference on Systems, Man and Cybernetics (SMC), pp. 2173-2180, 2011.

[11] H. Iima and Y. Kuroe, "Reinforcement Learning through Interaction among Multiple Agents," Institute of Control, Automation and Systems Engineers (ICASE) and the Society of Instrument and Control Engineers (SICE) International Joint Conference, pp. 2457-2462, October 2006.

[12] H. Iima and Y. Kuroe, "Swarm Reinforcement Learning Algorithms-Exchange of Information among Multiple Agents," The Society of Instrument and Control Engineers (SICE), 2007 Annual Conference, pp. 2779-2784, September 2007.

[13] H. Iima and Y. Kuroe, "Swarm Reinforcement Learning Algorithms Based on SARSA Method," The Society of Instrument and Control Engineers (SICE) Annual Conference 2008, pp. 2045-2049, August 2008.

[14] E. Di Mario, Z. Talebpour and A. Martinoli, "A Comparison of PSO and Reinforcement Learning for Multi-robot Obstacle Avoidance," 2013 IEEE Congress on Evolutionary Computation (CEC), pp. 149-156, June 2013.

[15] B. Dğan and T. Ölmez, "A Novel State Space Representation for the Solution of 2D-HP Protein Folding Problem Using Reinforcement Learning Methods," Applied Soft Computing, vol. 26, pp. 213-223, 2015.

[16] E. Pakizeh, M. Palhang and M. Pedram, "Multi-criteria Expertness Based Cooperative Q-learning," Applied Intelligence, vol. 39, no. 1, pp. 28-40, 2013.

[17] K.-S. Hwang, W.-C. Jiang and Y.-J. Chen, "Model Learning and Knowledge Sharing for a Multi-agent System with Dyna Q-learning," IEEE Transactions on Cybernetics, vol. 45, pp. 964-976, May 2015.

[18] M. N. Ahmadabadi and M. Asadpour, "Expertness Based Cooperative Q-learning," IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics, vol. 32, no. 1, pp. 66-76, 2002.

[19] M. Ahmadabadi, A. Imanipour, B. Araabi, M. Asadpour and R. Siegwart, "Knowledge-based Extraction of Area of Expertise for Cooperation in Learning," International Conference on Intelligent Robots and Systems, 2006 IEEE/RSJ, pp. 3700-3705, 2006.

[20] X.-S. Yang, "A New Metaheuristic Bat-inspired Algorithm," Conference on Nature Inspired Cooperative Strategies for Optimization (NICSO 2010), pp. 65-74, Springer, 2010.

[21] X.-S. Yang and A. Hossein Gandomi, "Bat Algorithm: A Novel Approach for Global Engineering Optimization," Engineering Computations, vol. 29, no. 5, pp. 464-483, 2012.

[22] X.-S. Yang, "Bat Algorithm for Multi-objective Optimization," International Journal of Bio-Inspired Computation, vol. 3, no. 5, pp. 267-274, 2011.

[23] J. Kennedy, "Particle Swarm Optimization," Encyclopaedia of Machine Learning, pp. 760-766, Springer, 2011.

[24] B. Hengst, "Model Approximation for HEXQ Hierarchical Reinforcement Learning," Machine Learning: ECML 2004 (J.-F. Boulicaut, F. Esposito, F. Giannotti and D. Pedreschi, eds.), vol. 3201 of Lecture Notes in Computer Science, pp. 144-155, Springer, Berlin, Heidelberg, 2004.

[25] T. G. Dietterich, "Hierarchical Reinforcement Learning With the MAXQ Value Function Decomposition," Journal of Artificial Intelligence Research, vol. 13, no. 1, pp. 227-303, 2000.

[26] D. Andre and S. J. Russell, "State Abstraction for Programmable Reinforcement Learning Agents," AAAI/IAAI, pp. 119-125, 2002.

[27] S. Ray and T. Oates, "Locking in Returns: Speeding Up Q-learning by Scaling," Proc. European Workshop on Reinforcement Learning (EWRL), pp. 32-36, 2011.

[28] R.-S. Sutton and A.-G. Barto, Reinforcement Learning: An Introduction, Cambridge, USA: MIT Press, 1998.

ملخص البحث:

يسمح منحنى تعلُّم كيو التعاوني لمتعلّمين متعدّدين بالتعلُّم بشكلٍ مستقلّ ومن ثمّ بتبادل قيم كيو الخاصة بهم فيما بينهم مستخدمين استراتيجية لتبادُل قيم كيو. وهناك مشكلة رئيسية ترتبط بهذا المنحنى، تتمثل في أن حُلول المتعلّمين قد لا تلتقي إلى المثالية؛ لأن قيم كيو المثالية قد لا يتمّ إيجادها. وهناك مشكلة أخرى هي أنّ بعض الخوارزميات التعاونية تكون ذات أداء جيّد فيما يتعلّق بالمشكلات ذات المهمّة الواحدة بينما يكون أداؤها ضعيفاً في المشكلات متعدّدة المهامّ.

تقترح هذه الورقة خوارزمية جديدة لتعلُّم كيو، وهي خوارزمية تعاونية تسمى خوارزمية الخفّاش، وتقوم بتنفيذ استراتيجية لتبادُل قيم كيو منبثقة عنها. والجدير بالذكر أنّ خوارزمية الخفّاش خوارزمية فعّالة تزيد من إمكانية إيجاد قيم كيو المثالية عن طريق إحداث التوازن بين الاستكشاف والاستغلال للأفعال عبر ضبط متغيرات الخوارزمية.

وقد تم اختبار خوارزمية الخفّاش لتعلُّم كيو باستخدام مسألتين هما: مسألة أقصر مسار، وهي مسألة أحادية المهمة، ومسألة التاكسي، وهي مسألة متعدّدة المهامّ. وتقترح النتائج التجريبية أنّ أداء خوارزمية الخفّاش لتعلُّم كيو أفضل عند مقارنته بأداء تعلُّم كيو ذي العامل الواحد وأداء بعض الخوارزميات التعاونية لتعلُّم كيو المعروفة جيداً.

EDITORIAL BOARD SUPPORT TEAM

LANGUAGE EDITOR

Haydar Al-Momani

EDITORIAL BOARD SECRETARY

Eyad Al-Kouz

JJCIT ADDRESS

WEBSITE: www.jjcit.org

EMAIL: jjcit@psut.edu.jo

ADDRESS: Princess Sumaya University for Technology, Khalil Saket Street, Al-Jubaiha.

B.O. BOX: 1438 Amman 11941 Jordan.

TELEPHONE: +962-6-5359949.

FAX: +962-6-7295534.

المجلة الأردنية للحاسوب و تكنولوجيا المعلومات

ISSN 2415 - 1076 (Online)
ISSN 2413 - 9351 (Print)

نيسان ٢٠١٧ المجلد ٣ العدد ١

JJCIT

www.jjcit.org

jjcit@psut.edu.jo

مجلة علمية عالمية متخصصة محكمة

تصدر بدعم من صندوق دعم البحث العلمي