In the Name of God

Journal of
Information Systems & Telecommunication
Vol. 11, No.2, April-June 2023, Serial Number 42

Research Institute for Information and Communication Technology
Iranian Association of Information and Communication Technology
Affiliated to: Academic Center for Education, Culture and Research (ACECR)

Print ISSN: 2322-1437
Online ISSN: 2345-2773
Publication License: 91/13216
Editorial Office Address: No.5, Saeedi Alley, Kalej Intersection., Enghelab Ave., Tehran, Iran,
P.O.Box: 13145-799 Tel: (+9821) 88930150 Fax: (+9821) 88930157
E-mail: info@jist.ir , infojist@gmail.com
URL: www.jist.ir

Indexed by:
- SCOPUS www. Scopus.com
- Index Copernicus International www.indexcopernicus.com
- Islamic World Science Citation Center (ISC) www.isc.gov.ir
- Directory of open Access Journals www.Doaj.org
- Scientific Information Database (SID) www.sid.ir
- Regional Information Center for Science and Technology (RICeST) www.ricest.ac.ir
- Magiran www.magiran.com

Publisher:
Iranian Academic Center for Education, Culture and Research **(ACECR)**

This Journal is published under scientific support of
Advanced Information Systems (AIS) Research Group and
Telecommunication Research Group, ICTRC

Acknowledgement

JIST Editorial-Board would like to gratefully appreciate the following distinguished referees for spending their valuable time and expertise in reviewing the manuscripts and their constructive suggestions, which had a great impact on the enhancement of this issue of the JIST Journal.

(A-Z)

- Afsharirad, Majid, Kharazmi University,Tehran, Iran
- Azizi, Sadoon, University of Kurdistan, Sanandaj, Iran
- Badie, Kambiz, Tehran University, Iran
- Bohrani, Mohammad, Allameh Tabatabaei University, Tehran, Iran
- Fathi, Amir, Urmia University, Urmia, Iran
- Ghaffari, Hamidreza, Ferdous Azad University, South Khorasan Province, Iran
- Ghasemi, Majid, Islamic Azad University Shahr-e-Qods, Iran
- Gerami,Mohsen, ICT Research Institute, Tehran, Iran
- Hasan, Junayed, Universiteit van Ulsan, Ulsan, South Korea
- Izadkhah, Habib, Tabriz University, Tabriz, Iran
- Khazaei, Mehdi, Kermanshah University of Technology, Kermanshah, Iran
- Kolahkaj, Maral, Islamic Azad University, Karaj Branch, Iran
- Kheirkhah, Fatemeh, ACECR, Tehran, Iran
- Mirzaei, Abbas, Islamic Azad University, Ardabil, Iran
- Moradi, Gholamreza, Amirkabir University,Tehran, Iran
- Omid Mahdi, Ebadati, Kharazmi University, Tehran, Iran
- Qingliang, Zhao, Northwestern University, Evanston, United State
- Qhasemzadeh Mohammad, Yazd University, Tehran, Iran
- Rizal, Achmad, Telkom University, Bandung , Indonesia
- Rasi, Habib, Shiraz University of Technology, Shiraz, Iran
- Rane, Milind, Vishwakarma Institute of Technology, Pune, Maharashtra, India
- Rahman, Saifur, Virginia Tech,Riva San Vitale, Switzerland,
- Roshani, Saeid, Islamic Azad University, Kermanshah Branch, Iran
- Roy, Sahadev Roy, NIT Arunachal Pradesh,
- Sable, Nilesh, Vishwakarma Institute of Technology, Pune, Maharashtra, India
- Saadatfar, Hamid, University of Birjand, Iran
- Shakibian, Hadi, Alzahra University, Tehran, Iran
- Tanha, Jafar, Tabriz University, Tabriz, Iran
- Tourani, Mahdi, University of Birjand, South Khorasan, Iran
- Tashtarian, Farzad, Islamic Azad Mashad University, Mashad, Iran
- Zayyani, Hadi, Qom University of technology, Qom, Iran
- Yaghoobi, Kaebeh, Ale Taha Institute of Higher Education, Tehran, Iran

Table of Contents

Journal of Information Systems and Telecommunication
Vol.11, No.2, April-June 2023, 75-83
http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

Broadband Low-cost Reflectarray based on a New Phase Synthesis Technique and a Class of Cross Bow-Tie Cells

Mahmood Rafaei-Booket[1*], Mahdieh Bozorgi[1], Seyed Mostafa Mousavi[2]

[1].Department of Electrical and Computer Engineering, University of Zanjan, Zanjan, Iran
[2].Missouri University of Science and Technology, USA

Received: 30 Nov 2021/ Revised: 04 Jun 2022 / Accepted: 05 Jul 2022

Abstract

In this paper, a class of Bow-tie unit cell on FR4 substrate is designed and investigated to be used in implementing a single-layer broadband Reflectarray Antenna (RA). In the analyzing step, two different parameters (length and angle) of the grounded Cross Bow-Tie (CBT) are varied to obtain the phase diagrams. Various degrees of freedom in the CBT structure are very helpful in designing a broadband RA. In the antenna design procedure, an efficient phase synthesis technique is applied to minimize the adverse effects of frequency dispersion causing by the differential space phase delay at different frequencies. This technique optimizes the metallic CBTs arrangement on the aperture of RA, and reduces the dependency of RA design to the CBT's phase variation. Consequently, combination of the CBT's phase behavior and the phase synthesis technique leads to designing a broadband RA with a good frequency response. In addition to, the Side Lobe Level (SLL) of the resultant RA is reduced remarkably. For validation of the obtained numerical results, an RA is designed and fabricated in 8.7~12.3GHz frequency bandwidth. The measurements show 27.03 dB as a maximum gain value at 10.2 GHz with a 1.5dB gain bandwidth of 34%. It is also shown that the implemented RA exhibits a reduced SLL (<-18dBi) within its operating bandwidth.

Keywords: Reflectarray; Cross Bow-Tie unit cell; Broadband Antenna; Phase Synthesis Method; Low-cost structure.

1- Introduction

The Reflectarray Antennas (RAs) are a suitable alternative for phased array antennas, and conventional reflectors in military applications, space telecommunication, and remote sensing. The considerable deployment RAs is due to their exceptional features such as lightness, compactness, and flexibility of controlling radiation pattern. In addition, the use of RAs eliminates the need for expensive and complex custom molds, making RAs affordable [1]. Meanwhile, the narrow bandwidth of RAs is their main disadvantage. The narrow bandwidth of their unit cell and differential spatial phase delay lead to achieving their narrow bandwidth performance [1-3]. For solving this problem, wideband unit cells are designed with the help of periodic substrate [4], stacked patches [5], anisotropic substrates [6-9], and complex unit cells [10, 11]. Although the use of multilayer unit cells can increase the phase ranges and operating bandwidth of RAs [12], on the other hand, the weight, loss, complexity, and manufacturing cost of RAs increase. To address this deficiency, two different solutions have been described in the literature. The first solution is based on designing a wide phase range unit cell that is performed by a trial-and-error technique. Such a technique is a time-consuming and tedious one and leads to complex and multilayer unit cells [10-12]. The alternative solution is to use the optimization technique described in [13]. This optimization technique minimizes the phase realization errors of each element at center and two extreme frequencies within working bandwidth. Although the latter technique is beneficial, the various metallic elements' error function values in the array do not have a clear relationship in the optimization procedure. This results in local minima, which in turn results in sub-optimal frequency performance of the antenna.

To design a broadband RA, this paper first presents a method for minimizing the adverse effects of the frequency dispersion. This method is a new phase synthesis technique to achieve an appropriate arrangement

✉ **Mahmood Rafaei-Booket**
Booket@znu.ac.ir

of elements on RA's aperture and thereby a broadband RA. Then, a single-layer unit cell consisting of sub-wavelength metal Cross Bow-Tie (CBT) elements on a grounded low-cost substrate is proposed. The low-cost substrate is considered to be thick FR4. Using such a thick lossy substrate needs a tradeoff between reducing the insertion loss of the unit cell and optimizing the reflected power [14, 15]. Simulation results show that the reflectivity of the proposed class of CBT unit cell is so remarkable. Moreover, additional freedom degree of such a unit cell in comparison to cross-dipole [4] increases the phase range, which is needed to design broadband RAs. The obtained reflection information of the under-study unit cell is then applied for designing a planar center-fed RA. The design by means of the mentioned phase synthesis technique leads to obtaining the most appropriate metallic CBTs arrangement on the aperture of RA. As a result, the operation bandwidth of the designed single-layer RA is increased and Side Lobe Level (|SLL|) of its radiation pattern is reduced in comparison to the described traditional method in [13]. Such a technique is thoroughly examined with experimental demonstration and the designed center-fed RA is fabricated and measured in X-band (8.7~12.3GHz).

2- A Technique for Selecting Optimum Arrangement of RA Design

In RA designing, the proper phase distribution on its aperture is the main step such a way that the realized phases have a minimum deviation from the calculated phase distribution in the working frequencies. Based on [1], for designing an RA with a single pencil-beam in (θ_b, φ_b), the required phase shift of the i^{th} element located in (x_i, y_i) point is given by:

$$\phi_R = k_0\left(d_i - \left(x_i \cos\varphi_b + y_i \sin\varphi_b\right)\sin\theta_b\right) \quad (1)$$

that, the length of path from horn antenna's phase center to i^{th} element on the array indicated by d_i. obviously, satisfying the relation (1) is quite easy for RA design in a single frequency. However, it becomes more difficult to meet this requirement in each frequency sample once the working bandwidth of RA is increased, and the designer is faced with more phase realization errors. Such errors are mainly due to the differential spatial phase delay factor as mentioned in [1]. To control the behavior of the antenna in the desired bandwidth, an optimization technique is proposed in [13] and an optimal element arrangement on the aperture of RA is achieved by minimizing the error function defined as follows:

$$e(m,n) = \sum_{i=l,c,u} \left|\Phi^{\text{desired}}(f_i)(m,n) - \Phi^{\text{achieved}}(f_i)(m,n)\right| \quad (2)$$

Where (m,n) refer to a particular location on RA lattice and $\Phi_{f_i}^{\text{desired}}(m,n)$ and $\Phi_{f_i}^{\text{achieved}}(m,n)$ represent desired and achievable phase delays at given frequencies (l, c, and u represent lower, center, and upper frequencies, respectively). This procedure first calculates the phase distributions that meet the requirements of the mask in the center and the two extreme frequencies. In the second step, the most appropriate element is selected to minimize the error function at each element of RA lattice regardless of the other elements. The lack of a relationship between error function values in [13] leads to a local minimum in the optimization procedure. To explain further, in this method, the value of the error function for one element may be close to zero, while the value for the neighboring element may be large, resulting in suboptimal antenna performance. Solving this problem, an optimization technique is used that minimizes the average of the error function values calculated for all RA elements [8].

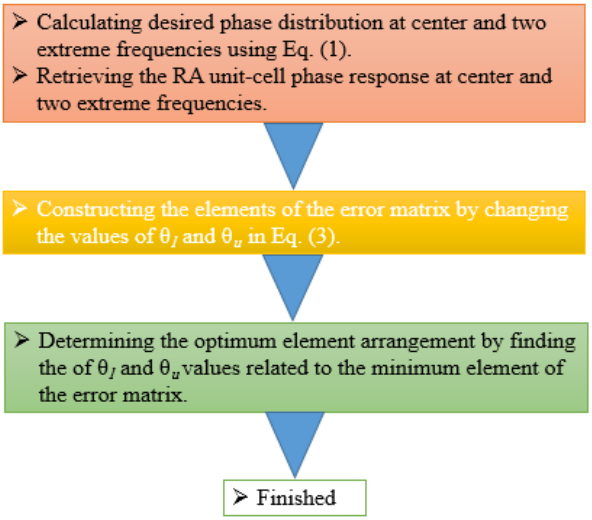


Fig. 1 Block chart of the introduced phase synthesis method.

For more explanation our phase synthesis method, suppose that the "$[\Phi]$" is the calculated phase shifts on the aperture of RA and "θ" is an arbitrary and a constant phase. In this case, "$[\Phi]+\theta$" leads to the identical radiation pattern. This property helps the designer to have freedom degrees in phase shifts distribution realizing with a minimum error and thereby attain an RA with optimum performance. To this end, the total phase error of the entire array is taken into account and the mentioned error function in (2) is modified as follows:

$$e(\theta_l,\theta_u)=\min\left\{\sum_{m,n}\left[\begin{array}{l}\left|\Phi_{f_l}^{\text{desired}}(m,n)-\Phi_{f_l}^{\text{achieved}}(m,n)+\theta_l\right|+\\ \left|\Phi_{f_c}^{\text{desired}}(m,n)-\Phi_{f_c}^{\text{achieved}}(m,n)\right|+\\ \left|\Phi_{f_u}^{\text{desired}}(m,n)-\Phi_{f_u}^{\text{achieved}}(m,n)+\theta_u\right|\end{array}\right]\times w(m,n)\right\} \quad (3)$$

for all members of the search space

and for $\theta_l=-180:\text{step}:180$ and $\theta_u=-180:\text{step}:180$

The relation of (13) is an objective function that needs to be calculated for database of different RA element's classes. It should be noted that the smaller the step in (3), the smaller the error, but a step of 1 degree or less does not significantly affect the antenna performance. On the other hand, the second principle is incorporated into (3) as a weighting function, $w(m,n)$, where elements closer to the center of the RA surface are given higher weights and elements closer to the RA edge are given lower weights. Thus, $e(\theta_l,\theta_u)$ is the best arrangement of RA cells minimizing (2). The structural information of the optimal elements of the RA is then automatically determined by selecting the arrangement that minimizes the error matrix.

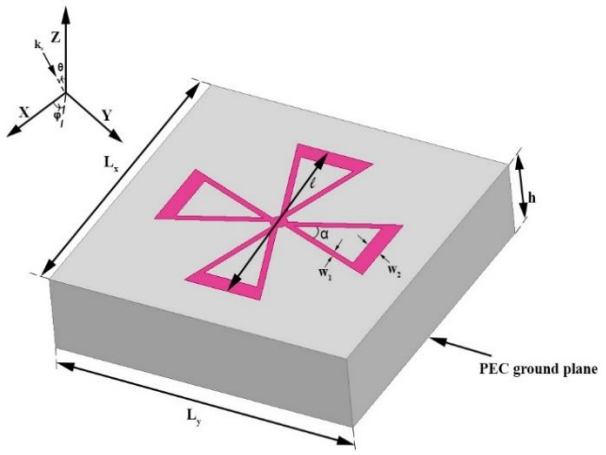


Fig. 2 Geometry of grounded CBT unit cell with FR4 substrate used for planar RA design.

Fig. 1 is a block diagram summarizing the phase synthesis technique introduced.

3- Element Design

Herein, the CBT unit cell is proposed inspiring from the bow-tie antenna structure. Its sketch and design parameters have been shown in Fig. 2. As shown in this figure, the reflected wave of CBT is affected by changing its length (l) and angle (α). In this case, the metallic array is printed on a grounded FR4 substrate with ε_r=4.4 and loss tangent δ_c=0.02 as a low-cost dielectric. In [15], analyzing a single-layer unit cell with FR4 dielectric, it was found that the utilizing of a thick lossy dielectric (FR4) results in low loss in respect to a thin one. Thus, the thickness of FR4 substrate has been chosen h=2.4mm. In addition, the period of lattice is considered to be smaller than $\lambda/2$ for designing a wideband RA [16]. The use of unit cells with smaller sizes leads to lower sensitivity to the angle of incident wave and improves the assumption of the infinite periodic structure. Meanwhile, (l /L_x) must be selected to achieve sufficient linear phase range for an RA designing. In this regard, L_x=L_y=0.35λ in the central frequency 10.5GHz, and 0.2<l/L_x<1 means the elements are sub-wavelength to achieve a maximum linear phase range [17]. The other parameters of CBT unit cell are optimized as: L_x=L_y=10mm, W_1=0.3mm, and W_2=0.9mm. In order to analyze the proposed unit cell and investigate the effects of the lossy substrate on the reflection characteristics, a common practice is adopted, where the element is assumed to be surrounded by an infinite periodic array of the same cells [1]. Such a simplification is called the local periodicity assumption which is validated by good agreement between theoretical RA design and its measurements [4].

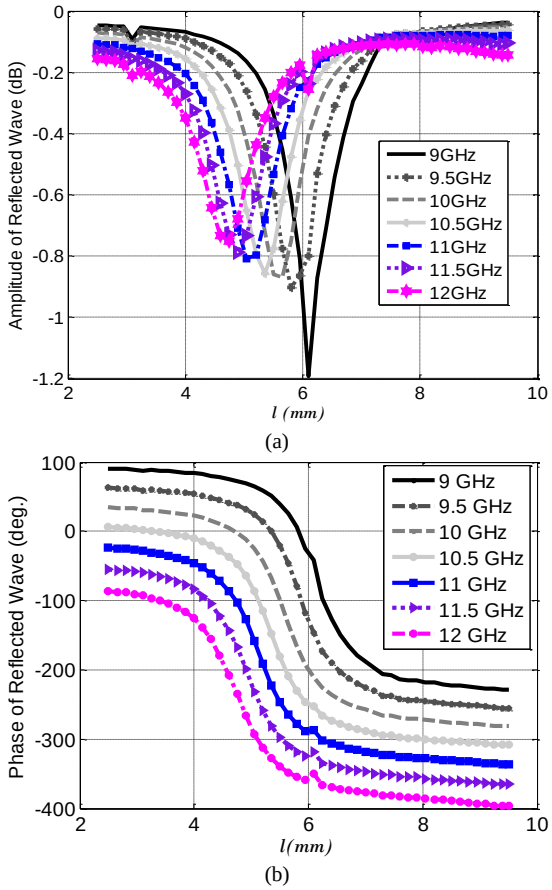
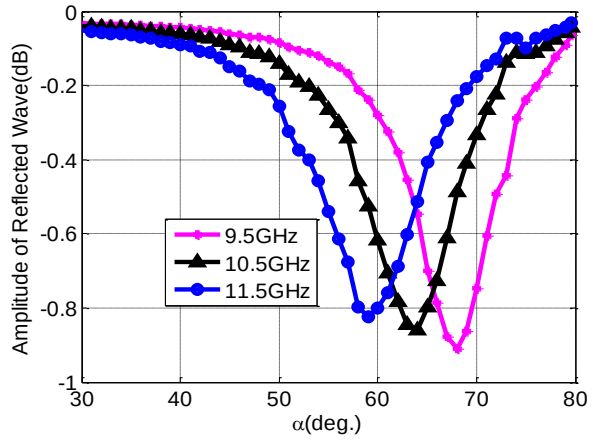


Fig. 3 Amplitude, and (b) phase diagrams of CBT unit cell versus of l variations (α=30°).

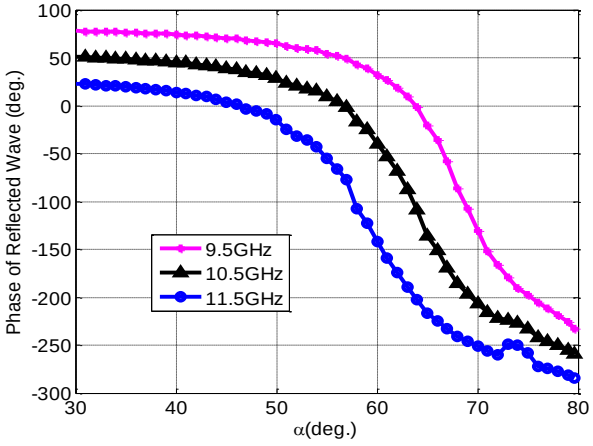
Table 1: Parameters of RA element for different classes

W_1 (mm)	W_2 (mm)	α(degree)	*l*(mm)
0.1~0.5	0.1~0.9	30°~80°	2.4-9.6
Step: 0.05	Step: 0.05	Step: 1°	Step: 0.2

Therefore, the full-wave analysis of the proposed CBT with periodic boundary conditions is carried out by means of a Finite Element Method (FEM)-based EM-solver. The reflection information of such a unit cell illuminated by a normally plane-wave is calculated in 9~12GHz frequency range, as shown in Fig. 3. The amplitude diagram is shown in Fig. 3(a) to estimate the unit cell loss. In Fig. 3(b), the phase responses are obtained versus of the element's length (*l*) variations. In these cases, the angle α=30°. To increase the phase range, the element length is left unchanged in its maximum value (*l* = 9.6mm) and the phase response is recalculated by means of the angle (α) variations from 30° to 80°. This range for angle α is allowable due to CBT's geometry.



(a)



(b)

Fig. 4 Amplitude, and (b) phase diagrams of CBT unit cell versus of α variations (*l*=9.6mm).

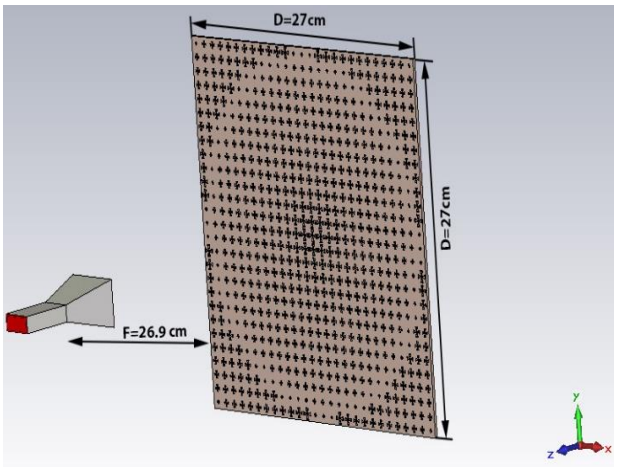
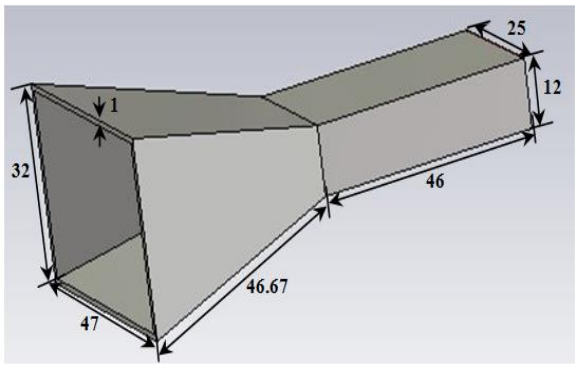
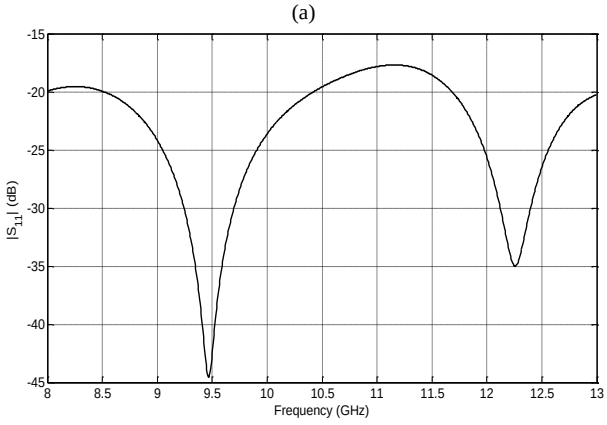


Fig. 5 Simulated RA and its feed antenna.

The reflected wave information versus variation of the angle α is visualized in Fig. 4. As shown in Fig. 4(b), another phase response with an approximately linear phase response is obtained in 9~12GHz.



(a)



(b)

Fig. 6 (a) Simulated horn antenna and its dimensions vs. mm, and (b) its impedance matching curve [25].

This demonstrates that the proposed element has an additional freedom degree to obtain a wider phase range

than cross-dipole element [4]. Fig. 3(a) and Fig. 4(a) show that, the average reflectivity of the unit cell with thick lossy substrate is close to 0dB for operating frequencies. This leads to the selection of the best values for α and l to reduce the sensitivity of the phase diagram in RA implementation. Various unit cell's classes have been obtained by varying W_1, W_2, l and α in Fig. 1. The different structural variations to make the phase response database are indicated in Table 1. The obtained phase and amplitude diagrams are used in the following to design a center-fed RA by means of the presented phase synthesis technique in block diagram of Fig. 1.

4- Optimum Broadband RA Design and Experimental Verification

In order to justify the effectiveness and practicality of the described optimization method, and proposed unit cell in Sec. 2, and Sec. 3, respectively, a low-cost single-layer RA is designed and manufactured in X-band having a very wide bandwidth. In this case, the calculated phase diagrams of the unit cell in Fig. 3(b) and Fig. 4(b) are used. In order to find the best performing element configuration at the RA frequency using the introduced phase synthesis technique, a database containing the unit cells' phase responses for all permutations was separately generated at the center (10.5 GHz) and at the two extreme frequencies (9GHz, and 12GHz). In these frequencies, the needed phase shift for each cell of RA aperture must be evaluated. To this end, one can use the method described in [1].

The corresponding equivalent elements' dimensions (l and α) are thus determined from the resultant phase diagrams (Fig. 3(b), and Fig. 4(b)) with the minimum error. In this regard, the center-fed RA depicted in Fig. 5 is simulated. As it is shown in the same figure, the planar array includes 27×27 CBT elements illuminated by a horn antenna. Since in center-fed RAs, feed blockage has its own role in the reducing the system efficiency, we have designed a horn antenna with small aperture size ($3.2\times4.7cm^2$) to decrease the effect of feed blockage. Its dimensions and schematic of the simulated horn are depicted in Fig. 6(a). Its return loss diagram is shown in Fig. 6(b) and indicates that the horn antenna works within the desired RA bandwidth. Moreover, the horn antenna's radiation patterns in H- and E-planes are respectively visualized in Fig. 7(a), and Fig. 7(b). In these patterns, Half Power Beam Width (HPBW) of the designed horn is 41^o for H-plane and 46.5^o for E-plane patterns, respectively. Its peak gain value is 13.3dBi at 10.5GHz. Thus, the exponent of feed pattern function represented by $\cos^q\theta$ is q=4.72, which is used in the RA design. Since the dimensions of center-fed RA is $27\times27cm^2$, the focal length (horn antenna's phase center) is calculated 26.9cm to maximize the antenna efficiency. According to the calculated phase diagrams, the required phase shifts of 27×27 elements of this center-fed RA are calculated and shown in Fig. 8. For such an RA, the half of the subtended angle from the aperture of RA to the horn antenna's phase center (θ_e) is 27.7^o. The phase diagram is calculated assuming that a normal incident plane wave irradiates all the RA's elements, but it is clear that such an assumption will reduce the overall gain and decrease the antenna efficiency. Therefore, the angular dependency of the phase diagrams is also taking into account. For this purpose, the phase diagrams in terms of $l(\alpha=30^o)$, and $\alpha(l=9.6mm)$ are recalculated for different oblique incident angles in the central frequency. As shown in Fig. 9, the little dependences of phase variations on the incident angle are observed once the incident θ is less than θ_e. Additionally, for avoiding the grating lobe in the radiation pattern, the element spacing in RA (L_x, L_y) must follow the array equation [18]:

$$L_x, L_y \le \frac{\lambda_0}{(1+\sin\theta_{inc})} \quad (4)$$

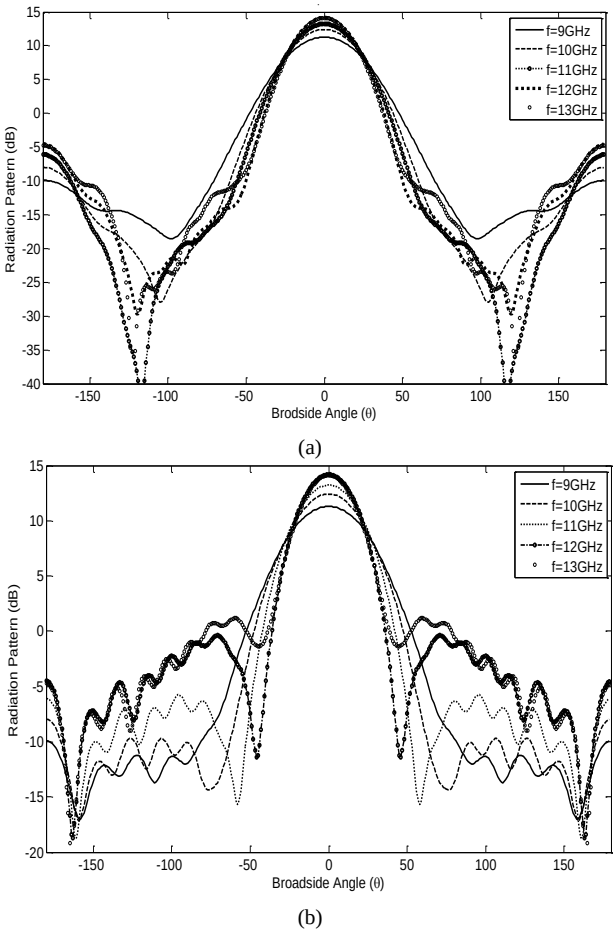


Fig. 7 Radiation patterns of the simulated horn antenna in (a) H-, (b) E-planes in frequency range 9~13GHz.

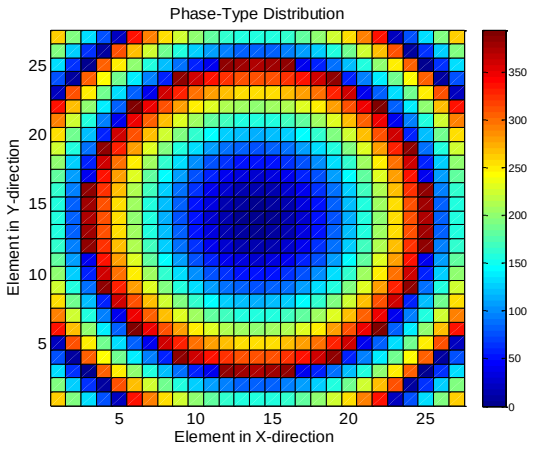


Fig. 8 The elements' phase shifts required in designing a center-fed RA with 27×27cells.

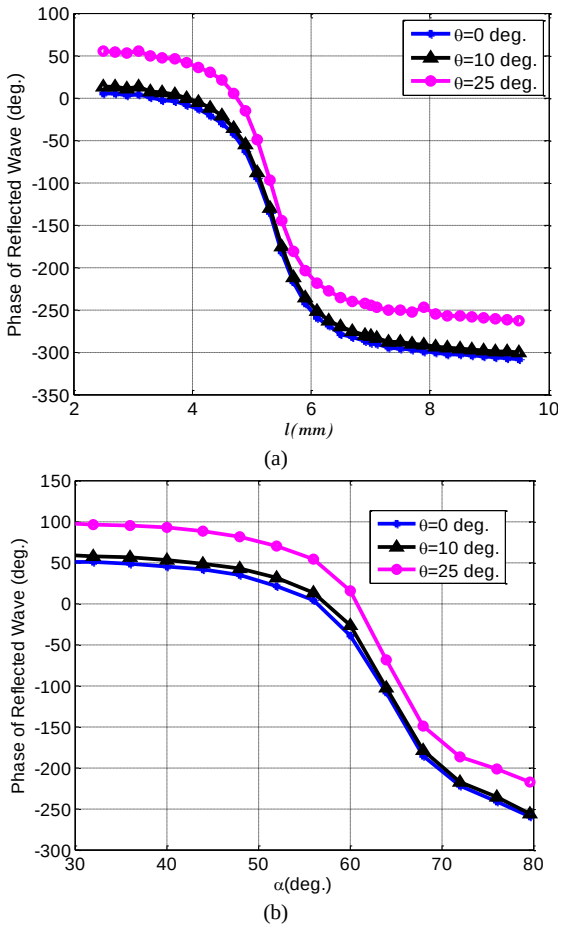


Fig. 9 Phase variations of the proposed CBT at central frequency 10.5GHz for various oblique incidences (θ_{inc} is the incident angle) when (a) l, and (b) α are changed.

in which, λ_0 is the free-space wavelength, and the incident angle θ_{inc} is less than θ_e. The unit cell's dimensions satisfy limitation of (4) due to $L_x = L_y$ =10mm=0.35λ_0. For really assessment of the introduced phase synthesis technique, the analytically calculated radiation patterns in H- and E-planes at 10.5GHz are respectively visualized in Fig. 10(a), and Fig. 10(b). In these figures, the radiation patterns calculated using the common approach [13] are observed. These figures show that our phase realization technique allows a lower SLL than the common one. Choosing the proper elements and reducing the error of RA's phase synthesis leads to the mentioned SLL reduction. It should be pointed out that the average error of the phase realization is decreased from 40^o in [13] to 5^o on each RA's element. The antenna structure including RA surface along with its horn antenna is simulated using an EM-solver and its surface current distribution is obtained at 10.75GHz, as shown in Fig. 11.

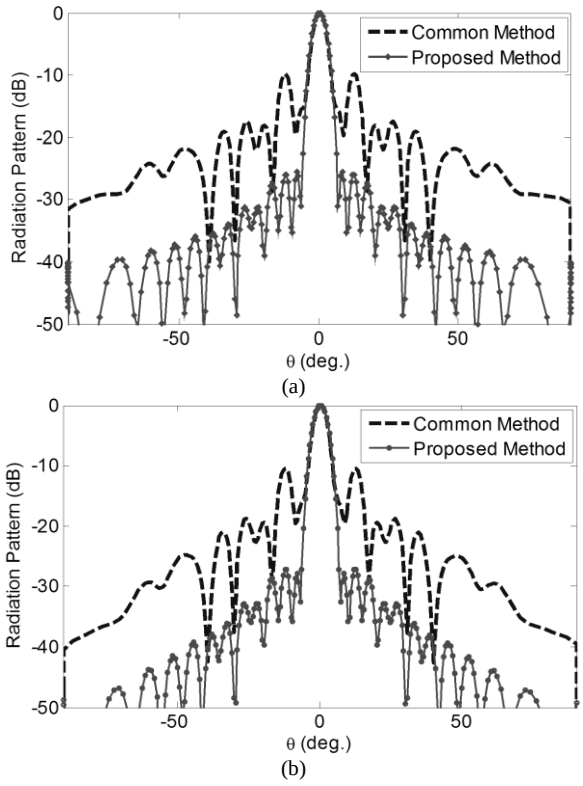


Fig. 10 Analytical radiation patterns at 10.5GHz, (a) H-plane, and (b) E-plane which are obtained by the RA aperture's phase distribution realized using the CBTs. The calculated radiation patterns by means of traditional method [13] are reported for comparison.

To validate the obtained numerical results, the designed system is fabricated. The photo of this low-cost implemented RA on FR4 substrate is observed in Fig. 12. The simulated radiation patterns and measured ones in E-, and H-plane at 10.2GHz are shown in Fig. 13(a), and 13(b), respectively. Measured Cross-polar radiation patterns are indicated by red dashed-lines in these figures.

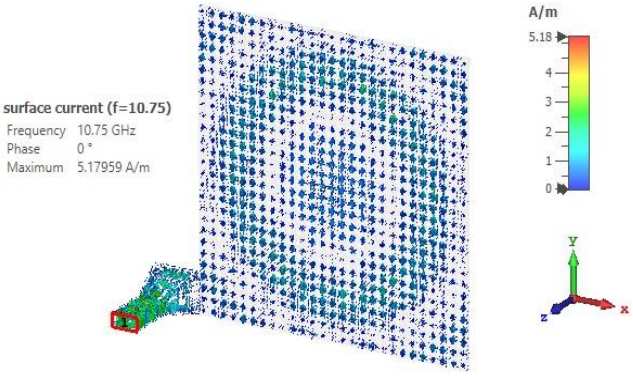
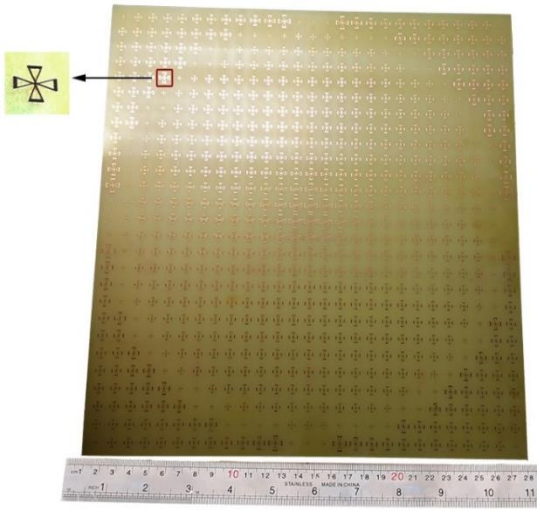
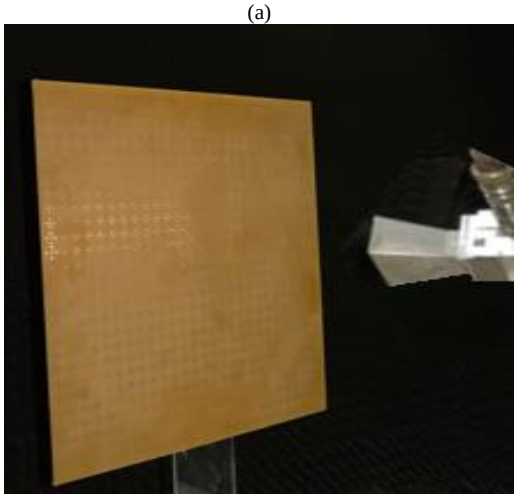


Fig. 11 Simulated current distribution of the designed RA.



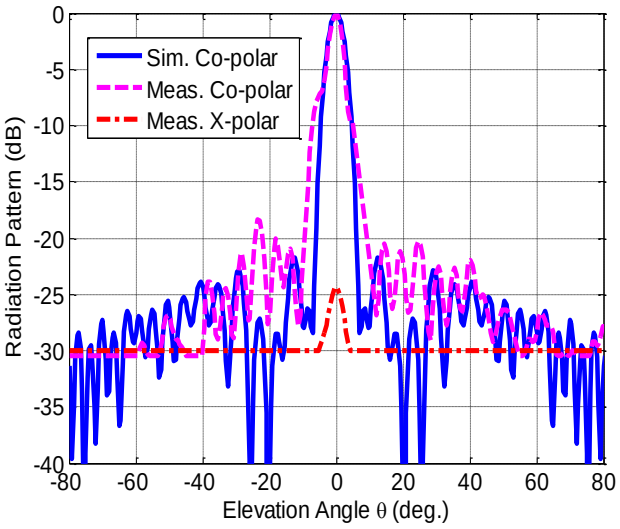
(a)



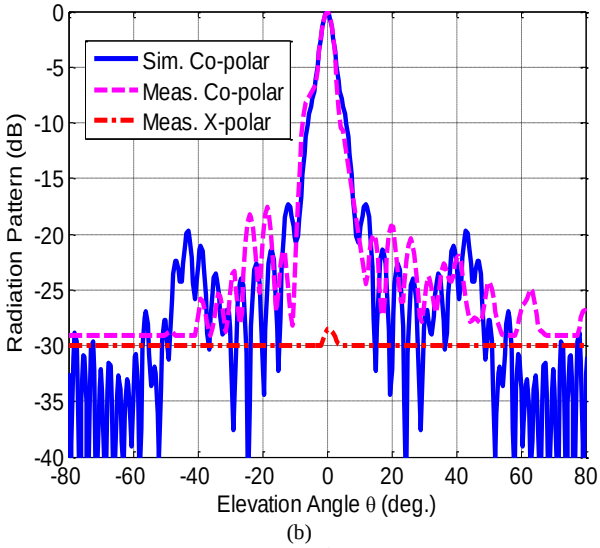
(b)

Fig. 12 Fabricated RA, (a) its photo from up view, and (b) RA along with horn antenna in anechoic chamber.

Also, the measured |SLL|<-18dB. Fig. 14 demonstrates the simulated and measured peak gain values versus frequency. As seen in the same figure, the maximum gain of the implemented RA is 27.03dB at 10.2GHz and its 1.5dB bandwidth of about 34% (8.7~12.3GHz) is achieved. Note that the goal of this study is not to design a high gain RA. However, a phase realization technique is introduced to broaden the RA's bandwidth and reduce its |SLL|. Obviously, if the number of array's cells and dimension of RA aperture are increased, the gain of the resultant structure can be enhanced. In some reported single-layer RAs like that [22], an air layer was added between dielectric and ground layer for gain enhancement.



(a)



(b)

Fig. 13 Normalized radiation patterns of the implemented RA at 10.2GHz in (a) E-plane, and (b) H-plane.

The comparison results of Table 2 confirm that a satisfactorily trade-off between bandwidth and |SLL| is obtained by combination of our optimization technique and proposed CBT unit cell. The maximum efficiency for such a system can be given by:

$$\eta_{total} = \lambda_0^2 \frac{G_{\max}}{4\pi A} \times 100, \quad (5)$$

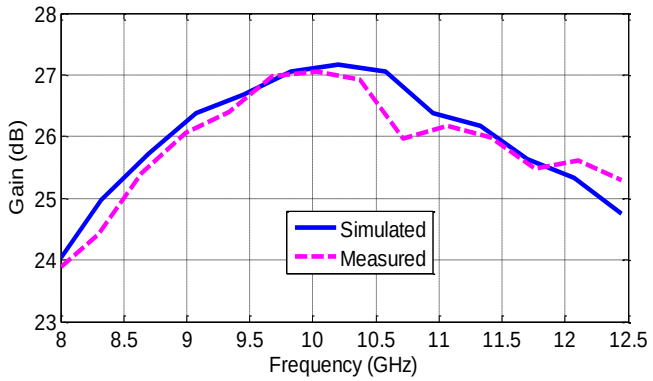


Fig. 13 Measured and simulated peak gain values of the implemented RA.

TABLE 2: Comparison of the Single-layer RAs

Ref.	Dim. of array (cm)	Cells No.	Freq. range (GHz)	G_{max} (dB)@ f_c	BW (%)	Max. phase range (deg.)	|SLL| (dB) @ f_c
[19]	40 ×40	1225	12.5 ~14.5	34.0	16.7	360	<-14
[20]	27×19	650	10.7 ~12.5	25.3	9.7	480	<-14
[21]	25×25	625	10.0 ~11.2	25.4	12	700	<-12
[13]	120×120	10000	11.4 ~12.8	40.6	11.6	400	<-16
[4]	27×27	729	8.95 ~12.1	26.57	29.5	400	<-14
[22]	Dia.=30	988	8.7 ~10.7	27.7	25	450	<-15
This Work	**27×27**	**729**	**8.7 ~12.3**	**27.03**	**34**	**610**	**<-18**

where G_{max} is the measured maximum gain (= 27.03dB) at 10.2GHz, and A is the RA's aperture area (=0.27×0.27m^2). Thus, the total efficiency is obtained 48%.

In Table 3, its estimated efficiencies and losses are summarized. As shown in the same table, the blockage of horn antenna is in the order of unit cell loss. It should be noted that the feed loss does not include loss in the transmission line between the horn antenna and transceiver. Therefore, one can obviously use an offset-fed design for increasing the RA efficiency.

TABLE 3: Estimated efficiency of the implemented low-cost broadband RA

Type	Efficiency (%)	Loss (dB)
Illumination	85	0.7
Spillover	72	1.42
Unit cell	91	0.41
Cross-pol [23]	95	0.22
Feed antenna [24]	91	0.41
Total	48	3.16

5- Conclusions

A metal cross bow-tied array on a thick lossy grounded FR4 dielectric is presented to implement a broadband, low-cost and single-layer RA. It is shown that presented unit cell with bow-tie patch layer gives us more phase range due to its two different design parameters (its length and angle). The obtained phase diagrams of this unit cell are used for implementation of an RA with 27×27 elements in X-band. Measured results indicate the implemented RA provides a gain of 27.03dB with a variation of less than 1.5dB over a relative gain-bandwidth of 34% (8.7~12.3GHz) and |SLL|<-18dB at 10.2GHz, which is an excellent performance compared with other single layer RAs designed in X-band. This feature is the result of using a robust phase realization technique, taking wave incident angle on each RA element into account and benefiting from a large database of RA elements.

References

[1] J. Huang, and J. A. Encinar, Reflectarray Antennas, Hoboken. NJ: John Wiley & Sons, 2008.

[2] D. M. Pozar, "Bandwidth of reflectarrays,", Electron Lett., Vol. 39, No. 21, 2003, pp. 1490-1490.

[3] M. Rafaei-Booket, and Z. Atlasbaf, "New Ku-band reflectarray antenna by using anisotropic superstrate on an artificial magnetic conductor," International Journal of Microwave and Wireless Technologies, Vol. 9, 2016, pp. 831-841.

[4] M. Rafaei-Booket, Z. Atlasbaf, and M. Shahabadi, "Broadband reflectarray antenna on a periodically perforated substrate," IEEE Trans. Antennas Propag., Vol. 64, No. 8, 2016, pp. 3711-3717.

[5] J. A. Encinar, and J. A. Zornoza, "Three-layer printed Reflectarrays for contoured beam space applications," IEEE Trans. Antennas Propag., Vol. 52, No. 5, 2004, pp. 1138-1148.

[6] M. Rafaei-Booket, and Z. Atlasbaf, "Metallic grating embedded in an anisotropic slab for realization of a reflectarray antenna," In 23th Iranian Conference on Electrical Engineering (ICEE), 2015, pp. 10-14.

[7] M. Bozorgi, and M. Rafaei-Booket, "Metallic array on a biased Ferrite substrate as a reflectarray antenna," In 9th International Symposium on Telecommunications (IST), 2018, pp. 80-85.

[8] M. Rafaei-Booket, and S. M. Mousavi, "Efficient analysis method and design approach for broadband reflectarrays with

isotropic/-artificial anisotropic substrates," IET Microwaves, Antennas & Propag., Vol. 14, No. 10, 2020, pp. 1108-1116.
[9] X. Li, Y. Wan, J. Liu, D. Jiang, T. Bai, K. Zhu, J. Zhuang, and W.-Q. Wang, "Broadband electronically scanned reflectarray antenna with liquid crystals," IEEE Antennas and Wireless Lett., Vol. 20, No. 3, 2021, pp. 396-400.
[10] E. Carrasco, M. Barba, and J. A. Encinar, "Reflectarray element based on aperture-coupled patches with slots and lines of variable length," IEEE Trans. Antennas Propag., Vol. 52, No. 3, 2007, pp. 820-825.
[11] E. Carrasco, J. A. Encinar, and M. Barba, "Bandwidth improvement in large Reflectarrays by using true-time delay," IEEE Trans. Antennas Propag., Vol. 56, No. 8, 2008, pp. 2496-2503.
[12] E. Ozturk, and B. Saka, "Multilayer Minkowski reflectarray antenna with improved phase performance," IEEE Trans. Antennas Propag., 2021, DOI: 10.1109/TAP.2021.3090533.
[13] M. R. Chaharmir, J. Shaker, N. Gagnon, and D. Lee, "Design of broadband, single layer dual band large reflectarray using multi open loop elements," IEEE Trans. Antennas Propag., Vol. 58, No. 9, 2010, pp. 2875-2883.
[14] H. Rajagopalan, and Y. Rahmat-Samii, "On the reflection characteristics of a reflectarray element with low-loss and high-loss substrates," IEEE Trans. Antennas Propag., Vol. 52, No. 4, 2010, pp. 73-85.
[15] F. Costa, and A. Monorchio, "Closed-form analysis of reflection losses in microstrip reflectarray antennas," IEEE Trans. Antennas Propag., Vol. 60, No. 10, 2012, pp. 4650-4660.
[16] J. Ethier, M. R. Chaharmir, and J. Shaker, "Loss reduction in reflectarray designs using sub-wavelength coupled-resonant elements," IEEE Trans. Antennas Propag., Vol. 60, No. 11, 2012, pp. 5456-5459.
[17] P. Nayeri, F. Yang, and A. Z. Elsherbani, "Braodband reflectarray antennas using double-layer subwavelength patch elements," IEEE Antennas Propag. Lett., Vol. 9, 2010, pp. 1139-1142.
[18] M. Bozorgi, and Z. Atlasbaf, "Spectral solution for scattering analysis of periodic plasmonic nano-antennas on iso/-anisotropic substrate," J. Lightwave Technol., Vol. 34, No. 11, 2016, pp. 2624-2630.
[19] Y. Mao, Sh. Xu, F. Yang, and A. Z. Elsherbani, "A novel phase synthesis approach for wideband reflectarray design," IEEE Trans. Antennas Propag., Vol. 63, No. 9, 2015, pp. 4189-4193.
[20] H. Hasani, M. Kamyab, and A. Mirkamali, "Low cross-polarization reflectarray antenna," IEEE Trans. Antennas Propag., Vol. 59, No. 5, 2011, pp. 1752-1756.
[21] M. Maddahali, and K. Forooraghi, "High efficiency reflectarray using smooth tapering in phase pattern on antenna surface," Microwave and Optical Lett.,Vol. 55, No. 4, 2013, pp. 747-753.
[22] R. S. Malfajani, and B. A. Arand, "Dual-band orthogonally polarized single layer reflectarray antenna," IEEE Trans. Antennas Propag., Vol. 65, No. 11, 2017, pp. 6145-6150.
[23] J. Huang, "Analysis of a microstrip reflectarray antenna for micro- spacecraft application," TDA Progress Report, 1995.
[24] C. A. Balanis, Antenna Theory: Analysis and Design, 3rd ed. John Wiley & Sons; 2005.
[25] M. Rafaei-Booket, and M. Bozorgi, "Low-cost inhomogeneous material for low-RCS reflectarray antenna implementation," AEU- International Journal of Electronics and Communications, Vol. 149, 2022, pp. 154182.

Journal of Information Systems and Telecommunication
Vol.11, No.2, April-June 2023, 84-93
http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

Comparative Study of 5G Signal Attenuation Estimation Models

Md Anoarul Islam[1], Manabendra Maiti[1], Judhajit Sanyal[1*], Quazi Md Alfred[2]

[1].Department of Electronics and Communication Engineering, Techno International New Town
[2].Department of Electronics and Communication Engineering, Aliah University

Received: 05Dec 2021/ Revised: 04 May 2022 / Accepted: 08 Jun 2022

Abstract

Wireless networks functioning on 4G and 5G technology offer a plethora of options to users in terms of connectivity and multimedia content. However, such networks are prone to severe signal attenuation and noise in a number of scenarios. Significant research in recent years has consequently focused on establishment of robust and accurate attenuation models to estimate channel noise and subsequent signal loss. The identified challenge therefore is to identify or develop accurate computationally inexpensive models implementable on available hardware for generation of estimates with low error and validate the solutions experimentally. The present work surveys some of the most relevant recent work in this domain, with added emphasis on rain attenuation models and machine learning based approaches, and offers a perspective on the establishment of a suitable dynamic signal attenuation model for high-speed wireless communication in outdoor as well as indoor environments, presenting the performance evaluation of an autoregression-based machine learning model. Multiple versions of the model are compared on the basis of root mean square error (RMSE) for different orders of regression polynomials to find the best-fit solution. The accuracy of the technique proposed in the paper is then compared in terms of RMSE to corresponding moderate and high complexity machine learning techniques implementing adaptive spline regression and artificial neural networks respectively. The proposed method is found to be quite accurate with low complexity, allowing the method to be practically applicable in multiple scenarios.

Keywords: 5G; Estimation; Attenuation Models; Machine learning; Dynamic Model; Autoregression.

1- Introduction

The present era has seen rapid advancement in the field of wireless communication technology, with extremely high data rates allowing users access to high quality multimedia content as well as streaming media services. In particular, the advent of 5G technology presents hitherto-unseen possibilities in the domain of wireless communication services. In such a scenario, reliability and Quality-of-Service (QoS) are two critical parameters that must be at acceptable levels to ensure user satisfaction.

The viability of 5G wireless communications was established in a seminal paper presenting different facets of millimeter wave wireless communication technology [1]. Corresponding attenuation models for 5G wireless communication signals have been reviewed in literature in recent times [2][3]. Diverse 5G-based applications have been proposed and established, in domains as diverse as agriculture [2] and security [4]. In all cases, the establishment and use of accurate attenuation models are critical to the success of proposed schemes. Other recent research establishes the impact of rain on channel noise and signal attenuation [5][6]. The present paper reviews recent techniques and models employed for estimation of rain attenuation, with focus on machine learning based approaches. The major contributions of the paper are as follows.

- Extensive review of recent literature documenting novel approaches to the problem of estimation of wireless communication (especially 5G) signal attenuation.
- Identification of novel solutions based on machine learning (ML) approaches, to improve model accuracy.
- Proposal of an adaptive autoregression-based estimation model to achieve suitably low root mean square error (RMSE) at low computational complexity, compared to other ML and non-ML-based techniques.
- Verification of effectiveness of proposed model through real-world experiments in low, moderate and high mobility scenarios.

The rest of the paper is organized in the following manner. Section 2 presents a survey of recent literature in the domain of attenuation modelling and estimation of wireless communication signals, with significant emphasis

✉ **Judhajit Sanyal**
judhajit.sanyal.2019@gmail.com

on machine learning based techniques employed. Section 3 presents the inferences drawn from the previous section and offers a basic conception of dynamic modelling which can help in improvement of the accuracy of estimation. Section 4 concludes the paper.

2- Literature Survey

The possibility of widespread commercial wireless communication in the millimeter wave bands has been explored some years back in a seminal work [1]. The challenges faced by wireless networks (up to 4G networks) have been extensively highlighted here, as well as the corresponding benefits offered by millimeter wave networks [1]. Following this paper and others in the same vein, extensive examination of attenuation models for 5G wireless signals have been carried out in recent years, due to noise and attenuation being critical limiting factors in the efficacy of modern wireless communication networks [2].

2-1- Attenuation Models

A number of established channel models have been examined in [2] in the context of their suitability for modelling the propagation of millimeter wave communication signals of a wide range of frequencies through free space. The authors in [2] have made extremely important contributions to the field by examining propagation of signals below 6 GHz as well as at 28 GHz and above, which are relevant for different classes of 5G communication networks. 5G D2D communications have been extensively surveyed in [3], and the benchmark measures for various aspects of such communication networks have been discussed here, inclusive of specific attenuation models for D2D networks. Heterogeneous access scenarios for 5G D2D networks have been examined in [4], with the models employed linking to the security aspects of such networks [4].

Rain attenuation is significant for millimeter wave signals, to the extent of around 7 dB per kilometer along the slant path in the 28 GHz band [2]. As a consequence, accurate estimation of the attenuation of communication signals affected by rain has been the focus of many researchers in recent times [5][6]. Terrestrial attenuation models in particular are specifically relevant for terrestrial 5G communication [5]. Rain statistics for K band (25.84 GHz) and E band (77.52 GHz) signals have been observed on a yearly basis to generate accurate estimates for short-range fixed links [6]. The work outlined in [6] allows for compensation of the wet antenna effect. The effect of rain is most pronounced in the tropical regions, as a consequence of which region-specific models are often used for achieving appreciable accuracy in such scenarios [7]. The effects of rainfall and knife-edge diffraction are examined in detail for fixed millimeter wave systems, in [8]. Attenuation models for multiple frequencies are often used in practical scenarios. However, in inclement weather conditions, such models may fail to generate accurate estimates and, in such cases, alternative means for generation of estimates can be considered [9]. Additionally, short-range terrestrial communication systems are affected by parameters such as the distance factor which affects the estimate of signal strength as well as overall link budget for both K (25 GHz) and E (75 GHz) bands [10]. In arid climates, dust storms may also severely hamper link capacity through random and anomalous diffraction of millimeter wave signals [11]. Recent researchers have also shown interest in the presently-unlicensed V band (60 GHz) for 5G backhaul networks, spanning both Line-of-Sight (LoS) and Non-LOS scenarios [12]. Significant variation in link performance is observed for both sub-6 GHz as well as 26/28 GHz bands in tropical locations, due to extensive rainfall [13]. Attenuation models for both outdoor [14] as well as indoor locations have been explored by researchers in recent times [15]. Indoor environments present an array of challenges due to diffraction as well as absorption by organic and inorganic obstacles, especially human body-based absorption [15]. The effects are significant at 28 GHz, as observed in [15]. A recent approach has also explored dynamic modelling for an indoor millimeter wave link (28-30 GHz), with favourable results [16].

Rain attenuation has also been studied by various researchers in recent years with the application of modified channel models such as the shadowed Rician fading model, which allows for accurate bit error rate (BER) and signal to noise ratio (SNR) estimation for satellite to land systems, and additionally identifies the effect of estimation error on the capacity of the satellite communication system under observation [17]. One recent significant work in the domain proposes a unified approach which allows for estimation of rain attenuation both on the slant or vertical path as well as the horizontal path [18]. Location specific attenuation models for slant path have also been proposed in [19] and [20], for specific tropical locations, which significantly outperform existing standard models such as ITU-R, Karasawa and DAH models, but may not be able to produce acceptably accurate results in other regions. Worst-month statistics are also a reflection of the accuracy of such models, and established models such as ITU-R have been found to come up short in regions with greater meteorological diversity, such as tropical locations [21]. In this context, one recent paper clearly delineates the characteristics of microwave and millimeter wave channels which allows the accurate characterization of such channels in a variety of environments [22]. Machine learning based approaches

have also emerged as important tools in this context, for different frequency bands [23][24].

Indoor attenuation models have also grown in importance from a commercial context during recent years. Consequently, researchers have sought to focus on multi-frequency model (for 14 and 22 GHz) establishment in indoor environments [25]. The significant 28 GHz band has also been examined using LoS and NLoS models as well as using X-band signal [26]. Path loss models are critical to the successful implementation of 5G communication systems, and researchers in [27] have extensively examined the comparative performance of different path loss models in estimation of path loss for sub-6 GHz 5G networks, with emphasis on both indoor as well as outdoor urban and industrial environments. Though generalized models for macroenvironments have been proposed in context of 5G networks, indoor performance of such models can be significantly improved [28]. One recent work however has achieved excellent results in terms of elevation of RAN bottlenecks in indoor and outdoor environments [29]. Another interesting work has illustrated the effects of low-emissivity glass on 5G signal in indoor environments [30].

2-2- Machine Learning Based Approaches

Among the different approaches employed by researchers to generate accurate attenuation predictions, machine learning techniques have increased in popularity. Supervised learning methods have been established to be effective in [5]. Such models are especially effective in scenarios where traditional models are unable to accurately predict signal attenuation [9]. Machine learning techniques can also be used in conjunction with different LoS and Non-LoS channel models to predict link performance in a wide range of network scenarios and mobility conditions [12][13]. They also allow integration of multiple factors, such as antenna geometry [14] and the generation of accurate estimates in dynamic network conditions with significant diffraction and attenuation [15][16]. Thus, such approaches can be effectively leveraged to increase the efficacy and robustness of a proposed attenuation model.

Among different machine learning based approaches, low complexity techniques have gained in popularity due to their ease of use and integrability into communication infrastructures at low costs. For example, spline-based machine learning approaches have been found to achieve greater accuracy than other regression-based methods in estimation of rain attenuation [23][24]. Similar techniques have also been applied to predict LoS and NLoS path losses in indoor environments with suitably low complexity and low root mean square error (RMSE) [26]. Other supervised machine learning techniques have also been found to yield appreciable accuracy in outdoor environments [5][9] as well as indoors [15][16][27] for different network configurations and scenarios. The techniques outlined in [15] and [16] are especially important considering the fact that consistently accurate estimates have often been proved to be difficult to generate in dynamic indoor environments which often allow more complex propagation scenarios to exist compared to corresponding outdoor implementation environments. Unsurprisingly, therefore, most proposed techniques and models are found to perform better in one or a few types of scenarios and locations. Other popular methods include ray-tracing for establishment and testing of multi-frequency indoor and outdoor models [31]. Wall correction factor-based modelling has also borne fruitful results for researchers, since such structures may lead to diverse fading scenarios and can significantly affect the estimates generated by models not compensating for fading and signal attenuation events engendered by such indoor and outdoor structures [32]. However, outdoor models are vulnerable to inclement meteorological phenomena, but otherwise achieve higher network performance [33]. Another innovative approach uses machine learning algorithms to offset environmental losses through accurate tracking of received signal strength [34]. Another recent paper has looked at machine learning based beam quality estimation for improvement of SNR through the application of deep neural networks [35]. Attenuation map-based positioning systems have been presented in [36] with the help of a deep learning architecture. A low-complexity pilot assignment algorithm presented in [37] allows the mitigation of channel state errors and noise for massive-MIMO systems. A graph-colouring based algorithm is presented for channel estimation in massive-MIMO D2D underlay systems for optimal pilot assignment, for improvement of parameters such as SNR [38]. A deep learning-based channel estimation approach is also presented in [39], for generation of estimates from received omni-beam patterns, in the context of vehicular communication. Another relevant recent work presents an ensemble prediction system for nowcasting of attenuation data for highly accurate prediction of attenuation events such as heavy rainfall [40]. Other ML-based techniques have also proved to yield excellent results in diverse fields such as in the health monitoring of electrical systems, employing a combination of the Continuous Wavelet Transform (CWT) and Convolutional Neural Network (CNN)-based approaches [41]. Residual Neural Network (ResNet)-based approaches have also yielded accurate results in sleep-stage detection through examination of EEG signals, and such approaches are expected to be effective in time-series based prediction of signal attenuation [42]. A feature extraction methodology based on fractal analysis can also be effective for implementing feature extraction for a given time-series [43], which is

another approach towards a high-accuracy solution for prediction of signal attenuation.

The survey of recent relevant literature indicates the presence of a significant research gap in terms of proposal of a suitable dynamic signal attenuation model for 5G communication with compensation for different meteorological events inclusive of rain, in different geographies around the world. The present work consequently stems from a need to address these issues in view of the great potential 5G communication possesses as well as the significant challenges posed to its effective implementation, which can be alleviated through the investigation of suitable models which can guarantee acceptably accurate performance in practical scenarios.

3- Proposed Model

On inspection of the recent research in the establishment of attenuation models for 5G millimeter wave communication signals, two major challenges are found to emerge. First, the attenuation of a 5G signal varies significantly due to meteorological phenomena such as rain, as well as the micro-environment of the network (for example, diffraction due to sand storms in arid regions). Second, attenuation models for indoor and outdoor environments are found to vary to a significant extent, and lack dynamicity to a certain degree.

In such a scenario, the application of suitable machine learning techniques allows for design and establishment of dynamic models which can adapt to changes in the network conditions and can therefore generate estimates with greater accuracy than typically used attenuation models. It is also to be kept in mind that adaptive machine learning based model are more robust in the face of significant meteorological variations, further increasing the usefulness of such models in the present context. As a consequence, a dynamic machine learning based model which accounts for both LoS and NLoS signal propagation would seem to be ideal in the given scenario.

The present work therefore proposes an autoregression-based machine learning model for generation of attenuation estimates which can be applied for real-time as well as non-real-time time series data. The basic system model is illustrated in the following Figure 1.

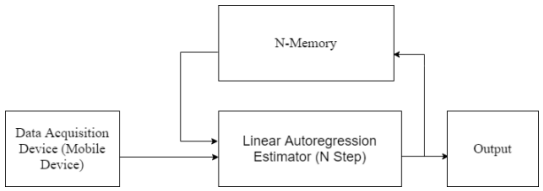


Fig. 1 N-Step Autoregression Estimator

The autoregression model generates regression estimates based on the previous *N* values of the signal attenuation and adjusts the weight of the values based on the estimation error. The estimation results are presented in the following section. The corresponding relations for autoregression estimation are expressed in equation 1 which follows, with *F* being the autoregression function, *Y* being the estimated output, *X* being the independent attenuation variable, *C* being the associated coefficient weight, and *t* being the instant of time.

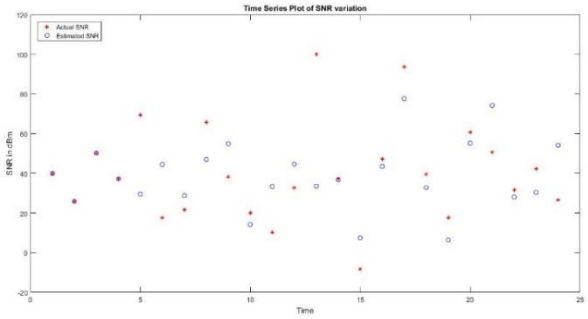
$$Y(t+1) = F_t\{\sum_1^k X_i^k(t)C_i(t)\} \quad (1)$$

The corresponding path loss attenuation model applied for simulation of the scenario is the standard 3GPP urban microenvironment model [2] with attenuation *X* being dependent on distance *d* as well as constants α, β and γ, as shown in equation 2.

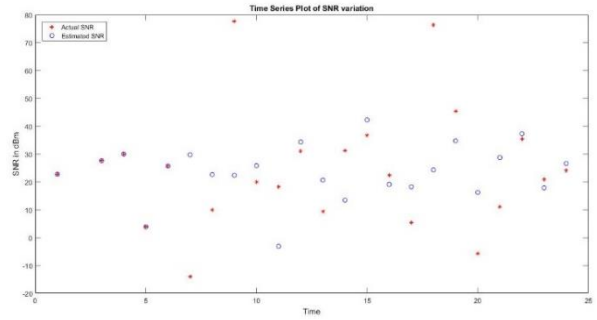
$$X = \alpha + \beta \log_{10} d + \gamma \quad (2)$$

4- Results and Conclusions

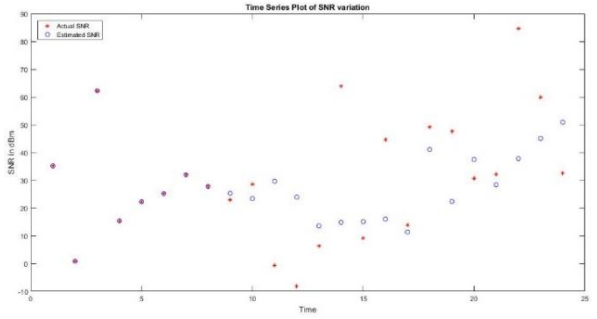
Five different types of ML based autoregression estimation techniques are used to generate attenuation estimates in this study. The techniques are: forward-backward technique, least-squares estimation, Yule-walker technique, Burg's Method and Geometric Lattice Technique. The individual results are obtained for statistical samples generated over 1000 test runs each, for polynomials of order 4, 6 and 8 respectively. The corresponding results are presented in Figures 2 to 6, which follow.



(a)

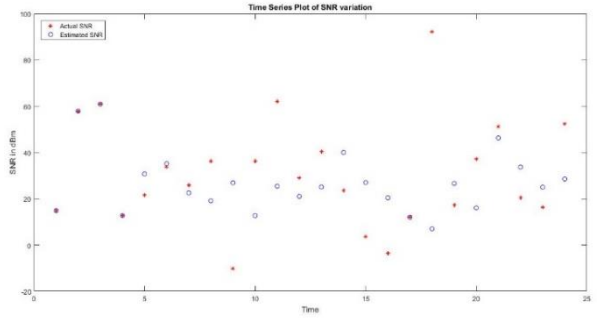


(b)

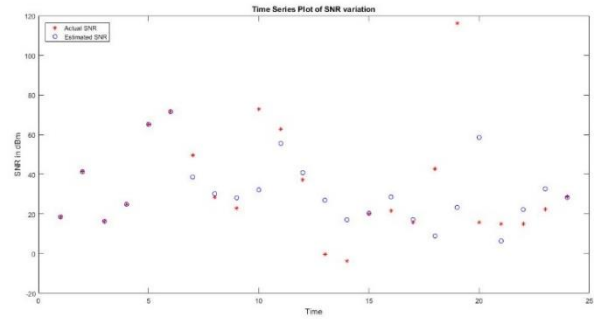


(c)

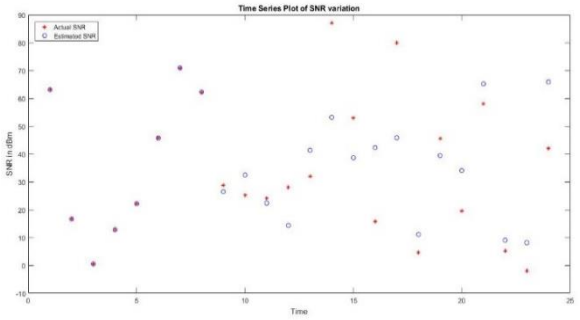
Fig. 2 Forward-Backward Method: (a) 4th order polynomial (b) 6th order polynomial (c) 8th order polynomial



(a)

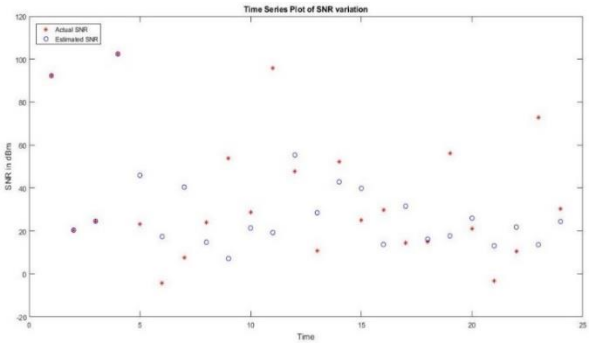


(b)

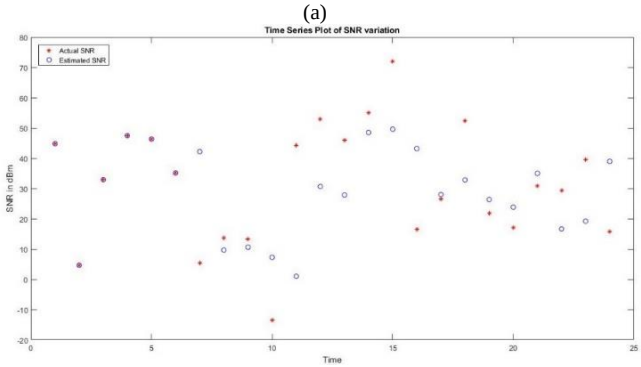


(c)

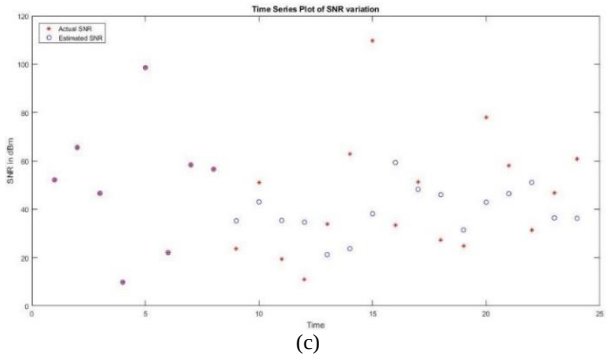
Fig. 3 Least-Squares Method: (a) 4th order polynomial (b) 6th order polynomial (c) 8th order polynomial



(a)



(b)



(c)

Fig. 4 Yule-Walker Method: (a) 4th order polynomial (b) 6th order polynomial (c) 8th order polynomial

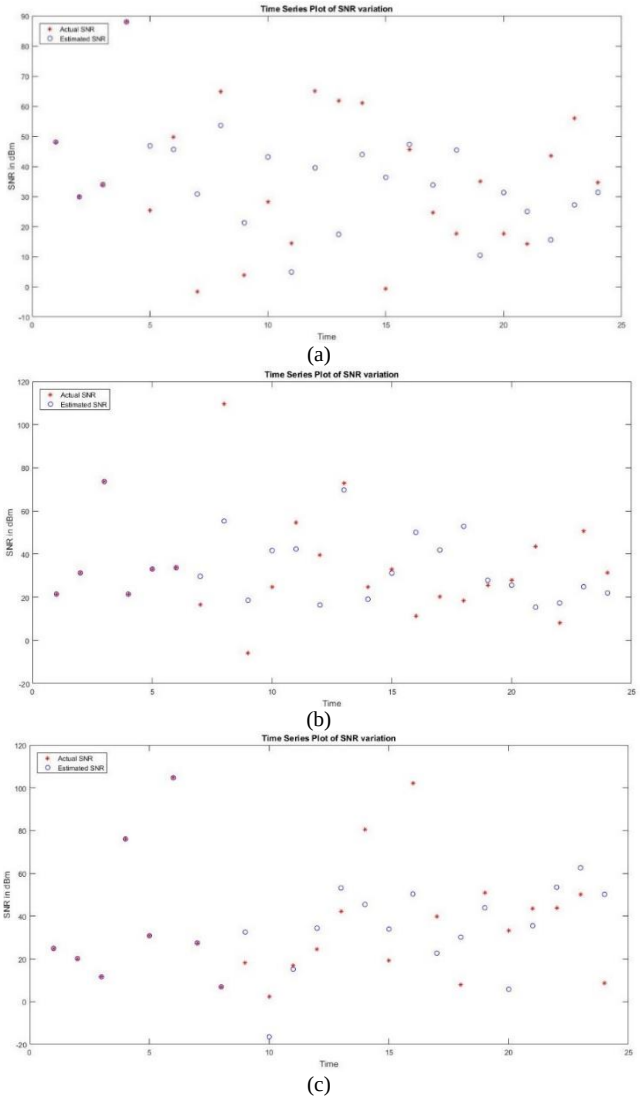
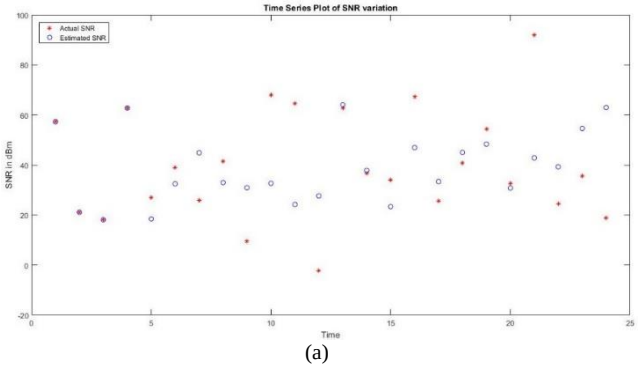


Fig. 5 Burg's Method: (a) 4th order polynomial (b) 6th order polynomial (c) 8th order polynomial



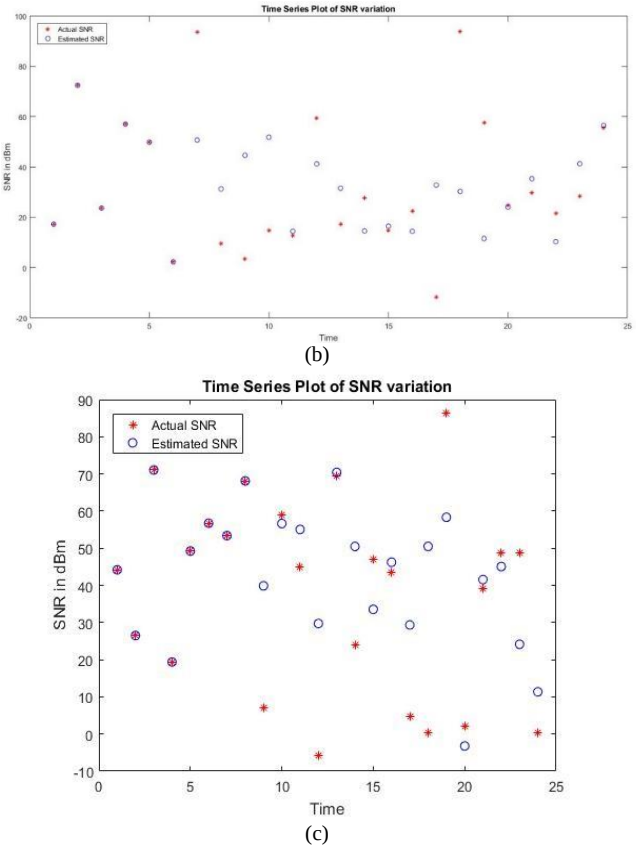


Fig. 6 Geometric Lattice Method: (a) 4th order polynomial (b) 6th order polynomial (c) 8th order polynomial

The RMSE or root mean square error metric is used to evaluate the accuracy of the estimated attenuation values in this work. The various techniques are therefore compared according to both the error as well as the RMSE metrics in the present work. The RMSE is measured due to the fact that the metric is statistically more significant than the mean error in terms of representation of the dataset, since it is less prone to bias than mean error metric. The corresponding values for each technique are presented in Table 1.

Table 1. Comparison of RMSE

Technique	***RMSE for Polynomial Order***		
	4	**6**	**8**
Forward-Backward	0.7758	1.2751	10.2541
Least-Squares	2.0739	20.9429	1.2008
Yule-Walker	1.8334	1.5183	0.5932
Burg's Method	12.4825	1.2436	2.1097
Geometric Lattice	2.8168	2.6201	39.9316

From Table 1, it is seen that even though the Forward-Backward or Burg methods may sometimes generate marginally more accurate estimates, they are prone to fluctuation of accuracy with polynomial order. The Yule-Walker method is optimal in the sense that it generates agreeably correct estimates even for higher order polynomial models and does not suddenly decrease in accuracy for any of the polynomials. The comparative RMSE performance of all techniques is shown in the following Figure 7, which validates the abovementioned inference.

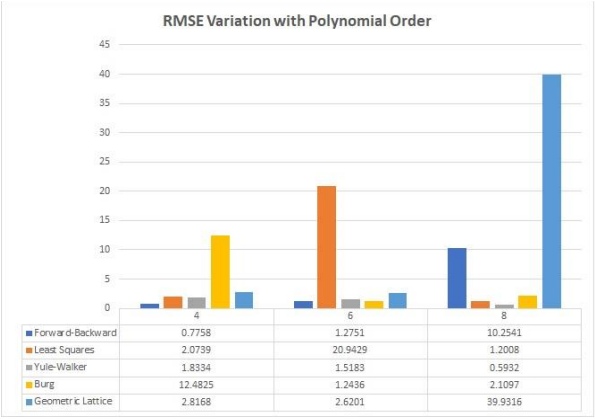


Fig. 7 Comparative RMSE Performance of Techniques

The autoregression-based machine learning technique is compared to corresponding artificial neural network (ANN) based method [39] and adaptive spline regression-based method [25]. The RMSE metric is used to compare between the three methods. The corresponding Table 2 presents the comparative results observed.

Table 2. Comparison of Autoregression, ANN and Adaptive Spline Regression Methods

Technique	***RMSE***	***Complexity***
Autoregression	0.5932	Low
ANN	0.0713	High
Adaptive Spline Regression	0.9811	Moderate

The complexity of the proposed method can be estimated in the following manner. Assuming the application of the Yule-Walker method, the power spectral density of the time series is repeatedly computed and adjusted using a small set of k samples (among a possible N samples). Assuming the worst-case convergence of the technique, that is, after N cycles, the worst-case time complexity of the proposed technique is $O(kN) << O(N^2)$, and considering the small set of samples k examined to generate estimates at each step, $k<<N$, which allows the worst-case complexity of the technique to be estimated as $O(N)$. This complexity is significantly lesser than the $O(N^2)$ complexity of the adaptive spline technique and the $O(N^r)$ for $r>2$ complexity of the ANN method.

The comparative results show that the autoregression based technique achieves suitably low RMSE at low complexity, compared to the other techniques, allowing this proposed technique to be easily implementable in practical scenarios. Another important fact that must be kept in mind is that the RMSE is not extremely low, which indicates that the model does not suffer from significant overfitting error, which in turn allows the proposed model to be more dynamic.

Next, the proposed technique is compared to the ANN and adaptive spline-based methods for different mobility scenarios, ranging from 1 m/sec (low mobility) to 30 m/sec (high mobility), with respect to the same RMSE parameter. In all cases, the transmitters are considered to be fixed while the devices receiving the signal are made mobile. The experiments are repeated to generate 1000 sets of results, which are then averaged to generate the results, for each of the techniques. The results achieved by the three techniques compared in the present work are illustrated in the following Table 3.

Table 3. Comparison of Autoregression, ANN and Adaptive Spline Regression Estimates for different Mobility Scenarios

Technique	***RMSE***	***Mobility***
Autoregression	0.61	Low (1m/sec)
ANN	0.09	Low (1m/sec)
Adaptive Spline Regression	0.93	Low (1m/sec)
Autoregression	0.69	Moderate (15m/sec)
ANN	0.11	Moderate (15m/sec)
Adaptive Spline Regression	1.31	Moderate (15m/sec)
Autoregression	0.89	High (30m/sec)
ANN	0.23	High (30m/sec)
Adaptive Spline Regression	1.75	High (30m/sec)

The comparative results are illustrated graphically in the following Figure 8.

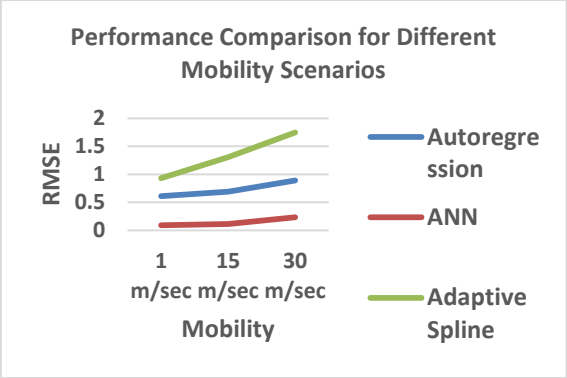


Fig. 8 Comparative RMSE Performance of Techniques in different Mobility Scenarios

From the results obtained through experiments and illustrated in Figure 8, a continuous increase of RMSE is observed for each of the three compared techniques, which is expected due to the fading conditions expected in the different mobility scenarios. However, the proposed technique, which maintains an RMSE value much lower than the adaptive spline method, has the slowest change in RMSE among the three methods, which indicates its stability in a dynamically changing mobility scenario. Such a stability may be attributed to the comparatively low-complexity approach that the proposed autoregression-based method takes in comparison to the other two techniques. The experimental results achieved consequently show that the proposed autoregression-based technique achieves acceptable performance in practical scenarios, which is a basic requirement that must be met for the model to be able to generate accurate estimates in significantly dynamic environments. Additionally, a neural network-based technique can also be used in conjunction with the proposed method in order to reduce the RMSE while restricting the overall complexity of such a hybrid technique to moderate levels. On the other hand, if reduction of complexity is a more significant issue for a particular implementation scenario, a linear spline regression method can be used in conjunction with the proposed autoregression technique to allow for low-complexity model design without significant loss in model accuracy.

5- Conclusions

On inspection of the recent research in the establishment of attenuation models for 5G millimeter wave communication signals, two major challenges are found to emerge. First, the attenuation of a 5G signal varies significantly due to meteorological phenomena such as rain, as well as the micro-environment of the network (for example, diffraction due to sand storms in arid regions). Second, attenuation models for indoor and outdoor environments are found to vary to a significant extent, and lack dynamicity to a certain degree. In such a scenario, the application of suitable machine learning techniques allows for design and establishment of dynamic models which can adapt to changes in the network conditions and can therefore generate estimates with greater accuracy than typically used attenuation models. It is also to be kept in mind that adaptive machine learning based model are more robust in the face of significant meteorological variations, further increasing the usefulness of such models in the present context. As a consequence, a dynamic machine learning based model which accounts for both LoS and NLoS signal propagation can be designed, as shown in this work, allowing optimization of RMSE of predictions at suitably low complexity, which in turn ensures that such a solution can be cheaply and easily implemented in practical scenarios for estimation of signal attenuation for 4G and 5G networks.

References

[1] T. S. Rappaport et al., "Millimeter Wave Mobile Communications for 5G Cellular: It Will Work!," in IEEE Access, vol. 1, pp. 335-349, 2013, doi: 10.1109/ACCESS.2013.2260813.

[2] T. S. Rappaport, Y. Xing, G. R. MacCartney, A. F. Molisch, E. Mellios and J. Zhang, "Overview of Millimeter Wave Communications for Fifth-Generation (5G) Wireless Networks—With a Focus on Propagation Models," in IEEE Transactions on Antennas and Propagation, vol. 65, no. 12, pp. 6213-6230, Dec. 2017, doi: 10.1109/TAP.2017.2734243.

[3] H. H. Hussein, H. A. Elsayed & S. M. Abd El-kader, "Intensive Benchmarking of D2D communication over 5G cellular networks: prototype, integrated features, challenges, and main applications," Wireless Networks, vol. 26, pp. 3183–3202, 2020. https://doi.org/10.1007/s11276-019-02131-2.

[4] Y. Sun, J. Cao, M. Ma, Y. Zhang, H. Li and B. NIU, "EAP-DDBA: Efficient Anonymity Proximity Device Discovery and Batch Authentication Mechanism for Massive D2D Communication Devices in 3GPP 5G HetNet," in IEEE Transactions on Dependable and Secure Computing, doi: 10.1109/TDSC.2020.2989784.

[5] S. N. Livieratos and P. G. Cottis, "Rain Attenuation Along Terrestrial Millimeter Wave Links: A New Prediction Method Based on Supervised Machine Learning," in IEEE Access, vol. 7, pp. 138745-138756, 2019, doi: 10.1109/ACCESS.2019.2939498.

[6] J. Huang, Y. Cao, X. Raimundo, A. Cheema and S. Salous, "Rain Statistics Investigation and Rain Attenuation Modeling for Millimeter Wave Short-Range Fixed Links," in IEEE Access, vol. 7, pp. 156110-156120, 2019, doi: 10.1109/ACCESS.2019.2949437.

[7] A. M. Al-Saman, M. Cheffena, M. Mohamed, M. H. Azmi and Y. Ai, "Statistical Analysis of Rain at Millimeter Waves in Tropical Area," in IEEE Access, vol. 8, pp. 51044-51061, 2020, doi: 10.1109/ACCESS.2020.2979683.

[8] Z. A. Shamsan, "Rainfall and Diffraction Modeling for Millimeter-Wave Wireless Fixed Systems," in IEEE Access,

vol. 8, pp. 212961-212978, 2020, doi: 10.1109/ACCESS.2020.3040624.

[9] F. D. Diba, M. A. Samad and D. -Y. Choi, "The Effects of Rain on Terrestrial Links at K, Ka and E-Bands in South Korea: Based on Supervised Learning," in IEEE Access, vol. 9, pp. 9345-9355, 2021, doi: 10.1109/ACCESS.2021.3049825.

[10] A. A. H. Budalal, M. R. Islam, K. Abdullah and T. Abdul Rahman, "Modification of Distance Factor in Rain Attenuation Prediction for Short-Range Millimeter-Wave Links," in IEEE Antennas and Wireless Propagation Letters, vol. 19, no. 6, pp. 1027-1031, June 2020, doi: 10.1109/LAWP.2020.2987462.

[11] Z. A. Shamsan, "Dust Storm and Diffraction Modelling for 5G Spectrum Wireless Fixed Links in Arid Regions," in IEEE Access, vol. 7, pp. 162828-162840, 2019, doi: 10.1109/ACCESS.2019.2951855.

[12] H. B. H. Dutty and M. M. Mowla, "Channel Modeling at Unlicensed Millimeter Wave V Band for 5G Backhaul Networks," 2019 5th International Conference on Advances in Electrical Engineering (ICAEE), 2019, pp. 769-773, doi: 10.1109/ICAEE48663.2019.8975439.

[13] A. Hikmaturokhman, M. Suryanegara and K. Ramli, "A Comparative Analysis of 5G Channel Model with Varied Frequency: A Case Study in Jakarta," 2019 7th International Conference on Smart Computing & Communications (ICSCC), 2019, pp. 1-5, doi: 10.1109/ICSCC.2019.8843632.

[14] S. K. M., S. A. Rao and N. Kumar, "Modeling and Link Budget Estimation of Directional mmWave Outdoor Environment for 5G," 2019 European Conference on Networks and Communications (EuCNC), 2019, pp. 106-111, doi: 10.1109/EuCNC.2019.8802001.

[15] Y. Dalveren, A. H. Alabish and A. Kara, "A Simplified Model for Characterizing the Effects of Scattering Objects and Human Body Blocking Indoor Links at 28 GHz," in IEEE Access, vol. 7, pp. 69687-69691, 2019, doi: 10.1109/ACCESS.2019.2919546.

[16] X. Cai, G. Zhang, C. Zhang, W. Fan, J. Li and G. F. Pedersen, "Dynamic Channel Modeling for Indoor Millimeter-Wave Propagation Channels Based on Measurements," in IEEE Transactions on Communications, vol. 68, no. 9, pp. 5878-5891, Sept. 2020, doi: 10.1109/TCOMM.2020.3001614.

[17] M. K. Arti, "Channel Estimation and Detection in Satellite Communication Systems," in IEEE Transactions on Vehicular Technology, vol. 65, no. 12, pp. 10173-10179, Dec. 2016, doi: 10.1109/TVT.2016.2529661.

[18] C. Lu et al., "A New Rain Attenuation Prediction Model for the Earth-Space Links," in IEEE Transactions on Antennas and Propagation, vol. 66, no. 10, pp. 5432-5442, Oct. 2018, doi: 10.1109/TAP.2018.2854181.

[19] J. X. Yeo, Y. H. Lee and J. T. Ong, "Rain Attenuation Prediction Model for Satellite Communications in Tropical Regions," in IEEE Transactions on Antennas and Propagation, vol. 62, no. 11, pp. 5775-5781, Nov. 2014, doi: 10.1109/TAP.2014.2356208.

[20] S. Mohanty, C. Singh and V. Tiwari, "Estimation of rain attenuation losses in signal link for microwave frequencies using ITU-R model," 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 2016, pp. 532-535, doi: 10.1109/IGARSS.2016.7729132.

[21] I. Shayea, T. Abd. Rahman, M. Hadri Azmi and M. R. Islam, "Real Measurement Study for Rain Rate and Rain Attenuation Conducted Over 26 GHz Microwave 5G Link System in Malaysia," in IEEE Access, vol. 6, pp. 19044-19064, 2018, doi: 10.1109/ACCESS.2018.2810855.

[22] S. Sun, T. S. Rappaport, M. Shafi, P. Tang, J. Zhang and P. J. Smith, "Propagation Models and Performance Evaluation for 5G Millimeter-Wave Bands," in IEEE Transactions on Vehicular Technology, vol. 67, no. 9, pp. 8422-8439, Sept. 2018, doi: 10.1109/TVT.2018.2848208.

[23] Md Anoarul Islam, Kausik Bhattacharya, Md. Abdul Alim Sheikh, P. K. Ghosh, Manabendra Maiti, Judhajit Sanyal, "Machine Learning Reinforced Ku-band Rain Attenuation Model for Tropical Locations", in 2019 Global Conference for Advancement in Technology (GCAT), Bangalore, India, October 18-20, 2019.

[24] M. A. Islam, M. Maiti, P. K. Ghosh & J. Sanyal, "Machine Learning-Based Rain Attenuation Prediction Model," In: Das N.R., Sarkar S. (eds) Computers and Devices for Communication. CODEC 2019. Lecture Notes in Networks and Systems, vol 147. Springer, Singapore. https://doi.org/10.1007/978-981-15-8366-7_3

[25] N. O. Oyie and T. J. O. Afullo, "Measurements and Analysis of Large-Scale Path Loss Model at 14 and 22 GHz in Indoor Corridor," in IEEE Access, vol. 6, pp. 17205-17214, 2018, doi: 10.1109/ACCESS.2018.2802038.

[26] M. A. Islam, M. Maiti, Q. M. Alfred, P. Kumar Ghosh and J. Sanyal, "Attenuation Modelling and Machine Learning Based SNR Estimation for 5G Indoor Link," 2020 IEEE VLSI DEVICE CIRCUIT AND SYSTEM (VLSI DCS), 2020, pp. 275-278, doi: 10.1109/VLSIDCS47293.2020.9179924.

[27] Noman Shabbir, Lauri Kütt, Muhammad M. Alam, Priit Roosipuu, Muhammad Jawad, Muhammad B. Qureshi, Ali R. Ansari, Raheel Nawaz, "Vision towards 5G: Comparison of radio propagation models for licensed and unlicensed indoor femtocell sensor networks", Physical Communication, Volume 47, 2021. https://doi.org/10.1016/j.phycom.2021.101371.

[28] D. Makoveyenko, S. Siden and V. Pyliavskyi, "Generalized 5G mmWave Propagation Model in an Urban Macro Environment," 2020 IEEE International Conference on Problems of Infocommunications. Science and Technology (PIC S&T), 2020, pp. 472-476, doi: 10.1109/PICST51311.2020.9468030.

[29] M. Sung et al., "RoF-Based Radio Access Network for 5G Mobile Communication Systems in 28 GHz Millimeter-Wave," in Journal of Lightwave Technology, vol. 38, no. 2, pp. 409-420, 15 Jan.15, 2020, doi: 10.1109/JLT.2019.2942636.

[30] H. Kim and S. Nam, "Transmission Enhancement Methods for Low-Emissivity Glass at 5G mmWave Band," in IEEE Antennas and Wireless Propagation Letters, vol. 20, no. 1, pp. 108-112, Jan. 2021, doi: 10.1109/LAWP.2020.3042524.

[31] J. Liu, D. W. Matolak, M. Mohsen and J. Chen, "Path Loss Modeling and Ray-Tracing Verification for 5/31/90 GHz Indoor Channels," 2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall), 2019, pp. 1-6, doi: 10.1109/VTCFall.2019.8891181.

[32] H. A. Obeidat et al., "An indoor path loss prediction model using wall correction factors for wireless local area network

and 5G indoor networks," in Radio Science, vol. 53, no. 4, pp. 544-564, April 2018, doi: 10.1002/2018RS006536.

[33] C. A. L. Diakhate, J. Conrat, J. Cousin and A. Sibille, "Millimeter-wave outdoor-to-indoor channel measurements at 3, 10, 17 and 60 GHz," 2017 11th European Conference on Antennas and Propagation (EUCAP), 2017, pp. 1798-1802, doi: 10.23919/EuCAP.2017.7928696.

[34] F. Yu et al., "Expansion RSS-based Indoor Localization Using 5G WiFi Signal," 2014 International Conference on Computational Intelligence and Communication Networks, 2014, pp. 510-514, doi: 10.1109/CICN.2014.117.

[35] H. Echigo, Y. Cao, M. Bouazizi and T. Ohtsuki, "A Deep Learning-Based Low Overhead Beam Selection in mmWave Communications," in IEEE Transactions on Vehicular Technology, vol. 70, no. 1, pp. 682-691, Jan. 2021, doi: 10.1109/TVT.2021.3049380.

[36] Gante, J., Falcão, G. & Sousa, L. Deep Learning Architectures for Accurate Millimeter Wave Positioning in 5G. Neural Process Lett 51, 487–514 (2020). https://doi.org/10.1007/s11063-019-10073-1

[37] H. Echigo, T. Ohtsuki, W. Jiang and Y. Takatori, "Fair Pilot Assignment Based on AOA and Pathloss with Location Information in Massive MIMO," GLOBECOM 2017 - 2017 IEEE Global Communications Conference, 2017, pp. 1-6, doi: 10.1109/GLOCOM.2017.8254813.

[38] H. Echigo and T. Ohtsuki, "Graph Coloring-Based Pilot Reuse with AOA and Distance in D2D Underlay Massive MIMO," 2018 IEEE 87th Vehicular Technology Conference (VTC Spring), 2018, pp. 1-6, doi: 10.1109/VTCSpring.2018.8417577.

[39] S. Moon, H. Kim and I. Hwang, "Deep learning-based channel estimation and tracking for millimeter-wave vehicular communications," in Journal of Communications and Networks, vol. 22, no. 3, pp. 177-184, June 2020, doi: 10.1109/JCN.2020.000012.

[40] J. Pudashine et al., "Probabilistic Attenuation Nowcasting for the 5G Telecommunication Networks," in IEEE Antennas and Wireless Propagation Letters, vol. 20, no. 6, pp. 973-977, June 2021, doi: 10.1109/LAWP.2021.3068393.

[41] M. J. Hasan, A. Rai, Z. Ahmad and J. -M. Kim, "A Fault Diagnosis Framework for Centrifugal Pumps by Scalogram-Based Imaging and Deep Learning," in IEEE Access, vol. 9, pp. 58052-58066, 2021, doi: 10.1109/ACCESS.2021.3072854.

[42] Hasan, Md J., Dongkoo Shon, Kichang Im, Hyun-Kyun Choi, Dae-Seung Yoo, and Jong-Myon Kim. 2020. "Sleep State Classification Using Power Spectral Density and Residual Neural Network with Multichannel EEG Signals" Applied Sciences 10, no. 21: 7639. https://doi.org/10.3390/app10217639.

[43] M. J. Hasan, J. Uddin and S. N. Pinku, "A novel modified SFTA approach for feature extraction," 2016 3rd International Conference on Electrical Engineering and Information Communication Technology (ICEEICT), 2016, pp. 1-5, doi: 10.1109/CEEICT.2016.7873115.

Journal of Information Systems and Telecommunication
Vol.11, No.2, April-June 2023, 94-101
http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

Hoax Identification of Indonesian Tweeters using Ensemble Classifier

Gus Nanang Syaifuddiin[1,2], Rizal Arifin[2*], Desriyanti[2], Ghulam Asrofi Buntoro[2], Zulkham Umar Rosyidin[2], Ridwan Yudha Pratama[2], Ali Selamat[3]

[1]. Department of Information Technology, Politeknik Negeri Madiun, Jl. Serayu No. 84 Madiun 63133, Indonesia
[2]. Faculty of Engineering, Universitas Muhammadiyah Ponorogo, Jl. Budi Utomo No. 10 Ponorogo 63471, Indonesia
[3]. Malaysia-Japan International Institute of Technology, Universiti Teknologi Malaysia, Jalan Sultan Yahya Petra, Kuala Lumpur 54100, Malaysia

Received: 21 Jan 2022/ Revised: 04 Jun 2022 / Accepted: 07 Jul 2022

Abstract

Fake information, better known as hoaxes, is often found on social media. Currently, social media is not only used to make friends or socialize with friends online, but some use it to spread hate speech and false information. Hoaxes are very dangerous in social life, especially in countries with large populations and ethnically diverse cultures, such as Indonesia. Although there have been many studies on detecting false information, the accuracy and efficiency still need to be improved. To help prevent the spread of these hoaxes, we built a model to identify false information in Indonesian using an ensemble classifier that combines the n-gram method, term frequency-inverse document frequency, and passive-aggressive classifier method. The evaluation process was carried out using 5000 samples from Twitter social media accounts in this study. The testing process is carried out using four schemes by dividing the dataset into training and test data based on the ratios of 90:10, 80:20, 70:30, and 60:40. The inspection results show that our software can accurately detect hoaxes at 91.8%. We also found an increase in the accuracy and precision of hoax detection testing using the proposed method compared to several previous studies. The results show that our proposed method can be developed and used in detecting hoaxes in Indonesian on various social media platforms.

Keywords: Hoax; Identification; Bahasa Indonesia; N-Gram; TF-IDF; Passive-Aggressive Classifier.

1- Introduction

With the development of technology, all kinds of information can be accessed very easily using various devices connected to the internet. One of the most widely accessed media by the public using the internet is social media initially, social media was created to communicate greetings between friends in cyberspace, but in its development now, social media is used to spread hate speech and fake information or hoax. Data from the Ministry of Communication and Information (KOMINFO) of the Republic of Indonesia shows about 800,000 Hoax Spreading Sites in Indonesia. The Ministry of Communication and Information's preventions include blocking sites that spread hoaxes and conducting socializations about hoax news and its dangers both in print media, television media, online media, and social media. Social media has a positive or negative impact. Among the positive effects is an increase in public literacy regarding certain conditions, cases, or events, for example, Covid-19. From social media, we can also find out public sentiment towards certain figures, data from Twitter, namely tweets of public opinion, there is a figure that can be used as data for sentiment analysis, now there is a lot of data on social media that can be used to find out public or customer sentiment towards certain products for example, and many more applications of data from social media. We have conducted a sentiment analysis of the 2019 Indonesian presidential election in previous research using machine learning algorithms. [1]. The results of our study can be used as an illustration of the sentiments of the Indonesian people towards the 2019 Indonesian presidential candidate, especially the Twitter social media user community.

✉ **Rizal Arifin**
rarifin@umpo.ac.id

Another positive impact of the development of information technology is that we can feel more and more, processes that used to be mostly done offline or offline means to meet and meet face-to-face, with technological developments that are very rapid and fast, one by one the processes that used to be offline are now being carried out online, one of which is the teaching and learning process, which we have experienced together for the past two years, the whole world is at war with Covid-19, where one way to stop the spread of the virus is by not having direct contact with Covid-19 patients, meaning by not meeting each other and meeting face to face. It will reduce the risk of contracting Covid-19. Therefore, the teaching and learning process is one of the sectors affected by the call to reduce face-to-face contact. In the past two years, almost all learning and sharing of knowledge worldwide have been carried out online [2]–[6] with the development of information technology. These processes that used to be offline can now still be done online, although there are still many shortcomings and weaknesses. Information technology products in teaching and learning, for example, e-learning, learning media with animated images, audio and video, educational games, and video conferencing.

However, the development of information technology also has a negative impact on the community if they are less able to sort and select information. Because in cyberspace or the internet, people can be anyone, use a mask of goodness or a mask of evil, or even use both masks simultaneously. Therefore, various types of information can spread quickly through the internet from these irresponsible anonymous, false, or hoax information is widely spread on the internet, especially on social media such as Twitter, Facebook, and Instagram, because these social media do not go through review process by editors or experts. This is what causes fake information or hoaxes to be accepted by the public quickly without filters that can damage the unity and integrity of the nation [7]–[9]. Fake information packaged well and with convincing language becomes as if it is true is known as a hoax. During the COVID-19 pandemic, for almost two years, we have received information about the virus, especially COVID-19; the information circulating there is true and good, but not a few are also misleading and spread throughout the world [10]–[12], misleading information This causes anxiety and fear in the Indonesian people in particular and the world community in general.

Currently, the spread of false information or hoaxes in Indonesia is increasingly widespread, and many are being spread every minute. Hoaxes are usually widely spread in every significant event in Indonesia, from the campaign for the Indonesian presidential election to natural events being used as material for false information by those who are not responsible. Indonesia is a country with a diverse population and ethnicity with the large number of people and customs. The spread of hoax information can damage the unity and integrity of the nation [13]–[16]. Therefore, serious efforts are needed from all parties, especially the authorities, namely the government, to overcome the spread of hoaxes in the community. Prevention carried out by the Ministry of Communication and Information (KOMINFO) includes blocking sites that spread hoaxes and socializing about hoax news and its dangers in print media, television media, online media, and social media [16]. Researchers in the field of information technology, especially artificial intelligence, have tried to apply various methods to accurately detect hoax information that is massively spread on social media. Other researchers, such as Hasan, et al., developed an artificial neural network-based method for diagnosing potential centrifugal pump failures [17] and classifying the sleep state of a human being [18]. In addition, Hasan et al. also developed a new approach to the method of segmentation-based texture fractal analysis (SFTA), which is reported to have better accuracy than conventional SFTA [19]. Some of the methods that researchers widely use are unigram, bigram and include n-grams, for the most accurate method is the N-gram method; the way these method works is to tokenize sentences according to length N, so researchers determine how long N is in n-grams to get most accurate accuracy [20]–[22]. The weighting method most widely used by artificial intelligence researchers, especially natural language processing (NLP) in sentence extraction, is the term frequency-inverse document frequency (TF-IDF). The TF-IDF process calculates the frequency of occurrence of a word in a sentence and then compares it with the inverse of the data. The TF-IDF process also calculates how often a word is in a sentence; the more often the word appears, the smaller the weight value, meaning that the word is unimportant in a sentence, usually like conjunction (and, which, in, will, with, etc.) [22]–[25], and Passive aggressive algorithms are large-scale learning algorithms that are widely used in big data applications. They do not need the speed of learning like Perception. However, they have regularization parameters, unlike Perception. this algorithm is excellent for detecting fake information or on social media sites like Twitter and WhatsApp, wherein new social media data is added every second. [26]–[28]. Although there has been a lot of research in the field of artificial intelligence, especially regarding the detection of false information or hoaxes, there is still a need to research hoax detection to continue to improve the accuracy and effectiveness of hoax detection as an effort to reduce and prevent the spread of hoaxes in the community. In this study, we built a model to identify false information in Indonesian using an ensemble classifier which is a combination of the n-gram method, term frequency-inverse document frequency, and passive-aggressive classifier method. This is aimed to obtain the best accuracy and precision of the hoax detection among the proposed methods.

2- Methods

The first stage in this research is data collection. The data used in this study are tweets from both personal accounts and online news accounts with a total of 5000 tweet data taken using several keywords from several topics. Tweet data was collected from July to August 2021. The format for data collection uses labels as shown in Table 1.

Table 1: Label format used in data collection.

Feature	Description
Topic/theme	Topic or theme of tweet data
Keyword	Keyword used to search the data
Tweet text	Tweet text from search results
Image	URL of the image included in the tweet (if any)
URL	Tweet URL
Label	Labels on tweet data (valid or hoax)

The processes of identifying hoaxes are carried out using several trusted references. Figure 1 shows the procedures or steps taken in the identification of hoaxes used in this study.

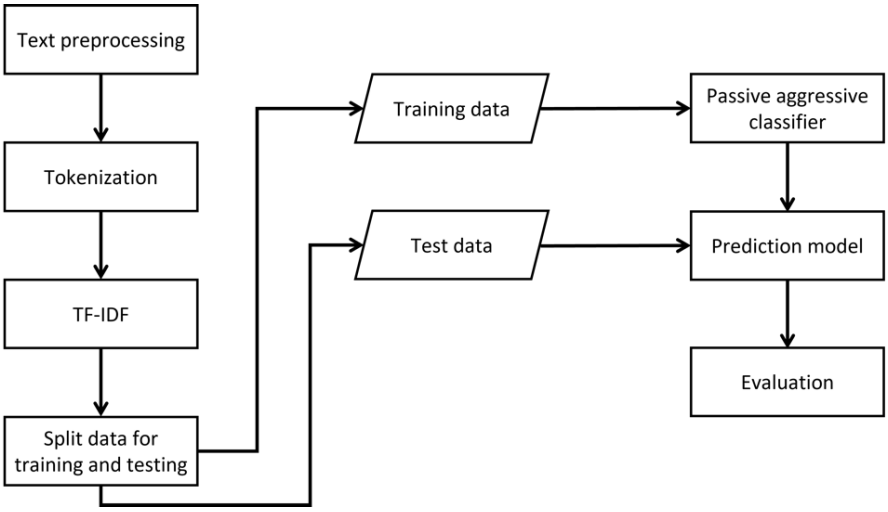


Fig. 1. Flowchart for hoax identification procedure using a combination of TF-IDF, n-gram, and passive aggressive classifier methods.

The stages in this research are as follows:

1. Several processes were carried out during the preprocessing stage to simplify and optimize the data processing. Some of the processes implemented include removing URLs, the # and @ symbols, punctuation marks, stop word lists, emojis, and numbers and changing the text into lowercase.

Table 2: Data preprocessing.

Stage	Tweet data
Original tweet	Buat para ortu yg sdh mantu atau mau punya cucu atau ada ponakan2: "Vaksin Penyebab Autis" Buat para Pasangan... http://fb.me/6NaAYEKv4
After preprocessing	buat para ortu yg sdh mantu atau mau punya cucu atau ada ponakan2 vaksin penyebab autis buat para pasangan

2. The tokenization in this study uses n-gram, a model often applied in document processing. N-gram tokenizes sentences of length N [20]. The result of n-grams will be calculated using TF-IDF. In this study, sentence tokenization was applied using n-grams, and the number of features produced is shown in Table 3.

Table 3. The number of features resulting from n-gram tokenization.

N-gram type	Feature count
Unigram	4366
Bigram	11643
Trigram	12524
Unigram + bigram	16009
Bigram+ trigram	24167
Unigram + bigram + trigram	28533

3. The next stage, TF-IDF, describes the importance of a word in a sentence or document. This process calculates the frequency of occurrence of a word and compares it with the inverse of the data [21]. This calculation allows an assessment of the role of a word in a sentence or document.
4. The stage before testing was splitting the data into training and testing. After the dataset was divided into training and testing data, the training data were trained

using a passive-aggressive classifier included in the category of online learning algorithms applied to machine learning. The passive-aggressive algorithms are a class of large-scale learning algorithms. They do not need a learning rate like Perception. They do, however, have a regularization parameter, unlike the Perception [28]. Furthermore, data testing was used for the prediction models. The testing process was conducted using four different schemes by dividing the dataset into training and testing data based on a ratio of 90:10, 80:20, 70:30, and 60:40, after referred to as splits 1, 2, 3, and 4, respectively.

5. The evaluation stage in this study uses the accuracy, precision, and recall from the experiments conducted.

Table 4. Confusion matrix.

Prediction	Positive actual (1)	Negative actual (0)
Positive prediction	True-positive (TF)	False-positive (FP)
Negative prediction	False-negative (FN)	True-negative (TN)

The process results were evaluated using the confusion matrix shown in Table 4. The test scenario is carried out using the n-gram feature (bigram, unigram, trigram, unigram + trigram, bigram + trigram, unigram + trigram) with various comparisons of the training and testing data.
The testing process was carried out using four different schemes by dividing the dataset into training and testing data based on a ratio of 90:10, 80:20, 70:30, and 60:40, referred to as split 1, 2, 3, and 4, respectively.

2-1- N-Gram

N-gram (adjacent n-gram) is a series of n characters or words extracted from a text. Usually, the n-grams that are often used are bigrams and trigrams, with the values of n being 2 and 3, respectively; N-grams are also called N-character chunks taken from a string [18].
Basically, the n-gram model is a probabilistic model designed by mathematicians from Russia in the early 20th century and later developed to predict the next item in a sequence of items. According to the application, items can be letters/characters, words, or others. One of them, the word-based n-gram model, is used to predict the next word in certain word order. In the sense that an n-gram is just a collection of words with each word having a length of n words. For example, an n-gram of size 1 is called a unigram; size 2 as “bigram”; size 3 as "trigram", and so on. In character generation, N-grams consist of n-character-long substrings of a string; in another definition, n-grams are n-character chunks of a string. This n-gram method takes n character pieces from a word that is continuously read from the source text to the end of the document.
For example the word "HOAX" can be broken down into the following n-grams:

unigram: H, O, A, X
bigram: HO, OA, AX
trigram: HOA, OAX and so on.

While in word generation, the n-gram method is used to take n-word pieces from a series of words (sentences, paragraphs, readings) which are read continuously from the source text to the end of the document.

2-2- TF-IDF (Term Frequency Inverse Document Frequency)

TF-IDF functions to convert text data into vectors by paying attention to whether a word is informative enough or not. TF-IDF makes words that appear frequently have a value that tends to be small, while words that occur rarely will have a value that tends to be large. Words that often appear are also called stopwords and are usually considered less important because they are only conjunction (at, will, with, etc.) [19].
TF-IDF stands for Term Frequency — Inverse Document Frequency. TF-IDF is a combination of 2 processes: Term Frequency (TF) and Inverse Document Frequency (IDF).

1. Term Frequency (TF)

Term Frequency (TF) counts the number of times a word appears in a document as shown in Eq. (1). Because the length of each document can be different [20], generally, the TF value is divided by the length of the document (the total number of words in the document).

$$tf_{t,d} = \frac{n_{t,d}}{Total\ number\ of\ terms\ in\ document} \quad (1)$$

Description
tf = frequency of occurrence of words in a document

2. Inverse Document Frequency (IDF)

After successfully calculating the Term Frequency value, we calculate using Eq. (2) the Inverse Document Frequency (IDF) value, which is a value to measure how important a word is [21]. The smaller the IDF value, the less important the word will be, and vice versa. IDF will assess words that often appear as less important words based on how they appear throughout the document.

$$idf_d = log(\frac{Number\ of\ document}{Number\ of\ document\ with\ term\ t'}) \quad (2)$$

After we have TF and IDF, next we can calculate the value of TF-IDF which is the product of TF and IDF using Eq. (3).

$$tfidf_{t,d} = tf_{t,d} \times idf_d \quad (3)$$

2-3- Passive Aggressive Classifier

Passive-aggressive algorithms include machine learning algorithms that are popularly used in big data applications [23].

The Passive-Aggressive Algorithm, which is usually used for large-scale learning, is also one of the online learning algorithms. In online machine learning algorithms, the input data come sequentially, and the machine learning model is updated sequentially, in contrast to conventional learning, where the entire training dataset is used at once.

This algorithm is advantageous in situations where there is a large amount of data, and it is computationally impossible to train the entire data set due to the sheer size of the data [22].

The Passive-Aggressive Algorithm is slightly like the Perceptron model because it does not require a learning speed. However, they do include a regularization parameter.

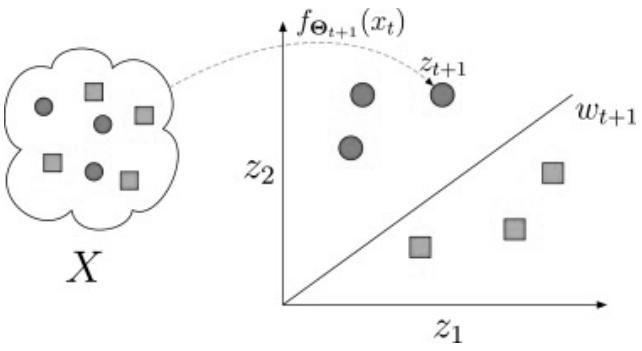


Fig. 2. Illustration Passive-Aggressive online learning

3- Results and Discussion

This research begins with the retrieval of data from Twitter. The data used in this study are tweets from personal accounts, group accounts, organizational accounts, and online news accounts, with a total of 5000 tweet data taken using several keywords from several topics. Tweet data were collected from July to August 2021. The following process is tokenization, which is the splitting of sentences into one token for each word. After being tokenized, each word is then given a weight using the TF-IDF word weighting method. Then enter the hoax news classification process or not with the Passive-Aggressive Classifier method; before entering the research dataset, testing is separated between hoax information and valid information, then the percentages obtained at the respective levels are 41% and 59%, for comparison. Next, the test data and training data are divided into 70% training data and 20% test data respectively from the dataset. The last stage of the research process is evaluating the prediction results, calculated using the accuracy model, namely the Confusion matrix.

In Table 5, the results of the hoax identification test are presented using several variations of the n-gram model for several combinations of the distribution of training data and test data, namely split 1, 2, 3, and 4 as previously defined split 1 = 90:10, split 2 = 80:20, split 3 = 70:30, and split 4 60:40. The trial application of the method used is to multiply the features using n-grams and divide the dataset by several division combinations. The comparison of the hoax information identification test results can be seen in Table 5.

Table 5: Comparison of the results of the hoax identification test.

N-gram model	Accuracy (%)			
	Split 1	Split 2	Split 3	Split 4
Unigram	90.98	91.39	90.41	87.68
Bigram	91.80	89.75	87.12	85.63
Trigram	81.97	79.51	80.55	80.49
Unigram + bigram	90.16	91.80	90.41	88.91
Bigram + trigram	89.34	89.75	87.12	84.80
Unigram + bigram + trigram	88.52	90.57	90.41	89.12

From Table 5, the highest accuracy value of all experiments is when the data split is split 1 with the bigram method. For the results of split 1, the bigram method is obtained with the highest accuracy value of 91.80%. The highest accuracy was obtained because in slip 1 the distribution of the training data was much larger than the test data, with a ratio of 90:10. The more training data, the easier the model is built to recognize the test data. As for split 2 data, the best accuracy is obtained by using a combination of unigram + bigram with an accuracy value of 91.8%. In testing the split 3 scheme, three models with n-gram tokenization received the same accuracy of 90.41%, namely unigram, unigram + bigram, and unigram + bigram + trigram. The maximum accuracy obtained in the split 4 data combination is lower than the maximum accuracy in the three previous data combinations. This is due to the lack of training data used in split 4, which is 60%.

Based on Table 5, we can see that the highest accuracy is obtained when using split 1 data division, which is 90:10 with the bigram model; the accuracy value reaches 91.80% for the test results on split 2 data with the unigram + bigram model the accuracy reaches 91.80%. After a series

of trials, it was found that the words that had the most significant TF-IDF value of 10 from the bigram and unigram + bigram models, the results were dominated by words or sentences that asked readers to spread the news.

method combined with 90% training data distribution and 10% testing data from the dataset. This study's bigram tokenization method produces the highest accuracy because bigram tokenization splits sentences into two-word tokens. In this research data, most hoax information emerges from two-word pairs.

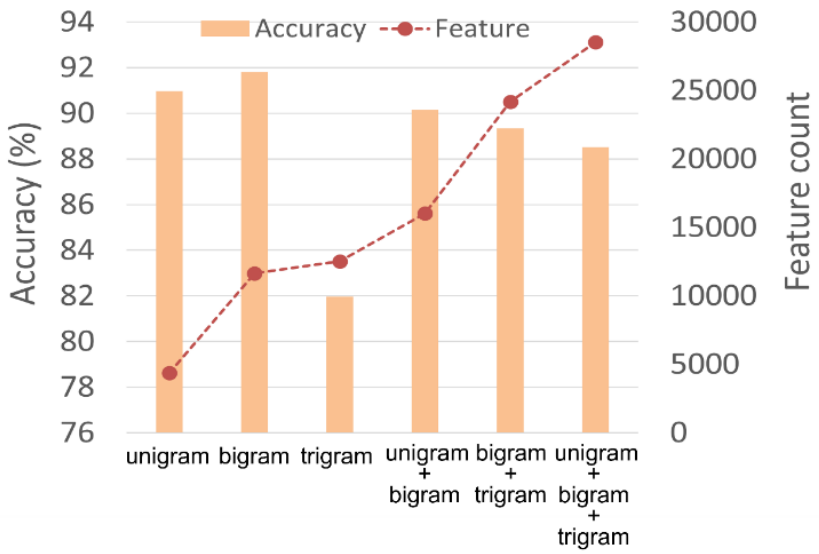


Fig. 2: Accuracy and feature count of hoax identification test results on several variations of n-gram on split 1 data combination.

From Figure 2, the more complex the n-gram used, the more features it has. However, from the several tests carried out, it can be seen that the results of the tests carried out on data split 1 show that the number of bigram features is still less than the number of features from unigram + bigram + trigram; it can be concluded that the number of features is not positively correlated with the accuracy obtained. It can also be seen in Figure 2 that trigrams which have more features than bigrams, get much lower accuracy.

Table 6. Words or word pairs that appear most often in hoaxes occurring in Bahasa Indonesia

Unigram	Bigram	Unigram + Bigram
'ortu' 'bocah' 'sungguh' 'miris' 'share' 'jiwa' 'kisah' 'ikut' 'mungkin' 'group'	'di share' 'share ke' 'ortu bocah' 'group ortu' 'ke group' 'hati miris' 'bisa di' 'bocah ini' 'akun facebook' 'dari sebuah'	['ke group' 'ortu bocah' 'di share' 'share ke' 'group ortu' 'bocah ini' 'ini datang' 'mungkin bisa' 'miris ini' 'sebuah akun']

Table 6 shows the words or word pairs that often appear in the Indonesian hoaxes used in this study. The highest accuracy in this study was when testing using the bigram

Table 7: Comparison between the performance of our method and those of other studies

Method	Accuracy	Precision	Recall	f-measure
This research	91.8	93.6	90.7	92.1
Zaman, et al. [29]	87.0	91.0	100.0	95.0
Pratiwi, et al. [30]	78.6	67.1	89.4	76.4

Zaman et al. [29] used the naive Bayes algorithm and user feedback to detect hoaxes, using the best ratio of training to test data, i.e., 70:30. Pratiwi et al. [30] used random repetition three times by applying the PHP-ml library and obtained the highest accuracy of 78.6%, with 70% training data and 30% testing data. A comparison of our classification performance with that of other studies is presented in Table 7. From the data shown in the table, it can be seen that there is an increase in the accuracy and precision of hoax detection testing with the proposed method in comparison to the two previous studies. This shows that our proposed method has the potential to be further developed and used in the detection of hoaxes in Bahasa Indonesia on various social media platforms.

4- Conclusions

Based on the results of this study, the model building can be used to identify hoaxes in Indonesia. The highest accuracy in this study was 91.8%, which was obtained when using a combination of tokenization bigrams with a split ratio of 1 and unigram + bigram with a split ratio of 2. The lowest accuracy of 79.51% was obtained when using a combination of trigram tokenization with a split ratio of 2. We also found that an increase in accuracy and precision of hoax detection testing can be achieved using the proposed method in comparison to other previous approaches.

Acknowledgments

This work was supported by a PTUPT 2021 research grant from the Ministry of Education, Culture, Research, and Technology of the Republic of Indonesia [contract # 313/E4.1/AK.04.PT/2021].

References

[1] G. A. Buntoro, R. Arifin, G. N. Syaifuddiin, A. Selamat, O. Krejcar, and H. Fujita, "The implementation of the machine learning algorithm for the sentiment analysis of Indonesia's 2019 presidential election," *IIUM Eng. J.*, vol. 22, no. 1, pp. 78–92, 2021, doi: https://doi.org/10.31436/iiumej.v22i1.1532.

[2] S. Pokhrel and R. Chhetri, "A Literature Review on Impact of COVID-19 Pandemic on Teaching and Learning," *High. Educ. Futur.*, vol. 8, no. 1, pp. 133–141, 2021, doi: https://doi.org/10.1177/2347631120983481.

[3] M. Aristeidou and S. Cross, "Disrupted distance learning: the impact of Covid-19 on study habits of distance learning university students," *Open Learn. J. Open, Distance eLearning*, vol. 36, no. 3, pp. 263–282, 2021, doi: https://doi.org/10.1080/02680513.2021.1973400.

[4] M. M. Zalat, M. S. Hamed, and S. A. Bolbol, "The experiences, challenges, and acceptance of e-learning as a tool for teaching during the COVID-19 pandemic among university medical staff," *PLoS One*, vol. 16, no. 3, p. e0248758, 2021, doi: https://doi.org/10.1371/journal.pone.0248758.

[5] C. Saxena, H. Baber, and P. Kumar, "No Title," *J. Educ. Technol. Syst.*, vol. 49, no. 4, pp. 532–554, 2021, doi: https://doi.org/10.1177/0047239520977798.

[6] A. R. Alsoud and A. A. Harasis, "The Impact of COVID-19 Pandemic on Student's E-Learning Experience in Jordan," *J. Theor. Appl. Electron. Commer. Res.*, vol. 16, no. 5, pp. 1404–141, 2021, doi: https://doi.org/10.3390/jtaer16050079.

[7] M. Celliers and M. Hattingh, "A Systematic Review on Fake News Themes Reported in Literature," in *Responsible Design, Implementation and Use of Information and Communication Technology*, 2020, pp. 223–234, doi: https://doi.org/10.1007/978-3-030-45002-1_19.

[8] T. Buchanan, "Why do people spread false information online? The effects of message and viewer characteristics on self-reported likelihood of sharing social media disinformation," *PLoS One*, vol. 15, no. 10, p. e0239666, 2020, doi: https://doi.org/10.1371/journal.pone.0239666.

[9] T. Khan, A. Michalas, and A. Akhunzada, "Fake news outbreak 2021: Can we stop the viral spread?," *J. Netw. Comput. Appl.*, vol. 190, p. 103112, 2021, doi: https://doi.org/10.1016/j.jnca.2021.103112.

[10] M. Montesi, "Understanding fake news during the Covid-19 health crisis from the perspective of information behaviour: The case of Spain," *J. Librariansh. Inf. Sci.*, vol. 53, no. 3, pp. 454–465, 2020, doi: https://doi.org/10.1177/0961000620949653.

[11] S. van der Linden, J. Roozenbeek, and J. Compton, "Inoculating Against Fake News About COVID-19," *Front. Psychol.*, vol. 11, p. 566790, 2020, doi: https://doi.org/10.3389/fpsyg.2020.566790.

[12] A. Alasmari, A. Addawood, M. Nouh, W. Rayes, and A. Al-Wabil, "A Retrospective Analysis of the COVID-19 Infodemic in Saudi Arabia," *Futur. Internet*, vol. 13, no. 10, p. 254, 2021, doi: https://doi.org/10.3390/fi13100254.

[13] Y. I. Ferdiawan, P. A. D. Nurjanah, E. P. Krisdyan, A. Hidayatullah, H. J. M. Sirait, and N. A. Rakhmawati, "HOAX Impact to Community Through Social Media Indonesia," *Cakrawala*, vol. 19, no. 1, pp. 121–124, 2019, doi: https://doi.org/10.31294/jc.v19i1.4452.

[14] K. Lutfiyah, "Hoax and Fake News During Covid-19: Is the Law Effective in Overcoming It?," *Indones. J. Int'l Clin. Leg. Educ.*, vol. 2, no. 3, pp. 345–360, 2020, doi: https://doi.org/10.15294/ijicle.v2i3.38422.

[15] N. M. Nasir, B. Baequni, and M. I. Nurmansyah, "Misinformation Related to Covid-19 in Indonesia," *J. Adm. Kesehat. Indones.*, vol. 8, no. 1, pp. 51–59, 2020, doi: http://dx.doi.org/10.20473/jaki.v8i0.2020.51-59.

[16] M. Rasidin, D. Witro, B. Yanti, R. Purwaningsih, and W. Nurasih, "The Role of Government in Preventing The Spread if Hoax Related The 2019 Elections in Social Media," *Diakom*, vol. 3, no. 2, pp. 127–3, 2020, doi: https://doi.org/10.17933/diakom.v3i2.76.

[17] M. J. Hasan, A. Rai, Z. Ahmad, and J.-M. Kim, "A Fault Diagnosis Framework for Centrifugal Pumps by Scalogram-Based Imaging and Deep Learning," *IEEE Access*, vol. 9, pp. 58052–58066, 2021, doi: https://doi.org/10.1109/CEEICT.2016.7873115.

[18] M. J. Hasan, D. Shon, K. Im, H.-K. Choi, D.-S. Yoo, and J.-M. Kim, "Sleep State Classification Using Power Spectral Density and Residual Neural Network with Multichannel EEG Signals," *Appl. Sci.*, vol. 10, no. 21, p. 7639, 2020, doi: https://doi.org/10.3390/app10217639.

[19] M. J. Hasan, J. Uddin, and S. N. Pinku, "A novel modified SFTA approach for feature extraction," in *2016 3rd International Conference on Electrical Engineering and Information Communication Technology (ICEEICT)*, 2016, pp. 1–5, doi: https://doi.org/10.1109/CEEICT.2016.7873115.

[20] H. E. Wynne and Z. Z. Wint, "Content Based Fake News Detection Using N-Gram Models," in *Information Integration and Web-based Applications & Services*, 2019, pp. 669–673, doi: https://doi.org/10.1145/3366030.3366116.

[21] H. Ahmed, I. Traore, and S. Saad, "Detection of Online Fake News Using N-Gram Analysis and Machine Learning Techniques," in *Intelligent, Secure, and Dependable*

Systems in Distributed and Cloud Environments, 2017, pp. 127–138, doi: https://doi.org/10.1007/978-3-319-69155-8_9.

[22] H. Ahmed, I. Traore, and S. Saad, "Detecting opinion spams and fake news using text classification," *Secur. Priv.*, vol. 1, p. e9, 2018, doi: https://doi.org/10.1002/spy2.9.

[23] J. Huang, "Detecting Fake News With Machine Learning," *J. Phys. Conf. Ser.*, vol. 1693, p. 012158, 2020, doi: https://doi.org/10.1088/1742-6596/1693/1/012158.

[24] B. Al Asaad and M. Erascu, "A Tool for Fake News Detection," in *2018 20th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, 2018, pp. 379–386, doi: https://doi.org/10.1109/SYNASC.2018.00064.

[25] M. J. Awan *et al.*, "Fake News Data Exploration and Analytics," *Electronics*, vol. 10, p. 2326, 2021, doi: https://doi.org/10.3390/electronics10192326.

[26] S. Gupta and P. Meel, "Fake News Detection Using Passive-Aggressive Classifier," in *Inventive Communication and Computational Technologies*, 2020, pp. 155–164, doi: https://doi.org/10.1007/978-981-15-7345-3_13.

[27] R. R. Mandical, N. Mamatha, N. Shivakumar, R. Monica, and A. N. Krishna, "Identification of Fake News Using Machine Learning," in *2020 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*, 2020, pp. 1–6, doi: https://doi.org/10.1109/CONECCT50063.2020.9198610.

[28] A. Chugh, Y. Arora, J. Singh, Shobhit, and Ronak, "Media Manipulation Detection System Using Passive Aggressive," *Int. J. Innov. Res. Comput. Sci. Technol.*, vol. 9, no. 3, pp. 48–52, 2021, doi: https://doi.org/10.21276/ijircst.2021.9.3.8.

[29] B. Zaman, A. Justitia, K. N. Sani, and E. Purwanti, "An Indonesian Hoax News Detection System Using Reader Feedback and Naïve Bayes Algorithm," *Cybern. Inf. Technol.*, vol. 20, no. 1, pp. 82–94, 2020, doi: https://doi.org/10.2478/cait-2020-0006.

[30] I. Y. R. Pratiwi, R. A. Asmara, and F. Rahutomo, "Study of hoax news detection using naïve bayes classifier in Indonesian language," in *2017 11th International Conference on Information Communication Technology and System (ICTS)*, 2017, pp. 73–78, doi: https://doi.org/10.1109/ICTS.2017.8265649.

Journal of Information Systems and Telecommunication
Vol.11, No.2, April-June 2023, 102-109
http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

Computational Model for Image Processing in the Minds of People with Visual Agnosia using Fuzzy Cognitive Map

Elham Askari[1*], Sara Motamed[1]

[1]. Department of Computer Engineering, Fouman and Shaft Branch, Islamic Azad University, Fouman, Iran

Received: 08 Feb 2022 / Revised: 15 May 2022 / Accepted: 18 Jun 2022

Abstract

The Agnosia is a neurological condition that leads to an inability to name, recognize, and extract meaning from the visual, auditory, and sensory environment, despite the fact that the receptor organ is perfect. Visual agnosia is the most common type of this disorder. People with agnosia have trouble communicating between the mind and the brain. As a result, they cannot understand the images seen. In this paper, a model is proposed that is based on the visual pathway so that it first receives the visual stimulus and then, after understanding, the object is identified. In this paper, a model based on the visual pathway is proposed and using intelligent Fuzzy Cognitive Map will help improve image processing in the minds of these patients. First, the proposed model that is inspired by the visual perception pathway, is designed. Then, appropriate attributes that include the texture and color of the images are extracted and the concept of the seen image is perceived using Fuzzy Cognitive Mapping, the meaning recognition and the relationships between objects. This model reduces the difficulty of perceiving and recognizing objects in patients with visual agnosia. The results show that the proposed model, with 98.1% accuracy, shows better performance than other methods.

Keywords: Visual agnosia; Fuzzy Cognitive Mapping; Visual model; Mind.

1- Introduction

Agnosia refers to neurological conditions that results in an inability to know, name, recognize, and meaning extract from the visual, auditory, and sensory environment. This disorder causes the loss of ability to receive information through one of the senses, despite the fact that the receptor organ is perfect. [1, 2] Discrete brain lesions can lead to various forms of agnosia that may include any sense of error in individuals. There are several types of agnosia, including:

1-Auditory agnosia is the inability to recognize objects by sound (such as a phone ring). 2- Finger agnosia, which leads to impaired naming and recognition of the fingers of oneself and others and often follows damage to the parietal lobe. 3- Agnosia of a special category, in which people are not able to name living things but can say the names of objects (or vice versa). In this agnosia, the disorder in the right temporal lobe, which is specific to the perception of living things, and the left lobe for the perception of inanimate objects is the cause of this problem. 4- Semantic agnosia, in which people are so-called "blind object". They use the non-visual sensory system to recognize objects. For example, sensing, smelling an object is one of the things through which people can understand the meaning and concept of the object. 5- In color agnosia, people have difficulty recognizing and recognizing color but are able to understand and distinguish colors. 6- Alexei agnosia is the inability to recognize texts. 7- Tactile agnosia is related to the sense of touch, that is, the touch of objects. People have difficulty recognizing objects by touch based on their size, weight and texture. 8- In temporal agnosia, one has difficulty in understanding the sequence, duration and duration of events. 9- Music agnosia is a kind of agnosia in the field of music. This disorder causes a lack of recognition of musical notes, rhythm and intervals and an inability to understand music. Other forms of agnosia involve very specific and complex processes in one sense. It is questionable to describe why people can identify one type of object and fail to identify and define another. [1]

Visual agnosia is the most common type of this disorder. Visual agnosia or disorder in object recognition is a condition in which the patient is not able to recognize objects visually and pictorially. Although the role and

✉ **Elham Askari**
askary.elham@gmail.com

function of the sense of sight is completely normal, means objects are observed, but the person is not able to percept their meaning, and cannot identify the object. [3]

In 1890, Heinrich Leicher argued that there were two conditions in which object recognition disorder could occur. They included if the damage would occur to the primary perceptual processing or if there would be a disorder in displaying the real object. If it was related to the display of the real object, it would not allow the object to be stored in visual memory, so the person would not be able to recognize the object. Leicher also proposed a visual recognition model that was presented at two distinct levels:

A. A perception, which expresses the function of perceptual processing of the stimulus (defect in perceptual processing).

B. Associative, which means perception with previous experiences.

Agnosia is caused by damage to the parietal lobe, temporal lobe, or occipital lobe of the brain. These areas store used and important memories of familiar objects, perspectives, sounds, and memory integration with perception and recognition. So, after damage to any of the lobes, symptoms will be revealed as follows:

- Parietal lobe: This type of damage is usually caused by a brain stroke. Indivisuals have trouble recognizing a familiar object (such as a key or pin). However, when they look at an object, they can recognize and identify it.
- The occipital lobe: In this injury, people cannot recognize objects such as a spoon or pencil even if they can see them. This disorder is called visual agnosia. They may not even recognize familiar faces.
- Temporal lobe: This damage can also cause auditory agnosia in which a person cannot hear sounds, as well as visual agnosia. [4, 5]

Diagnosis of agnosia requires careful examination of a person's mental state and cognitive abilities. In addition, the physician must carefully assess the individual's ability to perceive visual stimuli or other stimuli. In the case of visual agnosia, this process involves evaluating, measuring and testing a person's Field Of View(FOV), color perception, image, reading skill, face recognition, drawing and recognizing real objects and drawing lines. [6]

This disease could be diagnosed by brain imaging (such as Computed Tomography or Magnetic Resonance Imaging with or without angiographic protocol) or neurological and electroencephalographic tests, and treatment could be started as soon as possible. [6]

Andrea Serino et al. described the diagnosis of perceptual agnosia as a case report in their study. In perceptual agnosia, due to the nature of the injury factors, there will be primary visual impairment. Most injuries are due to lack of oxygen or carbon monoxide poisoning or heart attack, which often causes extensive neurological disorders. The report was about a patient with bilateral cortical injury in which the patient first suffered from cortical blindness and after improving, symptoms of perceptual agnosia were observed. The patient's low spatial intelligence prevented him from recognizing the unique attributes of visual stimuli. [7]

In a case study, Erickson et al. examined severe visual agnosia in a child with an electrophysiological pattern of output in the occipito-temporal regions. The child with sporadic seizure had an inability to recognize objects without visual impairment. After analyzing the maps, left occipito-temporal disorder was diagnosed and signs of perceptual agnosia were observed. [8, 9]

Barton et al. studied patients who had difficulty at object recognition (people with visual agnosia) and showed their differences from healthy individuals by considering the direction of their eye movement when looking at the image. Their interpretation was that eye movements could be modeled as a selection of highlighted points, indicating that agnosic individuals have an increasing reliance on visual highlighted parts such as brightness and contrast. In addition, it states that patients' different perceptual problems may be highlighted by selecting the weights of the various attributes involved in a map. Finally, they stated that highlights are not always a good predictor of agnosia diagnosis. [9]

Many computational models have been proposed to recognize mental structures. Moren and Balkenius presented a computational model for describing the structures of the brain that are involved in attention and perception. This model could help the mentally patients such as visual agnosia. The perception pathway of visual sensory inputs could be interpreted based on the structure of this model. [10-12] In the Moren and Balkenius model, a computational model of emotional learning in amygdala is introduced because the amygdala frequently intervenes in emotional reactions, learning and stimulators of new emotional symptoms, and forms an important part of the learning engine as well as attention. [13- 15]

The computational model of image perception is an engineering model that could be divided into three parts:

1-Primary Vision 2- Intermediate Vision and 3- High Level Vision. In the primary visual layer, a perception of the image regularities, including image texture, edge, slope, or symmetry, is performed. In intermediate vision layer, there are attempts at image analysis to divide the image into objects that are mostly namable. The output of the intermediate vision layer is a set of attributes that are extracted from the original image. These should contain data about the main structure of the body and be as constant as possible for different samples of a classification, as well as differing from each other for different classifications. [10, 11] In the high visual layer, the result of the previous stage is entered as a set of quantitative attributes that represent the object in a specific way. After reducing the amount of data about each object,

while the main and important data is preserved, the correct classification of data is done. [13]

Recently, a class of computational models, termed deep convolutional neural networks (DCNNs), inspired by the hierarchical architectures of ventral visual streams demonstrated striking similarities with the cascade of processing stages in the human visual system. [16-18] Seijdel et al. Used Deep Convolutional Neural Networks as 'artificial animal models for detection of patient with object agnosia. They indicated that DCNNs with 'lesions' in higher order layers showed similar response patterns, with decreased relative performance for manmade scenes and natural. [19]

There is practically no direct treatment for patients with agnosia. Speech therapy or occupational therapy may help to compensate for the illness. In engineering sciences, an attempt has been made to take a step towards intelligent diagnosis and treatment of the disorder by mind modeling of these patients. The perceptual model attempts to provide individuals' perception in order to help patients with difficulty in perception based on these models. [11, 20, 21] According to the agnostic people have difficulty in understanding, a model based on it should be used. Previous research has not paid attention to perception in these people and very little computational model research has been done on agnosia, so here with a tool like FCM that works well in perception, this problem will be solved. So, using the advantages of Moren's computational model and combining it with vision perception pathway, would be solve the problem of image processing in these patients. In this model, Fuzzy Cognitive Mapping (FCM) is used for perception because in FCM, objects and processes are related to values and are modeling methods of complex systems whose origin corresponds to fuzzy logic and neural networks.

This paper is organized in such a way that after the introduction, in the second part, the materials and proposed method are presented. In the third part, the results will be analyzed and in the fourth part, conclusions will be expressed.

2- Materials and Method

The purpose of the present study is to improve the image processing in the minds of people with visual agnosia by the mind modeling using the modified Moren model and Fuzzy Cognitive Mapping (FCM). In fact, in this paper, according to the image perception pathway and considering the problematic part of agnosic individuals, the problem of not processing images properly in the brains and minds of these patients is addressed. In normal individuals, after observing an image, the image information in the visual pathway begins to rise from different regions of the visual cortex of the brain, and this phenomenon is what is known as direct data processing in the brain. Neuronal activities in the visual cortex of the brain begins to rise from the lower regions to the higher regions to reach the stage of image perception. Because patients with agnosia cannot recognize the objects' meaning and attributes, then, the current study uses FCM to compensate this deficiency for image perception in the proposed model.

First, the database images are selected and the appropriate attributes are extracted. Then, based on the model proposed, the read images are perceived and recognized. The proposed model is designed based on the vision pathway and image perception in normal people. So, after explaining the vision pathway and object recognition in normal people, the proposed model and its details will be discussed.

2-1- Object Recognition in Normal Individuals

In all individuals, there are 3 stages of information processing for object recognition:

Stage 1: Receiving visual stimulus; stage 2: object perception; stage 3: object recognition. [22] After receiving the stimulus, our thoughts visualize all the information, in other words, the information is displayed in the form of an image, and then people relate this image to what they knew in the past and perceive its meaning. Humans are able to recognize many objects around them without any problem, although these objects might be in different positions and with different viewing angles as well as different sizes. Even the human is able to recognize objects when he does not see parts of them or another object is in the pathway of his sight. Although this is very simple and practical for humans and mammals, it is in itself a very difficult and complex computational process. [23]

The cerebral cortex is made up of neural cells that make up the outer layer of the brain. It controls complicated activities such as memory, learning, problem solving, planning, sight, hearing, and movement. The cerebral cortex is divided into two hemispheres, each of which is divided into four areas: the temporal, the frontal, the parietal, and the occipital. The occipital part is located at the back and processes visual information.

The visual cortex is divided into different areas. V1 (Primary Visual Cortex) and V2 (Secondary Visual Cortex) are the largest areas and the area of each is about 1100 to 1200 mm. The information received from the retina is located through the LGN from the thalamus to the V1 in two separate pathways and is processed independently.

Object recognition pathway (ventral branch): The function of this pathway is to recognize the objects' attributes (color, shape, etc.) and it is located in pathways of V1, V2, V4, AIT and PIT. In the ventral stream, what is

visualized is the output of what can be consciously seen and described. This stream creates a mental interpretation of the world around us. Color, relative size, texture and shape are all processed in this area. When people observe an object, a mental representation of that object is created, and this representation changes at least semi-permanently to what is referred to as memory. [23]

Object location Recognition Pathway (Dorsal Branch): The function of this pathway is to recognize the spatial characteristics of the scene (direction of movement, etc.) and it is located in the pathways of V1, V2, V3, MT and MST. Dorsal stream processes the visual input, so it is often referred to as the visual motor. Unlike the ventral, the dorsal stream does not form the permanent memories, but provides a continuous update of the recorded information for two seconds. [23] Figure (1) shows the different areas of the cerebral visual cortex.

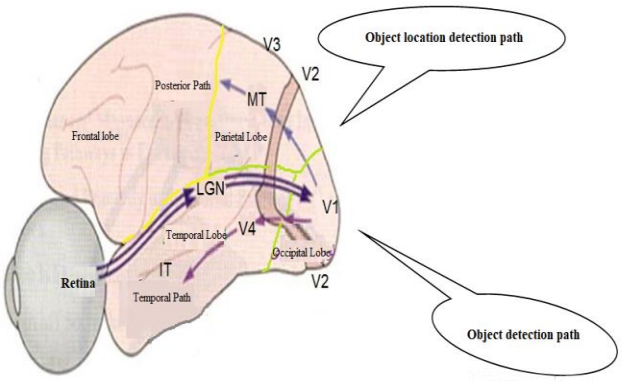


Fig. 1 Different areas of the cerebral visual cortex

The pathway of image perception in ordinary people contains several areas with extensive connections, each of which is responsible for processing part of the raw information collected by the eye. [24]

The Main Visual Pathways: In mapping the visual pathway, Mishkin stated that striate visual areas could be well separated not only anatomically but also functionally [25] including the Tectopulvinar pathway (system for visual motor control), the subcortical pathway including sc and pulv processes of some aspects of unconscious perception, and the Geniculostriate pathway (a system for visual perception). Figure (2) shows the cognitive structure of the visual pathways.

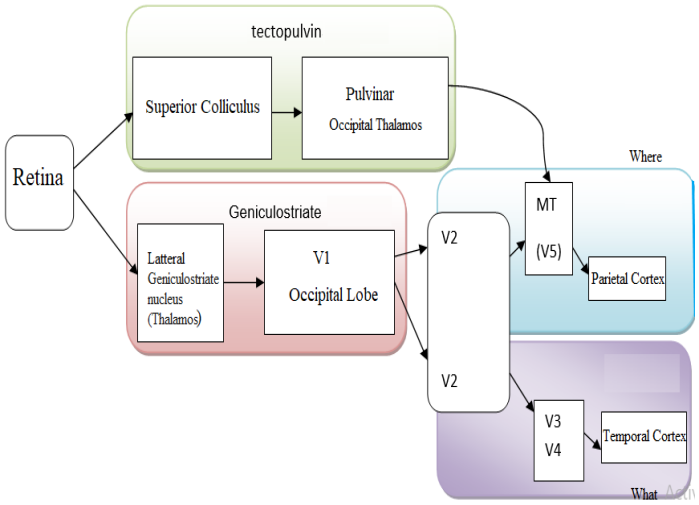


Fig. 2 The cognitive structure of the visual pathways

If the temporal lobe is damaged, these roles will also be impaired. Patients with right or left temporal lobe injuries have difficulty distinguishing and recognizing images, and even skills related to the formation of geometric shapes. Patients with right or left temporal lobe injuries have some difficulties reminding and recognizing objects and remembering their location. Visual information processing in the V4 region is integrated with stored memory patterns. Damage to this pathway can lead to visual agnosia. [25]

2-2- The Proposed Model

The proposed model has been designed based on the human visual model from observation to image perception. Due to the lack of perception of images in these patients, the perception problem in the present model has been solved with FCM. The structure of the proposed model is shown in Figure (3).

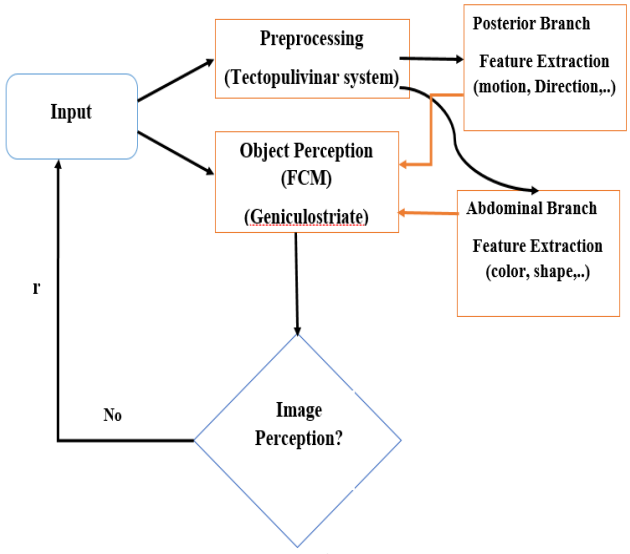


Fig. 3 The structure of the proposed model

As mentioned above, the proposed model, is based on the visual pathway from observing image to perception in the brain. The image is first read as input. The input image in the Tectopulvinar module will be pre-processed (noise cancellation, resolution improvement, etc.). In the posterior and abdominal branch modules, image features are extracted and entered into the Geniculostriate module for image perception. If the image is not perceived, the feedback r helps the system to be trained in perception again. Each of the components of the proposed model will be described below.

Feature Extraction: After selecting the images, the attributes have to be extracted. Here, those faetures with the most important role in separating image pattern classifications are extracted. The features used are described below.

Texture: GLCM are used to extract attribute from tissue. The features obtained in this way are contrast, entropy, energy and homogeneity.

A. Contrast: Contrast measures textures, using the following formula [26].

$$C = \sum_i \sum_j (i-j)^2 C(i,j) \qquad (1)$$

B. Energy: Energy measures the textural uniformity of images and uses the following formula. [27]

$$E = \sum_i \sum_j C^2(i,j) \qquad (2)$$

C. Entropy: Entropy measures the degree of irregularity using the following formula. [27]

$$H = \sum_i \sum_j C(i,j) \log\left(C(i,j)\right) \qquad (3)$$

D. Homogeneity: Homogeneity measures the elements' distribution and uses the following formula.

$$HO = \sum_i \sum_j \frac{C(i,j)}{1+|i-j|} \qquad (4)$$

Color torque**:** Color torques in which the recognition of color distribution in an image is measured in the same way as the unique central torques describe a probability distribution. Color torques are mainly used for color indexing purposes as attributes in image retrieval functions to compare the similarity of two images based on the color. [28]

A. Median: The median, which is the first color torque, could be interpreted as the average color in the image and calculated from the following equation.

$$E_i = \sum_{j=1}^{N} \frac{1}{N} Pij \qquad (5)$$

Where N is the number of pixels in the image and Pij is the i-th pixel value of the image in the i-th color channel.

B. Standard deviation: The second color torque is the standard deviation, which is obtained by considering the second root of the color distribution variance.

$$\sigma_i = \sqrt{\left(\frac{1}{N}\sum_{j=1}^{N}\left(p_{ij}-E_i\right)^2\right)} \qquad (6)$$

Where E_i is the median value, or the first color torque for the i-th color channel of the image.

C. Skewness: The third color torque is skewness. It measures how the color is distributed and then gives information about the shape of the color distribution. Skewness can be calculated from the following equation.

$$s_i = \sqrt[3]{\left(\frac{1}{N}\sum_{j=1}^{N}\left(p_{ij}-E_i\right)^3\right)} \qquad (7)$$

D. Color indexing: The Color torque could be used to compare how two images are similar. This is a relatively new approach to color indexing. Color indexing is the main function of color torque. Images can be indexed and the index will include the calculation of color torque. Therefore, if you have an image and want to find similar images in the database, the color torques of the image in question are calculated. The following function will then

be used to calculate the similarity score between the image in question and the database images. [29]

$$d_{mom}(H,I)=\sum_{i=1}^{r}\omega_{i1}\left|E_i^1-E_i^2\right| +\omega_{i2}\left|\sigma_i^1-\sigma_i^2\right|+\omega_{i3}\left|s_i^1-s_i^2\right| \quad (8)$$

Fuzzy Cognitive Mapping Design (FCM): FCM is a soft computational method for modeling systems and is extensively used for complex systems analysis and decision making. FCM is a directed graph to represent causal relationships between multiple concepts and has been widely applied in various fields to support decision making and task classification. Fuzzy cognitive mapping is a method of presenting efficient knowledge and reasoning that is based on human experience and knowledge and includes experts' opinions about a mental reality and requires data entry and training.

Useful attributes of FCMs such as simplicity, supporting inconsistent knowledge, the field of cause and effect relationship for modeling knowledge and conclusion, as well as learning capability, make them applicable to many different scientific fields of knowledge modeling, forecasting and decision making. In fact, FCM describes specific fields using nodes, concepts (variables, states, inputs, and outputs), causal relationships, and signed fuzzy relationships between them that could be positive or negative. Learning methods are used to train FCMs, which include updating the weights of causal relationships. Fuzzy Cognitive Mapping (FCM) is interpreted as follows. [26]

If FCM is the number of N nodes Ci, the value of each node in each iteration is computed as follows.

$$A_i^{t+1}=F\left(A_i^t+\sum_{j=1}^{n}A_j^t w_{ji}\right) \quad (9)$$

Where A_i^t is the value of the concept Ci at time t + 1, A_j^t the value of the concept Ci at time t, Wji corresponds to the fuzzy weight between the two nodes and F is the threshold function that converts the product to a number in the interval [0-1]. The function used in the present study is the logistic function given in the following equation.

$$F(x)=\frac{1}{1+e^{-\lambda\chi}} \quad (10)$$

Each node is a fuzzy set that could be excited from 0 to 100%. FCM connects objects and processes to values. The FCM modeling method is consistent with fuzzy logic and neural networks that grow in feedback. First an FCM is early valuated, then the activation level of each node takes a certain value of the system and different concepts are free in interaction. Activating one node affects the other nodes to which it is connected. This continues until the system reaches a constant equilibrium point or a finite cycle or a turbulent behavior. [30]

3- Discussion and Results

The test uses a 500 k Corel database that includes five classifications: Africa, beach, buildings, buses and dinosaurs. After calling the images, the features extracted from the images and the training is performed. In the test stage, the type and classification of the images will be recognized. To determine the efficiency of the proposed model, the criteria of accuracy, precision, recall and F1measure have been used. The following formulas show how to calculate them.

$$Accuracy=\frac{T_p+T_n}{T_p+F_p+F_n+T_n} \quad (11)$$

$$\mathrm{Pr}ecision=\frac{T_p}{T_p+F_p} \quad (12)$$

$$\mathrm{Re}call=\frac{T_p}{T_p+F_n} \quad (13)$$

$$F1\,Score=\frac{2\times \mathrm{Pr}ecision\times \mathrm{Re}call}{\mathrm{Pr}ecision+\mathrm{Re}call} \quad (14)$$

Where Fn is unrecalled related images, Tp recalled related images, Tn unrecalled unrelated images and Fp recalled unrelated images. Table (1) shows the comparison of the accuracy value of the proposed method with SVM and neural network methods in the number of different images.

Table 1: Comparing the accuracy of recognizing different types of images with the number of different images in the proposed model and other methods

Number of Images	***100***	***200***	***300***	***400***	***500***
Proposed Method (FCM)	82%	85.30%	86.90%	94.2%	98.1%
SVM	67%	69.4%	73.1%	86.3%	91.4%
Neural Network (MLP)	42%	58.2%	63.5%	78%	81.4%

As it could be seen, the proposed model performs better for more images, and with 98.1% accuracy has succeeded in recognizing objects while SVM, with 91.4% and neural network with 81.4% accuracy have recognized. Figure (5) shows the comparison of the performance of proposed model with other methods based on the accuracy, precision, recall and F1 measure.

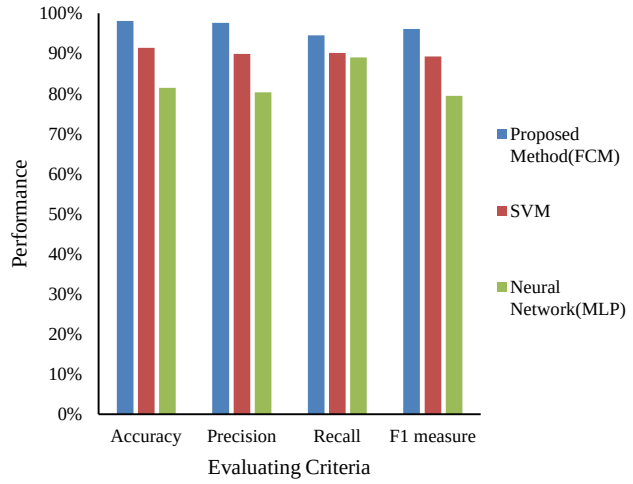


Fig.4. comparison of the performance of proposed model with other methods

As it is clear in the above diagram, the proposed method has performed better. Figure (6) shows the comparison of the performance of proposed model with the Moren mind, Balkenius model and CNNs.

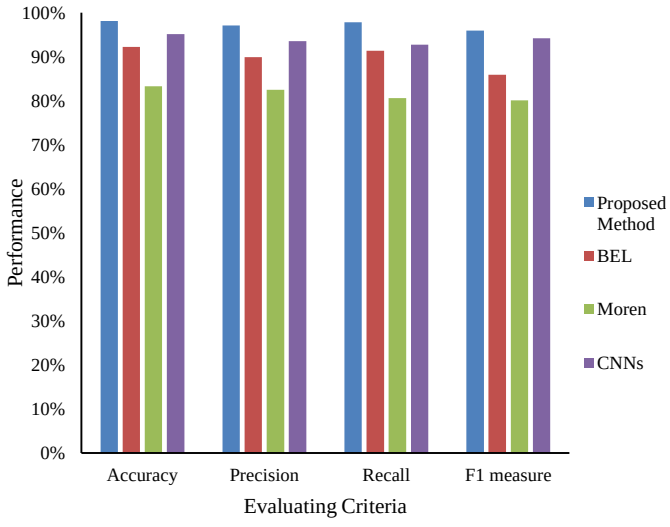


Fig.5: comparison of the performance of proposed model with other methods

As can be seen in the above diagram, the proposed model performed better than the other two methods. The accuracy of the proposed model was also compared with the Moren, Balkenius model and CNNs in different number of images and have been reported in Table (2).

Table 2: Comparing the accuracy of recognizing different types of images with the number of different images in the proposed model, BEL and Moren

Evaluation Criteria	***100***	***200***	***300***	***400***	***500***
Proposed Method	84.1%	87.30%	89.30%	94.6%	98.1%
BEL Model	65.1%	67.4%	76.2%	86.9%	92.2%
Moren Model	44.1%	59.2%	68.2%	780.1%	83.3%
CNNs	81.9%	82.6%	87.3%	92.1%	92.1%

As it could be seen, the proposed model has been able to have higher accuracy than other mind models due to the use of fuzzy logic.

4- Conclusions and Future Work

In this paper, a model was proposed to solve the problem of recognizing objects in patients with visual agnosia. In the proposed model, the human visual system was inspired by the structure of the visual cortex of the brain. In this model using FCM, intelligently helped to improve the perception of images in the minds of these patients. Perception was performed by recognizing the meaning and relationships between the objects in the seen image using extracting the features of contrast, energy, entropy, heterogeneity, color torque, median and standard deviation from the images. The proposed model reduced the difficulty of perceiving and recognizing objects in patients with visual agnosia. The accuracy criterion was compared in the proposed model with other models and it was revealed that the proposed model with 98.1% had a higher accuracy compared to SVM and neural network methods.
In the future, the accuracy of the proposed computational model can be improved with the help of new artificial

intelligence and image processing tools. For example, adding the capabilities of deep and convolutional neural networks to the current computational model can modify the performance of this model. In addition, other options can be added such as early diagnosing the disease using electroencephalography data. The proposed model is a simulation of a system that can help people with visual agnosia to understand and recognize images. In the second and practical phase of this model, a tool should be developed to help these people to identify the images.

References

[1] M.J. Yang, T.J. Nail, J. Winer, "Left Parietal Tumors Presenting with Smartphone Icon Visual Agnosia: Two Cases of a Modern Presentation of Gerstmann Syndrome", World Neurosurgery, Vol.142, 2020, pp. 233-238.

[2] N. Seijdel, H.S. Scholte, E.H.F.de. Haan, "Visual features drive the category-specific impairments on categorization tasks in a patient with object agnosia", Neuropsychologia, Vol.161, 2021.

[3] MJ. Farah," Visual agnosia: Disorders of object recognition and what they tell us about normal vision", MIT Press, American Psychological Association, 1990, pp: 322-329.

[4] P.S. Vecera, S.K. Gilds, S, "What processing is impaired in appreceptive agnosia? Evidence from normal subjects", Journal of Cognitive Neuroscience, Vol. 10. NO. 5, 1998, pp.568–580.

[5] M. Goodale, AD. Milner AD, "Separate visual pathways for perception and action",Trends Neurosci, Vol. 15. No. 1, 1992, pp.5-20.

[6] E. Golderg,"Associative Agnosias and the functions of the left hemisphere",Vol. 12, 1990, pp: 467-484.

[7] R. Cecere, N. Dundon, C. Bertini,"When apperceptive agnosia is explained by a deficit of primary visual processing, Elsevier Ltd., 2014, pp:12-27.

[8] L. Iterson, L. T.L.Sie, P.B. Augustijn, A.C.S. Rooze, F.E. Jansen, "Acquired visual agnosia as an uncommon presentation of epileptic encephalopathy in a 6-year-old boy with CSWS", Epilepsy & Behavior Reports, 2021, Vol.16.

[9] K. Erikssona, A. Inenb, K. Hirvonenc, P. Nieminena, M. Koivikko,"Visual agnosia in a child with non-lesional occipito-temporal CSWS", Brain and Development, Vol.25, No.4, 2003, pp.262–267.

[10] E. Askari, S.K. Setarehdan, A. Sheikhani, M.R. Mohamadi, M. Teshnelab, "Modeling the connections of brain regions in children with autism using cellular neural networks and electroencephalography analysis", Artificial Intelligence in Medicine, Vol. 89, pp.40-50.

[11] R. Ptak, F. Lazeyras, "Functional connectivity and the failure to retrieve meaning from shape in visual object agnosia", Brain and Cognition, Vol.131, 2019, pp.94-101.

[12] X. Duanmu, "Image Retrieval Using Color Moment Invariant", in Seventh International Conference on Information Technology,2010, pp: 200-203.

[13] F. yaghouti, S. motamed. "Recognition of Facial Expression of Emotions Based on Brain Emotional Learning (BEL) Model", Advances in Cognitive Science, Vol. 20, No. 4, 2019, pp.46-61.

[14] J. Morén, "Emotion and learning- a computational model of the Amygdala", Ph.D. thesis, Lund, Sweden: Lund University, 2002.

[15] E. Lotfi, "Mathematical modeling of emotional brain for classification problems, Proceeding of IAM", Vol. 2, No. 1, 2013, pp.60-71.

[16] S.M. Khaligh Razavi, N. Kriegeskorte, "Deep supervised, but not unsupervised, models may explain IT cortical representation", PLoS computational biology, Vol. 10. No. 11, 2014.

[17] R.M. Cichy, A. Khosla, D. Pantazis, A. Torralba, A. Oliva, "Comparison of deep neural networks to spatio-temporal cortical dynamics of human visual object recognition reveals hierarchical correspondence", Scientific Reports, Vol. 6, No. 27755, 2016.

[18] U. Güçlü,, M.A.J. van Gerven, "Deep Neural Networks Reveal a Gradient in the Complexityof Neural Representations across the Ventral Stream", The Journal of Neuroscience, Vol. 35, No. 27, 2015, pp. 10005-10014.

[19] N. Seijdel, H.S. Scholte, E.H.F. de Haan. "Visual features drive the category-specific impairments on categorization tasks in a patient with object agnosia", Neuropsychologia, Vol. 161, 2021.

[20] H. Karnath, J. Rüter, A. Mandler, M. Himmelbach, "The anatomy of object recognition—Visual form agnosia caused by medial occipitotemporal stroke", The Journal of Neuroscience", Vol. 29, No. 18, 2009, pp: 5854–5862.

[21] B. Kolb, I. O. Whishaw, "Fundamentals of Human Neuropsychology 6th ed", New York, NY., Worth Publishers, ISBN 978-0-7167-9586-5, 2009,pp:325-331.

[22] H. Lissauer,"A case of visual agnosia with a contribution to theory", Cogn Neuropsychol, Vol.5, 1988, pp.153–192.

[23] B. LFinlay, "Focusing the eyes and recognizing objects: evo-devo and the sensitive period", Current Opinion in Behavioral Sciences,Vol. 36, 2020, pp.36-41.

[24] H. Sufikarimi, K. Mohammadi, "Role of the secondary visual cortex in HMAX model for object recognition", Cognitive System Research, Vol. 64, 2020, pp.15-28.

[25] M. Mishkin, L. G. Ungerleider, K. A. Macko, "Object vision and spatial vision: Two cortical pathways", Trends in Neurosciences, Vol. 6, 1983, pp.414-417.

[26] P. Lazzaro, D. Murra, B. Schwortz, "Pattern recognition after image processing of low-contrast images, the case of the Shroud of Turin", Pattern Recognition, Vol. 46, No. 7, 2013, pp.1964-1970.

[27] S. Etemad, M. Amirmazlaghani, "Color texture image retrieval based on Copula multivariate modeling in the Shearlet domain", Engineering Applications of Artificial Intelligence, Vol. 102, 2021.

[28] C. Singh, E. Walia, K. PreetKaur, "Color texture description with novel local binary patterns for effective image retrieval", Pattern Recognition, Vol.76, 2018, pp.50-68.

[29] J.K. Kamarainen, V. Kyrki, H. Kalviainen, "Invariance properties of gabor filter-based features-overview and applications", IEEE Transactions on Image Processing, Vol. 15, 2006, pp.1088-1099.

[30] EP. Georgiou, A. Kannappan, "Fuzzy cognitive map ensemble learning paradigm to solve classification problems: Application to autism identification", Applied Soft Computing, Vol. 12, 2012, pp:3798-3809.

Journal of Information Systems and Telecommunication
Vol.11, No.2, April-June 2023, 110-122
http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

Performance Analysis and Activity Deviation Discovery in Event Log Using Process Mining Tool for Hospital System

Shanmuga Sundari M[1], Rudra Kalyan Nayak[2*], Vijaya Chandra Jadala[3], Sai Kiran Pasupuleti[4]

[1]. Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation
[2]. School of Computing Science and Engineering, VIT Bhopal University, Bhopal-Indore Highway, Kothrikalan, Sehore, Madhya Pradesh - 466114
[3]. Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation
[4]. Department of CSE, Prasad V Potluri Siddhartha Institute of Technology

Received: 31 Aug 2021/ Revised: 01 May 2022 / Accepted: 28 May 2022

Abstract

All service and manufacturing businesses are resilient and strive for a more efficient and better end in today's world. Data mining is data-driven and necessitates significant data to analyze the pattern and train the model. Assume the data is incorrect and was not collected from reliable sources, causing the analysis to be skewed. We introduce a procedure in which the dataset is split into test and training datasets with a specific ratio to overcome this challenge. Process mining will find the traces of actions to streamline the process and aid data mining in producing a more efficient result. The most responsible domain is the healthcare industry. In this study, we used the activity data from the hospital and applied process mining algorithms such as alpha miner and fuzzy miner. Process mining is used to check for conformity in the event log and do performance analysis, and a pattern of accuracy is exhibited. Finally, we used process mining techniques to show the deviation flow and fix the process flow. This study showed that there was a variation in the flow by employing alpha and fuzzy miners in the hospital.

Keywords: Alpha Miner; Event log; Fuzzy Miner; Hospital Process; Process Mining.

1- Introduction

In hospital systems, the activities are more volatile and dynamic and affect the efficiency and productivity that are the keys to survival and flourishing in the industry. Business Process Management (BPM) [1] is the basic foundation that efficiently produces the process, which will lead to more minor failures. BPM is an effective method to solve the emotional problems much early in the process activities. The root cause analysis [2] is the best technique to overcome the problem and improve process performance and efficiency. It gives visibility, accountability, and scalability in solving the operational issues and problems in the production or process.

Nowadays, many industries and enterprises are using business process management to construct their technique and improve the operations in process-centric flow, by extension, their efficacy in their field. Process Mining is the central part of the business process management approach. It will extract the input from the event logs collected from the hospital front desk and construct the process flow using this event log. The event log contains all traces of the activities in each flow. The process model is derived from the event log, and further, the model is used to analyze the different categories of the process. Process mining will streamline the related subsequent activities to help the process successfully and reduce the overhead of the process.

Process Mining [3] is the technique that explores all the activities in the process and trains the process model based on the dataset activities executed using the process mining algorithms. It shows the patterns bringing transparent results and delivers the graph with time constraints based on factual evidence and insights from the event log. These are automated process mining algorithms that can produce better and more accurate results when compared with the manual data mining of the process. It is being implemented as how the process ought to train a model.

The majority of the research in the literature is focused on organizational process mining. Medical treatment

✉ **Shanmuga Sundari M**
mshanmugasundari@gmail.com

processes refer to the clinical activities that are responsible for the care of patients. In contrast, administrative functions refer to the total activities carried out within an organization to ensure that successful medical treatment procedures are implemented.

On the other side, there is another set of underlying issues in the healthcare system that is mainly invisible and overlooked. The lack of consistent healthcare systems across the country is mostly to blame for these intangible issues. There is no defined technique to track and control the patient journey process, to put it another way. A patient's journey refers to the entire treatment process, which begins with a consultation with a doctor and ends when the patient has received all necessary therapy, and the case is considered finished. Because many patients are treated in specialized units creating a different process model for each care unit.

The use of process mining in a typical patient treatment procedure in a hospital is demonstrated in this study. We provide a viable solution to tackle most of the visible problems in the healthcare sector by solving the invisible problems utilizing process mining tools and techniques in this work. We used different algorithms to check the conformance of the process. The algorithm shows the more deviation that will find the time taking activity in the entire process.

2- Literature Review

Process mining is the best technique to find patterns [4] from the automatic extraction of the process models, which the process can implement to analyze the process. It is also used to define the conformance checking [5] with the business flow and identify the next level improvements. Business process analysis follows the below perspectives:

1. Data flow: Discover the data flow in the process and produce the pattern based on the data analysis.
2. Process flow: Discover the activities in the process and discover the model for the process; based on this, the process model is enhanced.

We are implementing the process flow in the hospital domain for the above insights with ProM Tool [6]. The information gained from the hospital is changed to the required format of process mining and then derives the conformance checking in emergency activities [7].

Van Der Aalst et al. [8] used the event log activities in the workflow process in water and road maintenance in the Netherlands to find the accuracy and measure the performance of the process from the organizational and case perspectives. The author significantly impacted the performance in terms of all views. He analyzed the outcome of the process in specific benchmarks of the domain variance. By using these activities, the author found abnormal behaviors.

R. Tripathy et al. [9] used a fuzzy C-Mean algorithm to identify the process's prediction. The root cause analysis is demonstrated and incorporated with the practical level. Process mining is used to enhance the process and rerun the flow to correct the process flow. The survey [10] showed that unplanned disruptions and uncertain changes would affect the performance of the process. Uncertainty of the changes will always affect the resource and operational issues in the industry.

All traces of activities in the process must be validated and identified based on the aspects of data evaluation [11]. The results are interpreted, and an assessment of the outcomes is deployed to furnish the following enhancement and future investigations.

The management lacks visibility in a few areas, such as process accuracy, quality, and resource utilization. The mining process will visualize the patterns with deviations which shows the data visualization as well as aiding exploration and automation [12]. The result provides the visualization of the process, and enhancement [13] will be carried out using these deviations to improve the process to the next level.

Petri net [14] is the graphical description of the activity flow and process flow structures, showing the dependencies between the activities [15]. The alpha algorithm proposed extends the process mining to the invisible tasks using the classification algorithm [16] in the hospital and completes the process.

Md Junayed Hasan et al. [17] explained the time-frequency imaging technique using machine learning classifications. It focused on the image's time frame and did the analysis. By evaluating the electroencephalogram (EEG) [18], data of polysomnography (PSG) recorded for three regions of the human brain, namely the prefrontal, central, and occipital lobes, a classification framework for automatic sleep stage recognition in both male and female human subjects was developed. The residual neural network (ResNet) architecture is used to automatically learn the characteristic features of different sleep stages from the raw EEG data's power spectral density (PSD) without using any artifact removal techniques. ResNet's residual block uses EEG data to learn the fundamental properties of various sleep stages while avoiding the vanishing gradient problem.

To obtain more accurate results, an optimum multilevel thresholding hybrid method is combined with Genetic Algorithm (GA) [19] and Particle Swarm Optimization (PSO), named HGAPSO, with an optimization strategy for classification based on grey level range. Process mining is another technique which works before all of the above machine learning methods.

3- Proposed System

The proposed system is a process mining tool to identify the event log from the hospital data, and change the event log into a structured format. The process mining tool will create the model based on the automated algorithms, and the base model will iterate with the event log and produce the different patterns, which will guide us to find the deviation in the process. Fig.1 shows the framework of our proposed model in process mining. The conformity checking is reviewed using the algorithms, and the process enhancement is concluded with the development of discovery model.

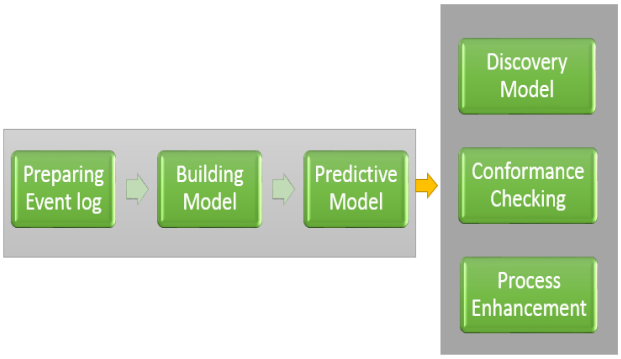


Fig. 1 Framework of proposed process mining

4- Process Mining Algorithms

4-1- Alpha Miner

An event log [20] consists of multiple traces. A trace [21] is a sequence of activity names abstracted from other attributes in the order of events.
The sample event log is:

$$T1 = [\langle a, b, c, d\rangle^3, \langle a, c, b, d\rangle^2, \langle a, e, d\rangle] \quad (1)$$

Eq. (1) gives the sample event log that explains the sequence of activities in the single trace *T*1. It is the order of series and number of times executed in the activity trace. The activities are *a*, *b*, *c*, and *d* in different patterns means the order of *a*, *b*, *c*, and *d* are executed three times, and *a*, *c*, *b*, and *d* are executed two times, and *a*, *e* and *d* executed one time. The primary purpose of the process mining is to retrieve good possible paths, for instance, that produce Petri net [22]. The Petri net starts the Activity with *a*, followed by *b*, *c*, and ends with activity *d*. Fig. 2 shows the Petri net of the above activity trace *T1*. Similarly, our hospital data set will evaluate the entire traces and analyze the pattern. The system is enhanced using the model-based result to refine the new model.

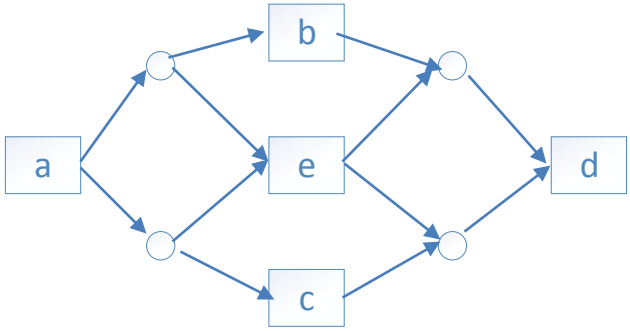


Fig. 2 Petri net for Trace T1

Fig. 2 explains the detailed Petri net of the trace *T1*. The graph starts with Activity a and ends with *d* activity. Based on the traces, the dependency [23] matrix is evaluated, and the order of the relation is calculated based on the dependency matrix.
Rule 1 (Predicting the followed Activity): In *the T1* trace, the flow of activities has to be traced based on the sequence order.
Rule 2 (Predicting entire Traces): In *T1*, the whole activities are followed until the end of the process, and the model is predicted based on all the activities.
Once the transition structure is obtained, the Concurrent Transition Sequence (CTS) is completed. Then the sub-trace log is analyzed for the entire flow, which helps find the trace's significance in the whole hospital dataset—the Petri net result in the reduced structure of input and output functions in the concurrent system.
In the below example, Trace *T2* has a few activity flows ranges *t1* to *t7*. From this, the sub trace is identified using the conditions below.
For example, *T2 = {< t1, t2, t3, t4, t6, t7 >, < t1, t2, t4, t3, t6, t7 >, < t1, t2, t5, t6, t7 >}*, and *N2* is a Petri net model with a concurrent structure. We have CTS= *{t3, t4, t5}*, $\sigma 1|CTS =< t3, t4 >$, $\sigma 2|CTS =< t4, t3 >$, $\sigma 3|CTS = < t5 >$, and the sub-trace log is T_{sub} = *{< t3, t4 >, < t4, t3 >, < t5 >}*.

4-2- Fuzzy Miner

The Fuzzy Miner [24] is different from other algorithms since it will not produce a graph like the Petri net and causal net [25]. The Fuzzy Miner can work on the dynamic structure. The Fuzzy Miner is applicable for semi-structured data in the extensive database. The problem with process graphs is that we cannot differentiate between choice and parallelism. The complex and uncertain activities will happen in the actual time process. The unstructured process [26] is complicated to find the flow. The traditional process cannot support the unstructured process in outpatient discovery [27]. The model discovered in a fuzzy miner is a highly complex net,

and it is called a spaghetti diagram [28]. A fuzzy algorithm is a road map of metaphor.
When less structured activities are available in the dataset, it describes the view in high-level undesired details.
Activities based on the hospital department involved in the treatment and timestamps [29] connected to entering and exiting the event.
Activity in the event log has precedence [30] connection to the department and a timestamp.

Precedence Diagram: Let $W = (E, I, A, C, t, i, a, c)$ be an event log and $S = (N, L)$ and SPD. We say that $Sc = (W, S, la, ln)$ is a *connected SPD*, where $la: A \rightarrow P(N) \setminus \emptyset$ and $ln: NP(A) \setminus \emptyset$, such that for all $a \in A$ and $n \in N$ holds that $n \in la(a) \equiv a \in ln(n)$.

These precedence relations will provide the bonding between the activities identified as a significant role in the traces. The precedence matrix illustrates the value based on the connectivity of the activities.
The main goal of the fuzzy miner algorithm is to cluster the activities that are in the same sequence. These activities are further divided into multiple observations to sense the trace. The main idea behind fuzzy clustering is to find the similarity metrics of the activities. We can choose the number of clusters in the fuzzy miner algorithm to maximize the similarity and minimize the complexity.

5- Experimental Setup

5-1- Dataset

We collected the 1000 Traces dataset [31] from the hospital. In the treatment of patients, each trace will have a variety of actions. Within the traces, the activities are denoted by the letters tXX. The XX values are assigned to each hospital department. We displayed a few department numbers in Table 1.

Table 1. Example Activities from the Event log

Sl. No.	*Trace number*	*Department name*
1	*t21*	X-Ray
2	*t26*	Scan
3	*t31*	Blood test
4	*t41*	Consultation
5	*t51*	Nurse care

The related departments are considered based on the patient's treatment. The activities flow will be different, and the trace is illustrated according to their treatment.

```
<trace>
    <string key="concept:name" value="trace 1"/>
    <event>
        <string key="concept:name" value="t11"/>
    </event>
    <event>
        <string key="concept:name" value="t21"/>
    </event>
    <event>
        <string key="concept:name" value="t31"/>
    </event>
    <event>
        <string key="concept:name" value="t41"/>
    </event>
    <event>
        <string key="concept:name" value="t51"/>
    </event>
    <event>
        <string key="concept:name" value="t61"/>
    </event>
    <event>
        <string key="concept:name" value="t26"/>
    </event>
    <event>
        <string key="concept:name" value="t44"/>
    </event>
    <event>
        <string key="concept:name" value="t54"/>
    </event>
    <event>
        <string key="concept:name" value="t65"/>
    </event>
```

Fig. 3 Example trace in the Event log

XES (eXtensible Event Stream) [28], applicable in the ProM tool. The event log's XML (eXentisible Markup Language) format is displayed in Fig.3, including all the traces of events. There are 1000 traces in the event log converted in the XML format. Then this format is changed [29-32].

Fig. 4 Event Dashboard

The dashboard of the activities is shown in Fig. 4. The dataset has 1000 cases and 15995 actions. After noise removal from the dataset, this is the XES file that will be ready to send as input to the process mining. The events are consolidated as a trace in the next step in the mining. The dataset has split 70:30 ratio for train and test the model. The dataset is divided into categories based on how the actions occur. The data is organized into traces, thought of as activity splits. With deviating traces and typical traces, the trace split has occurred.

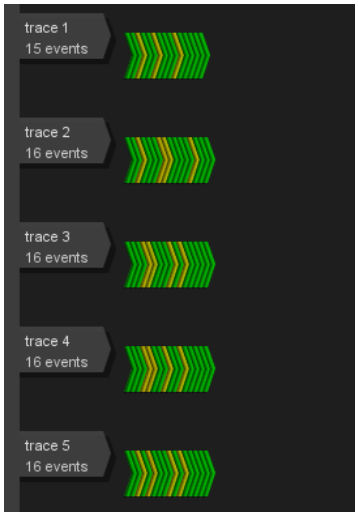


Fig. 5 Activity traces

Fig. 5 displays the activity traces. XML file is the input of the ProM tool then the tool changes the input file to XES format. This stream of activities is called traces which is shown in Fig.4. Hereafter, these traces are the input of the process mining algorithms. All 1000 traces are implemented and ready to apply as input in the algorithms. The green color lines are high-frequency events, and the yellow color lines are low-frequency events. All these events are connected and considered a single trace for a patent treatment flow. For example, in trace 1, 15 events have three low-frequency events. These low-frequency events have approximately 45 to 55 percent. The remaining other activities are 85 to 99 percentage frequency values.

Trace alignment can be represented as a matrix T= {activity i, j} Minimum number of traces <=i, j >= maximum number of traces. The possible traces can be satisfied in the above relation to form a trace alignment. If there are any gaps between the activities, they can be identified using $\Sigma' \; a \cup \{-\}$. The symbol represents the gaps between the activities that do not connect with other activities.

In this paper, the trace alignment is taken, combining all cases in the hospital data using a sequence trace approach. We can use the multiple activities in the event log to diagnose the sequence alignment of all activities. Numerous sequence traces are combined and produce a final trace diagram, as shown in Fig. 6. There are 1000 traces formed according to the sequences. We found one series with four traces of similar flow of lines in activities. Likewise, there are two sequences of 3 traces and three of 2 traces found in the sequence flow. The remaining traces occurred in one sequence flow. There are significant categories also mentioned in the traces. It is proposed to use this approach to find the deviations and understand the traces.

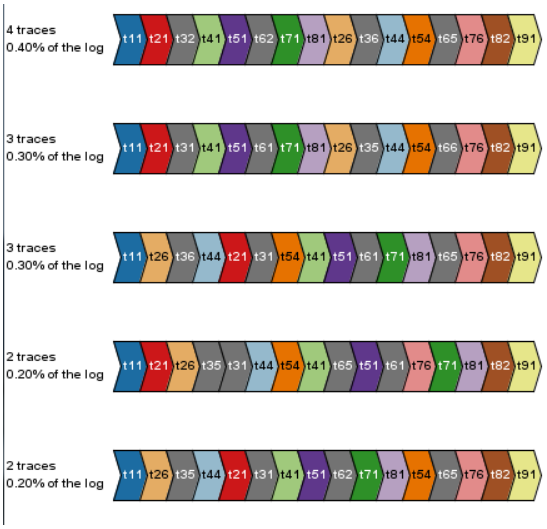


Fig. 6 Traces of Hospital data

In Fig. 7, we drilled down the first sequence flow with four traces. The same activity flow happened four times in the hospital sequence. Fig.7 shows the four activity sequences in the consequent order of activities. Likewise, the traces are designed based on a series of activities. In Fig. 7, the second trace has three similar activity flows. It shows the movements in a trace diagram that can help quickly understand all the flows in the hospital system.

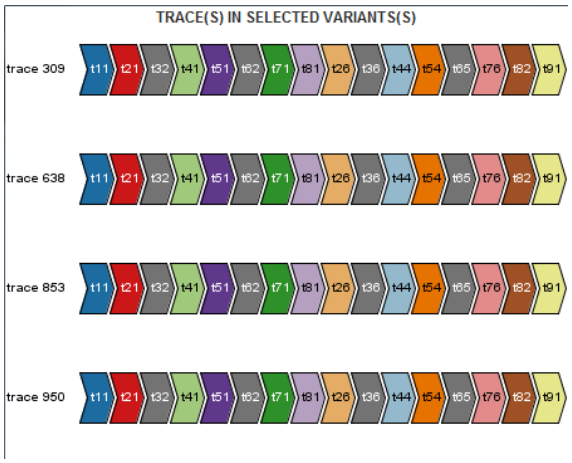


Fig. 7 Selected first trace

5-2- Pattern Evaluation

Fuzzy miner process mining: Conformance checking techniques can quantify the different deviations. For each deviation, the "process model may be wrong," or the "event log (i.e., real data) may be wrong."In the context of

compliance and auditing, deviations are often considered undesirable. The term "normative model" refers to negative deviations. However, there are many possibilities of non-conforming activities in the traces, i.e., not affecting the performance of the process, called positive deviants or successful exceptions.

The term "positive deviance" refers to the uncommon process, but successful behaviors can be considered in the process diagram. Irregularities in processes are sometimes needed (like breaking glass to pull a fire alarm). Also, in hospitals, doctors save lives daily by deviating from the medical guidelines. However, flexibility does not imply that it is not valuable to investigate deviations and learn from them, e.g., to change procedures or enforce controls.

Fuzzy miner in hospital dataset:

In Fig. 8, event log traces are applied to the fuzzy miner algorithm that shows the flow of each Activity in the traces. The dark and bold line shows massive activity flow in the entire system, and the lighter color line shows less activity flow in the hospital. The more chances of bottleneck will be possible in the high activity movement path in the process.

This simplified process model shows the darker nodes representing the cluster of less significant activities. Every node in the diagram is rendered with a dependency matrix value. For example, node *t51* shows a value of 0.503, representing this trace comprises 50% of the total traces connected with this node *t51*. When the flow comes to *t54*, the value of *t54* is 0.523. Then the flow is split into two different traces like *t66 and t65*. The total value of 0.523 is divided into two different values that are 0.268 for *t66* and 0.254 for *t65*.

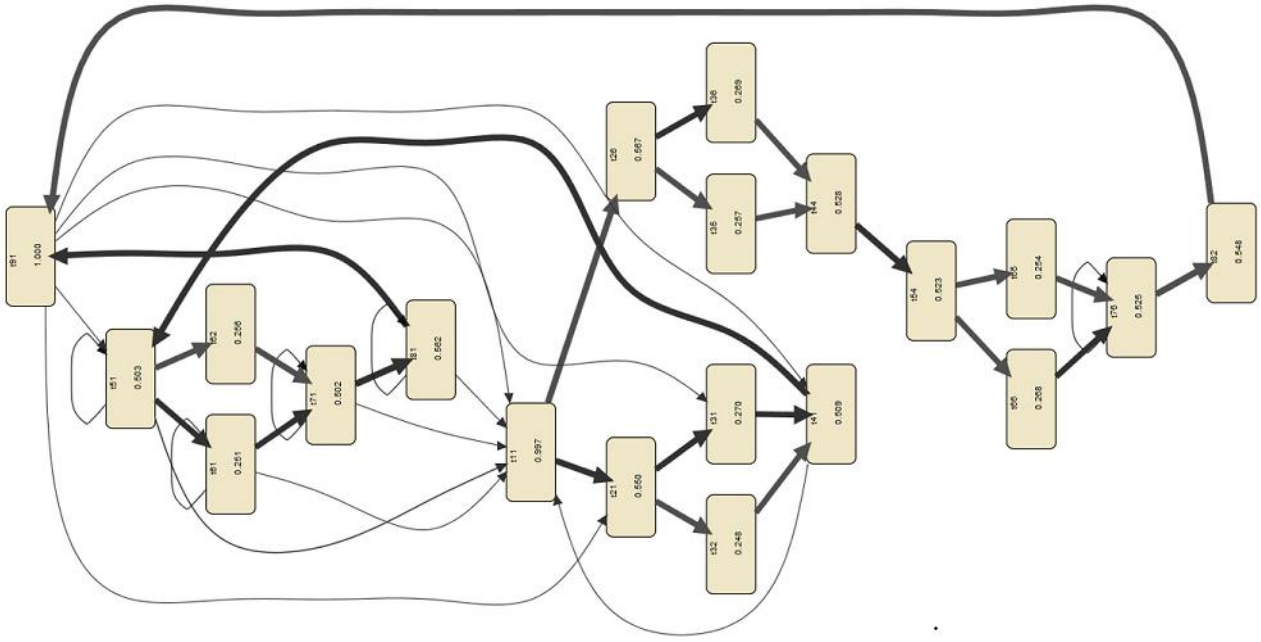


Fig. 8 Fuzzy miner for Hospital dataset

Fig. 8 shows a causal net flow diagram in the simplified process model. Bright square nodes represent significant activities, and the darker nodes are the primary node where all others are connecting through from this node. It is identified as a high significance node in the causal net diagram. All the nodes are denoted clearly with the name and probability of occurrences in the flow. The cluster node is more responsible for connecting all other nodes marked in a darker line. The edges are also labeled with proper correlation values and differentiated with color notation. When we click on the nodes, it will show the importance of the nodes, and we can visualize the connectivity between nodes.

Fuzzy miner is one of the algorithms to animate the process using the event log with various representations. The event log is the primary key point to generating an animation. If this is the case, the spirit will produce the results connecting with the other attribute relations. The features are more dynamic in the hospital dataset, which relies on the various changes and challenges in the resultant diagram.

Cluster in Fuzzy Miner:

Fig. 9 displays the most significant clusters in the hospital dataset. Node t11 is the starting node, and t91 is the ending node. In between, there are two clusters formed due to the lack of resources in the hospital. Cluster_27 was created for five element traces, and cluster_28 was developed for 13 element traces. It is an essential cluster to consider in critical resources in the hospital dataset. The process mining displays the timestamp deviation across the complete sequence of actions. Many patients have gathered around nodes 27 and 28, and they must wait for their acts. It could be due to a shortage of resources to attend the activity or a poor resource working culture. That is why the cluster forms between those two nodes. These two nodes require extra attention from management.

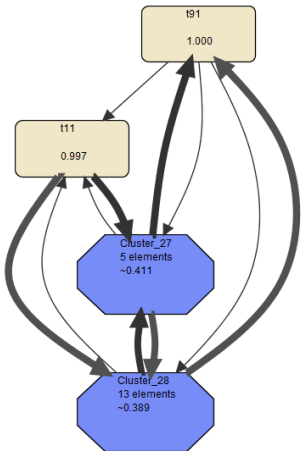


Fig. 9 Cluster of Fuzzy miner for Hospital dataset

Alpha Miner Process Mining:

Finally, Petri net diagram is drawn using the alpha miner algorithm shown in Fig.7. In Alpha miner, the event traces are connected using XOR and AND connectors. The possible flow of activities connects according to the flow of patient's treatment. Trace starting node with the patient registration at the front desk in the hospital and ending activity may differ based on the treatment.

In Fig.10, the basic structures are created with XOR and AND that use split and join conditions to generate a loop structure called a spaghetti graph in the Petri net. The connectivity with the activities is synchronized with non-exclusive conditions connected. The workflow of Petri net represents the business flow with the alpha algorithm. Petri net is a network that was initially considered static. Then the changes happened dynamically during the transition. But in the Petri net, all activities are considered tokens. So the reply tries to make the place changes in the node structure. We will try to replay the transition to eliminate deviation in the process. Circles in the diagram represent the places, and the squares represent the transitions. In this diagram, Activity is the base attribute. All features are designed according to the time stamp based on the base attribute remaining.

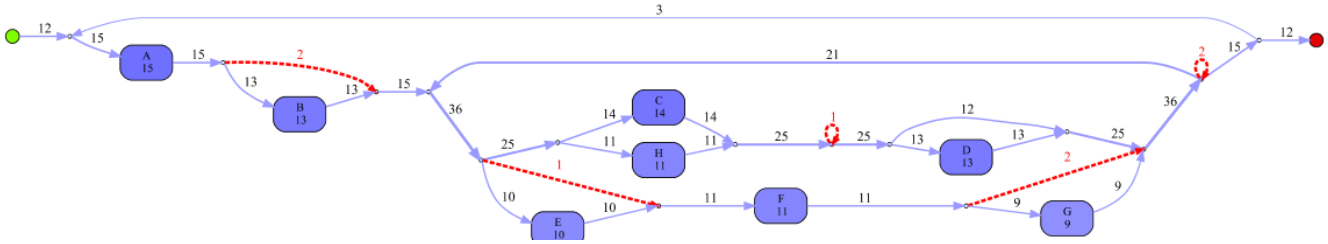


Fig. 10 Petri net with transition

The flow of state determines the waiting stage of activities. If the node is waiting for a particular resource long, that will create a low performance. First, transition results in a state with the red color mentioned as the conjunction of evolution is shown in Fig.10. If the network is static, then the movement is free to flow. The form in a Petri net is called the marking. From the initial level to the inactive status of transition has to be identified to make the node flow in a better chance. It is called reachable marking. That means it can access all activities without missing any action in the flow.

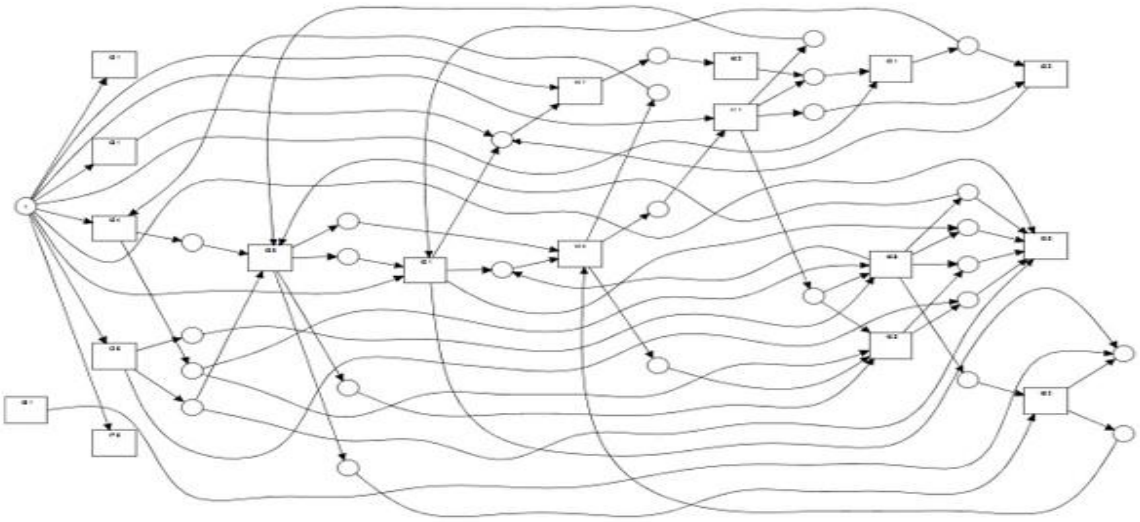


Fig. 11 Petri net diagram for hospital data

If t11 starts the trace from that, multiple other traces can connect based on patients' travel to the hospital. So by this trace, all activities illustrate that the connectivity of tokens may change. In this case, Petri net generates a complete behavior pattern that will express the dependency between nodes or activities. And that is what we need to travel the entire model in the Petri net and find the required abstraction of the Petri net. Now we can look a little deeper into the Petri net and analyze the complexity of the design.

The ProM plug-ins are used to analyze different abstract levels of the result in the hospital dataset. So, if we give the examples that will judge the attributes suitable for the tool and provide the suggestions.

Based on that, we can choose the visualization report, and then it will show the result in graphical representation. Some plug-ins are used to find the entire flow structure and replay activities that will repair the system in terms of enhancement. Sometimes the activities are divided into multiple activities and found whether the sequence flow is correct or not. Moreover, label splitting is also used to find the extensions and refinement approach. ProM uses these extensions to guarantee accurate results.

We displayed the nodes that indicate frequent activities flow, as shown in Fig.11. The previous Petri net offers the entire traces flow called a spaghetti diagram. Process mining can reduce this complicated flow further with a more significant activity sequence. ProM tool has the slider to move the frequency level from low to high. It can use this to identify the essential activities.

Petri net has a complete functionality diagram initially. Later, Petri net has refined and applied the abstract transitions with lower-level models to retain the integrated model of the entire workflow system. The model has been designed after obtaining all possibilities of activities flow, either top-level or bottom-level, from distributed running logs. The event log of an abstract procedure always has a few details: case id, time duration of the event, and the Activity that occurred in the particular time frame. The attributes are different in the event log, reflecting the result based on the features.

Inductive Alpha Miner:

The OR and XOR logic is used to construct the reason for Petri net in the hospital system. The case id and Activity are considered the key points based on which the Petri net is drawn. Fig. 11. And Fig.12 show the Petri net with the attributes of case id and activities followed by the patients in the hospital. Similarly, we add the post-set activities and can build a successor set. The relation between the activities is constructed, and a new model is determined. The casual net shows the transitions corresponding to the successor activities, after which the model and the relation are built.

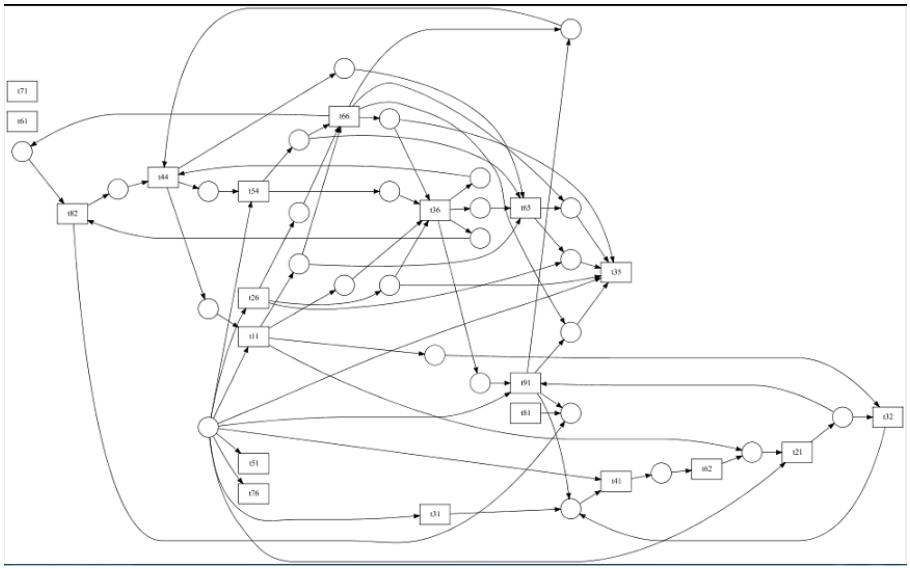


Fig. 12 Inductive Alpha miner for hospital data

The multiple sequences of activities are traced as a single workflow that can be represented in Fig. 13. The number of traces used in the dataset is connected with the activities used in the actual treatment, which will give the sequence graph in dotted format.

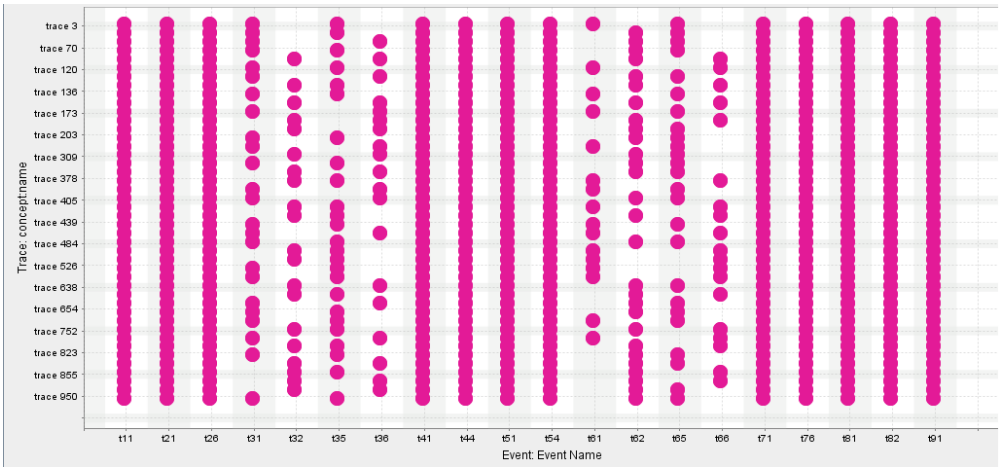


Fig. 13 Event log sequence based on traces

Node Connectivity in Alpha Miner:

The node's connectivity is displayed in Fig.14. Node *t11* is the starting node, and *t91* is the ending node. The complete flow and connectivity of the nodes are shown. The edges of the nodes depict the direction of the node's relation. The array marks provide node guidance and the path from the source node to the destination node. Petri Nets can use process mining to describe the diagram at a high level.

✉ **Shanmuga Sundari M**
mshanmugasundari@gmail.com

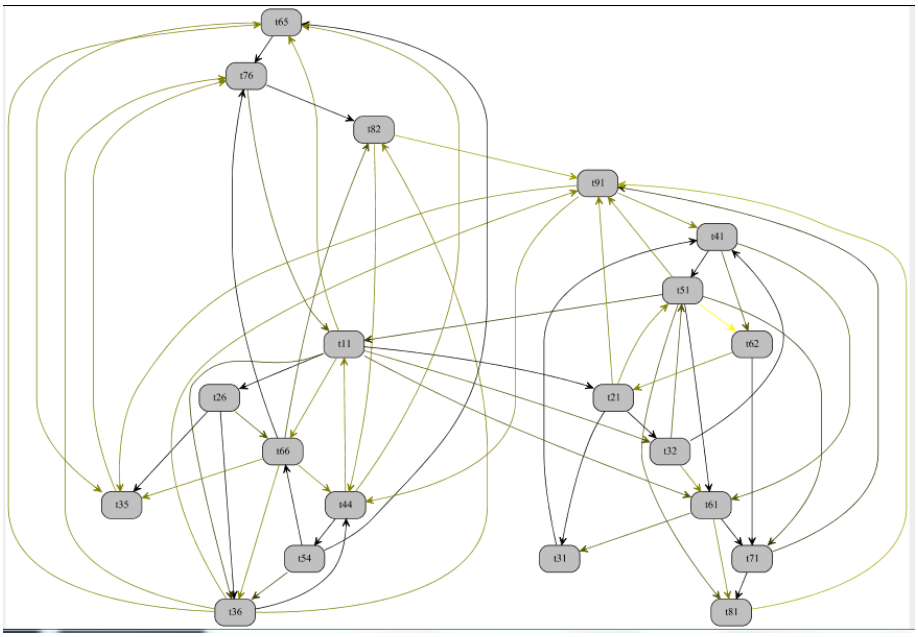


Fig. 14 Node connectivity in Petri net

Deviation will decrease the performance of the process. There are different categories of actions:

(1) high-impact actions: If any changes happen in this Activity, the whole structure will be affected. This node is crucial in timestamp as well.

(2) low-impact actions: If low sequence action can remove this from the flow, it is not affecting the hospital activity.

(3) no-impact actions: This can be removed at any time.

Fig.15 shows the deviation in the hospital database. It shows the deviation nodes. The nodes which are connected in red color edges are deviations. It will be a deviation if the nodes take more time to communicate with the other nodes because of traffic congestion.

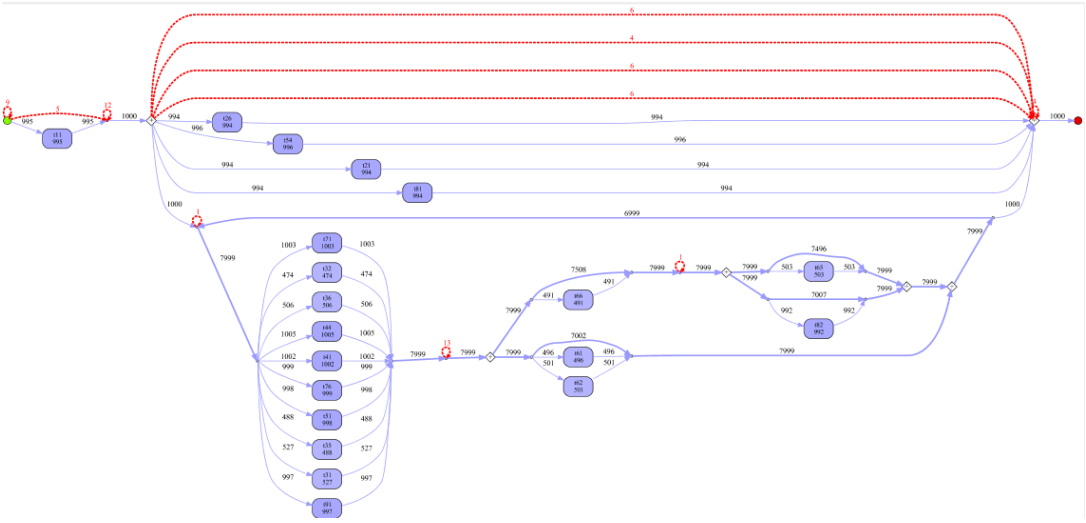


Fig. 15 Deviation for Hospital data

6- Results and Analysis

The result of this research is explained in the performance metric of events. This metric evaluation is presented in process mining, and the assessment is based on the relationship between activity sequences. For this test, we used 1000 patient's traces of activities, and process mining did each Activity in different departments in the hospital. The metrics are dependency and time stamp between the activities. By consideration of these metrics, the fuzzy miner produces the proper matrix for dependency and cluster formation for timestamp. It provides the better analysis than alpha algorithm. Notably, we restrict this test with two algorithms in process mining and find the deviation in the process. Based on this classification the metrics are more accurate using fuzzy miner.

Fuzzy miner interacts with the activities and finds high and low-performance groups. The graphical view can compare the identical core activity in the sequence. The dependency relationship of the traces in the mining process is depicted in Fig. 16. The values are displayed in the matrix based on the interdependence of the activities.

In the fuzzy miner, the blue-colored activities are incredibly bonding. These are considered the bottleneck activities that must take care of in the process. Management can find the bottleneck using this matrix and causal net diagram. Fuzzy miner gives this as a solution to monitor the process flow.

Matrix	t11	t21	t26	t31	t32	t35	t36	t41	t44	t51	t54	t61	t62	t65	t66	t71	t76	t81	t82	t91
t11	-0.68918...	0.356255...	0.186411...	-0.77785...	-0.75558...	-0.72240...	-0.75598...	-0.79517...	-0.79997...	-0.75028...	-0.77015...	-0.75784...	-0.76370...	-0.72331...	-0.76000...	-0.79754...	-0.82727...	-0.74651...	-0.77399...	-0.79721...
t21	-0.75880...	-0.88277...	-0.44908...	0.382342...	0.062083...	-0.69921...	-0.71202...	-0.87748...	-0.71230...	-0.79721...	-0.78695...	-0.86466...	-0.86485...	-0.76789...	-0.82983...	-0.87751...	-0.90638...	-0.79262...	-0.82647...	-0.88005...
t26	-0.68661...	-0.43020...	-0.75011...	-0.65463...	-0.66415...	-0.05208...	0.112765...	-0.65128...	-0.78596...	-0.69225...	-0.77001...	-0.74676...	-0.74443...	-0.72526...	-0.75714...	-0.78352...	-0.82718...	-0.74441...	-0.77385...	-0.79477...
t31	-0.74635...	-0.91595...	-0.48511...	-1.0	-0.91006...	-0.69054...	-0.77455...	0.013182...	-0.69317...	-0.78450...	-0.72369...	-0.91061...	-0.91071...	-0.75096...	-0.86487...	-0.91975...	-0.96341...	-0.78454...	-0.83171...	-0.91967...
t32	-0.72021...	-0.86347...	-0.52569...	-0.91002...	-0.84428...	-0.67554...	-0.78509...	-0.12782...	-0.70589...	-0.75541...	-0.71053...	-0.84179...	-0.84556...	-0.72958...	-0.80204...	-0.86364...	-0.88921...	-0.76100...	-0.80409...	-0.86347...
t35	-0.66063...	-0.57802...	-0.72433...	-0.70099...	-0.65483...	-0.69858...	-0.74687...	-0.67172...	-0.16121...	-0.65838...	-0.75270...	-0.71662...	-0.71410...	-0.69948...	-0.74545...	-0.75991...	-0.81143...	-0.71804...	-0.75008...	-0.78223...
t36	-0.72327...	-0.61182...	-0.76400...	-0.81163...	-0.74939...	-0.74683...	-0.84732...	-0.70277...	-0.06947...	-0.66564...	-0.81246...	-0.82208...	-0.81522...	-0.74451...	-0.84661...	-0.84332...	-0.92157...	-0.75313...	-0.80936...	-0.86188...
t41	-0.75338...	-0.88014...	-0.58867...	-0.91971...	-0.86364...	-0.68181...	-0.74654...	-0.88299...	-0.63129...	0.359784...	-0.61835...	-0.85848...	-0.85868...	-0.68019...	-0.77031...	-0.88031...	-0.80859...	-0.79511...	-0.76448...	-0.88285...
t44	-0.75628...	-0.67414...	-0.79749...	-0.78593...	-0.70905...	-0.78012...	-0.88522...	-0.57162...	-0.88313...	-0.59816...	0.457963...	-0.78967...	-0.76496...	-0.78073...	-0.86451...	-0.79373...	-0.93399...	-0.74193...	-0.83911...	-0.88291...
t51	-0.67937...	-0.79939...	-0.67098...	-0.78439...	-0.76097...	-0.67538...	-0.70120...	-0.79503...	-0.57968...	-0.74814...	-0.57962...	-0.04212...	0.011095...	-0.65524...	-0.68016...	-0.79045...	-0.65908...	-0.74630...	-0.69422...	-0.79707...
t54	-0.72069...	-0.73215...	-0.76580...	-0.76197...	-0.75452...	-0.75270...	-0.80380...	-0.60591...	-0.83934...	-0.61730...	-0.80527...	-0.64921...	-0.69903...	-0.08442...	-0.05167...	-0.72760...	-0.86077...	-0.69896...	-0.80501...	-0.83901...
t61	-0.72242...	-0.86458...	-0.73025...	-0.90229...	-0.84537...	-0.72245...	-0.82917...	-0.86472...	-0.81158...	-0.76315...	-0.73049...	-0.84292...	-0.84666...	-0.69084...	-0.74647...	0.090414...	-0.77863...	-0.76047...	-0.72729...	-0.86458...
t62	-0.72280...	-0.86164...	-0.74984...	-0.91067...	-0.84556...	-0.73461...	-0.83293...	-0.86491...	-0.77433...	-0.75806...	-0.69899...	-0.84666...	-0.84685...	-0.70302...	-0.77880...	0.160818...	-0.82758...	-0.76085...	-0.73641...	-0.86477...
t65	-0.66408...	-0.75011...	-0.72518...	-0.75083...	-0.73846...	-0.69710...	-0.74744...	-0.71778...	-0.78324...	-0.62973...	-0.75341...	-0.67618...	-0.70302...	-0.70033...	-0.74602...	-0.70529...	0.083338...	-0.67861...	-0.74850...	-0.77777...
t66	-0.72185...	-0.82659...	-0.76258...	-0.89799...	-0.82356...	-0.74246...	-0.84306...	-0.75457...	-0.86138...	-0.64989...	-0.81139...	-0.78921...	-0.78235...	-0.74602...	-0.84590...	-0.75462...	0.104150...	-0.69394...	-0.80536...	-0.86429...
t71	-0.75617...	-0.87743...	-0.77882...	-0.91971...	-0.86364...	-0.75986...	-0.84956...	-0.88028...	-0.79102...	-0.79969...	-0.69280...	-0.86475...	-0.86493...	-0.69525...	-0.77973...	-0.88031...	-0.72297...	0.002205...	-0.70236...	-0.86929...
t76	-0.79045...	-0.89721...	-0.82727...	-0.96341...	-0.91030...	-0.81439...	-0.92508...	-0.81784...	-0.93403...	-0.69129...	-0.87499...	-0.78223...	-0.83115...	-0.81199...	-0.91776...	-0.69564...	-0.99659...	-0.64605...	1.0	-0.92474...
t81	-0.68640...	-0.79463...	-0.70744...	-0.78431...	-0.76086...	-0.71776...	-0.75299...	-0.79723...	-0.67915...	-0.74990...	-0.62332...	-0.76307...	-0.75523...	-0.63168...	-0.60577...	-0.79961...	-0.53613...	-0.74806...	-0.49699...	0.344011...
t82	-0.72258...	-0.80373...	-0.77350...	-0.81050...	-0.78924...	-0.74973...	-0.81208...	-0.73680...	-0.83645...	-0.57210...	-0.80474...	-0.68338...	-0.65189...	-0.75050...	-0.81102...	-0.55030...	-0.87748...	-0.54288...	-0.80447...	0.379650...
t91	-0.64847...	-0.83158...	-0.69365...	-0.89407...	-0.81237...	-0.67628...	-0.81390...	-0.82782...	-0.82791...	-0.69716...	-0.76232...	-0.81348...	-0.81367...	-0.67689...	-0.81319...	-0.83175...	-0.90408...	-0.69371...	-0.75845...	-0.83158...

Fig. 16 Relation matrix for traces using fuzzy miner

There are more chances of bottlenecks in the entire flow. In Fig. 16, *t44* and *t54* show values of 0.457963 which means that the dependency of these traces is highly significant. Traces *t36* to *t44* value is -0.06947, which shows less significance between the processes. This matrix in fuzzy miner shows the dependency connectivity in between the traces.

According to this matrix, we can find that the pair of (*t81, t71) and (t76, t82)* are the most significant nodes, which means this node is intermediate level to connect other nodes. Node *(t51, t61)* is the many minor nodes that do not affect the flow if it takes more time to process.

The negative values and low-frequency values are not affecting the flow of the process. These are the activities that are not under the bottleneck resources. All positive resources and high-frequency values are considered bottleneck resources, and these are to be monitored carefully in the process flow. This dependency matrix gives the final frequency values for the bonding between each Activity. So, Fuzzy miner is proved to be one of the best algorithms to show the frequency values using the causal net diagram.

7- Conclusions and Future Work

Process mining techniques are a robust business model to generate patterns and analyze the process with the sequence of activity traces. Organizations or any business domain want to improve their production in the initial stage of development. So process mining will help in the beginning to evaluate the process deviations and find the bottleneck in the process sequence to improve their process in the early stage itself. If the process is organized without variation, the management will reach efficient data arrival. This research experimented with process mining in hospital datasets by applying alpha and fuzzy algorithms using the ProM tool.

This research gives the analysis of finding deviation in both the algorithms. Since our data set is unstructured, the fuzzy algorithm produces the dependency relation in an event log with the proper workflow diagram. The fuzzy miner shows the result with a relation matrix that is

the exact outcome of the connectivity of process mining. So, our dataset produces an efficient workflow using a fuzzy miner algorithm. We achieved the main objective of this research and found a better algorithm for activity deviation discovery and performance in process mining in the hospital system. In future, our model can be applied over large datasets like ontology, cardiology and it would incorporate process improvements.

References

[1] G. Li and R. M. De Carvalho, "Process Mining in Social Media: Applying Object-Centric Behavioral Constraint Models," IEEE Access, 2019, vol. 7, pp. 84360–84373, doi: 10.1109/ACCESS.2019.2925105.

[2] "Root Cause," 2021. [Online]. Available: https://appian.com/process-mining/root-cause-analysis.html#:~:text=The root cause analysis aims,impact factors such as bottlenecks.

[3] J. Xu and J. Liu, "A Profile Clustering Based Event Logs Repairing Approach for Process Mining," IEEE Access, 2019, vol. 7, pp. 17872–17881, doi: 10.1109/ACCESS.2019.2894905.

[4] G. Akhila, N. Madhubhavana, N. V. Ramareddy, M. Hurshitha, and N. Ravinder, "A survey on health prediction using human activity patterns through smart devices," Int. J. Eng. Technol., 2018, doi: 10.14419/ijet.v7i1.1.9472.

[5] W. Li, Y. Fan, W. Liu, M. Xin, H. Wang, and Q. Jin, "A Self-Adaptive Process Mining Algorithm Based on Information Entropy to Deal with Uncertain Data," IEEE Access, 2019, vol. 7, pp. 131681–131691, doi: 10.1109/ACCESS.2019.2939565.

[6] Q. Zeng, H. Duan, and C. Liu, "Top-Down Process Mining from Multi-Source Running Logs Based on Refinement of Petri Nets," IEEE Access, 2020, vol. 8, pp. 61355–61369, doi: 10.1109/ACCESS.2020.2984057.

[7] Z. Huang et al., "Safety Assessment of Emergency Training for Industrial Accident Scenarios Based on Analytic Hierarchy Process and Gray-Fuzzy Comprehensive Assessment," IEEE Access, 2020, vol. 8, pp. 144767–144777, doi: 10.1109/ACCESS.2020.3013671.

[8] W. Van der Aalst, Process mining: Data science in action. 2016.

[9] R. Tripathy et al., "Spectral Clustering Based Fuzzy C-Means Algorithm for Prediction of Membrane Cholesterol from ATP-Binding Cassette Transporters," in Intelligent and Cloud Computing, Springer, 2021, pp. 439–448.

[10] C. Subbalakshmi, G. Ramakrishna, and S. Krishna Mohan Rao, "Evaluation of data mining strategies using fuzzy clustering in dynamic environment," 2016, doi: 10.1007/978-81-322-2529-4_55.

[11] M. Anila and G. Pradeepini, "Study of prediction algorithms for selecting appropriate classifier in machine learning," J. Adv. Res. Dyn. Control Syst., 2017.

[12] G. Dorgo, K. Varga, and J. Abonyi, "Hierarchical frequent sequence mining algorithm for the analysis of alarm cascades in chemical processes," IEEE Access, 2018, vol. 6, pp. 50197–50216, doi: 10.1109/ACCESS.2018.2868415.

[13] J. Jin, W. Sun, F. Al-Turjman, M. B. Khan, and X. Yang, "Activity pattern mining for healthcare," IEEE Access, 2020, doi: 10.1109/ACCESS.2020.2981670.

[14] T. G. Erdogan and A. Tarhan, "Systematic Mapping of Process Mining Studies in Healthcare," IEEE Access, 2018, doi: 10.1109/ACCESS.2018.2831244.

[15] W. Li, H. Zhu, W. Liu, D. Chen, J. Jiang, and Q. Jin, "An anti-noise process mining algorithm based on minimum spanning tree clustering," IEEE Access, 2018, vol. 6, pp. 48756–48764, doi: 10.1109/ACCESS.2018.2865540.

[16] P. I. C. Kumari, P. Gayathri, N. Rajesh, S. Umar, G. C. Sekhar, and A. M. Abdul, "Designing of medical processor unit for intelligent network-based medical usage," Indones. J. Electr. Eng. Comput. Sci., 2016, doi: 10.11591/ijeecs.v4.i3.pp532-537.

[17] M. J. Hasan, A. Rai, Z. Ahmad, and J. M. Kim, "A Fault Diagnosis Framework for Centrifugal Pumps by Scalogram-Based Imaging and Deep Learning," IEEE Access, 2021, vol. 9, pp. 58052–58066, doi: 10.1109/ACCESS.2021.3072854.

[18] M. J. Hasan, D. Shon, K. Im, H. K. Choi, D. S. Yoo, and J. M. Kim, "Sleep state classification using power spectral density and residual neural network with multichannel EEG signals," Appl. Sci., 2020, vol. 10, no. 21, pp. 1–13, doi: 10.3390/app10217639.

[19] M. J. Hasan, J. Uddin, and S. N. Pinku, "A novel modified SFTA approach for feature extraction," 2016 3rd Int. Conf. Electr. Eng. Inf. Commun. Technol. iCEEiCT 2016, 2017, pp. 1–5, doi: 10.1109/CEEICT.2016.7873115.

[20] M. S. Sundari and R. K. Nayak, "Process mining in healthcare systems: A critical review and its future," Int. J. Emerg. Trends Eng. Res., 2020, vol. 8, no. 9, pp. 5197–5208, doi: 10.30534/ijeter/2020/50892020.

[21] V. S. Reddy and B. T. Rao, "A combined clustering and geometric data perturbation approach for enriching privacy preservation of healthcare data in hybrid clouds," Int. J. Intell. Eng. Syst., 2018, doi: 10.22266/ijies2018.0228.21.

[22] K. M. Hanga, Y. Kovalchuk, and M. M. Gaber, "A graph-based approach to interpreting recurrent neural networks in process mining," IEEE Access, 2020, vol. 8, pp. 172923–172938, doi: 10.1109/ACCESS.2020.3025999.

[23] A. E. Marquez-Chamorro, K. Revoredo, M. Resinas, A. Del-Rio-Ortega, F. M. Santoro, and A. Ruiz-Cortes, "Context-Aware Process Performance Indicator Prediction," IEEE Access, 2020, vol. 8, pp. 222050–222063, doi: 10.1109/ACCESS.2020.3044670.

[24] R. Tripathy, R. K. Nayak, P. Das, and D. Mishra, "Cellular cholesterol prediction of mammalian ATP-binding cassette (ABC) proteins based on fuzzy c-means with support vector machine algorithms," J. Intell. Fuzzy Syst., 2020, vol. 39, no. 2, doi: 10.3233/JIFS-179934.

[25] A. Massmann, P. Gentine, and J. Runge, "Causal inference for process understanding in Earth sciences," 2021, pp. 1–24, [Online]. Available: http://arxiv.org/abs/2105.00912.

[26] A. K. A. De Medeiros, A. J. M. M. Weijters, and W. M. P. Van Der Aalst, "Genetic process mining: An experimental evaluation," Data Min. Knowl. Discov., 2007, doi: 10.1007/s10618-006-0061-7.

[27] E. Kim et al., "Discovery of outpatient care process of a tertiary university hospital using process mining," Healthc.

Inform. Res., 2013, vol. 19, no. 1, pp. 42–49, doi: 10.4258/hir.2013.19.1.42.

[28] P. Nets, A. Networks, and R. C. Language, "Modeling of Resource Allocation Mechanisms in Distributed Computing Systems using Petri Nets and Stochastic Activity Networks (SAN): a Review and Reo-based Suggestion."

[29] R. K. Nayak. Sundari, M. Shanmuga, Efficient Tracing and Detection of Activity Deviation in Event Log Using ProM in Health Care Industry, 2021, 5th International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud (ISMAC 2021).

[30] S. C. Sekaran, V. Saravanan, R. RudraKalyanNayak, and S. S. Shankar, "Human Health and Velocity Aware Network Selection Scheme for WLAN/WiMAX Integrated Networks with QoS," Int. J. Innov. Technol. Explor. Eng. (IJITEE), ISSN, pp. 2278–3075.

[31] https://data.4tu.nl/articles/dataset/BPI_Challenge_2012/12689204/1.

[32] P. Selvaraj, V. K. Burugari, D. Sumathi, R. K. Nayak, and R. Tripathy, "Ontology based Recommendation System for Domain Specific Seekers," Proc. 3rd Int. Conf. I-SMAC IoT Soc. Mobile, Anal. Cloud, I-SMAC 2019, no. December, 2019, pp. 341–345, doi: 10.1109/I-SMAC47947.2019.9032634.

Journal of Information Systems and Telecommunication
Vol.11, No.2, April-June 2023, 123-134
http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

Cache Point Selection and Transmissions Reduction using LSTM Neural Network

Maliheh Bahekmat[1], MohammadHossein Yaghmaee Moghaddam[1*]

[1]. Department of Computer Engineering, Ferdowsi University, Mashhad, Iran

Received: 12 Sep 2021 / Revised: 04 Jan 2022 / Accepted: 19 Feb 2022

Abstract

Reliability of data transmission in wireless sensor networks (WSN) is very important in the case of high lost packet rate due to link problems or buffer congestion. In this regard, mechanisms such as middle cache points and congestion control can improve the performance of the reliability of transmission protocols when the packet is lost. On the other hand, the issue of energy consumption in this type of networks has become an important parameter in their reliability. In this paper, considering the energy constraints in the sensor nodes and the direct relationship between energy consumption and the number of transmissions made by the nodes, the system tries to reduce the number of transmissions needed to send a packet from source to destination as much as possible by optimal selection of the cache points and packet caching. In order to select the best cache points, the information extracted from the network behavior analysis by deep learning algorithm has been used. In the training phase, long-short term memory (LSTM) capabilities as an example of recurrent neural network (RNN) deep learning networks to learn network conditions. The results show that the proposed method works better in examining the evaluation criteria of transmission costs, end-to-end delays, cache use and throughput.

Keywords: Reliability; Selection of Cache Points; Middle Caching; Wireless Sensor Networks.

1- Introduction

Wireless sensor networks also are used for the collection data for monitoring of environmental information. Reliability of data transmission in wireless sensor networks is very important in the case of a high lost packet rate due to link problems of buffer congestion. "Internet of Things" (IoT) is a modern technology in which any creature (human, animal, or object) can send data through communication networks, whether the Internet or intranet. The data sending process between IoT devices is automatic according to the configuration at specific times (usually permanently and instantaneously), without demanding the "human-to-human" or "human-to-computer" inter-action. Wireless Sensor Networks (WSNs) can play a significant role in promoting to cast the cheap and straightforward network for connecting IoT devices [5, 6]. However, battery-powered sensor nodes impose nodes to have a limited energy resource. Charging or replacing the sensor battery may be unpleasant or impossible in a work setting based on WSNs. Therefore, when the node loses its energy, it may not be efficient for assessment and monitoring [3, 4]. Therefore, one of IoT-based wireless sensor networks' critical problems is severe energy limitation [7]. Since these networks' efficiency depends a lot on the network's life span and network coverage, it is necessary to consider energy-saving algorithms to design IoT-based wireless sensor networks with long life.

Nowadays, researchers have developed dynamic management methods to overcome the energy consumption issues in IoT-based WSN's [8,9]. However, the increasing reliability rate is an essential concern for the dynamic management energy resources methods [10,11]. The reliability rate is relevant to the successful data transfer in IoT-based WSNs. In General, two mechanisms are used to be confident in IoT-based WSNs in practice: pack-based reliability and event-based reliability. Pack-based reliability requires sending all received data by sensor nodes to well, which could waste limited energy resources of nodes while event-based reliabilities do not need sending all received data, and it depends on sending the data covering the event. The areas related to sensor nodes are mostly interfering with each other, and for this reason, they are similar to each other in higher levels of sensed data. In monitoring applications,

✉ **MohammadHossein Yaghmaee Moghaddam**
hyaghmae@um.ac.ir

the sensor network is in an environment that is supposed to monitor events such as fire or flood. This is possible with the application of small, cheap and smart sensor nodes. The sensors are equipped with low-power wireless interfaces used to communicate with each other. In environmental applications such as temperature and humidity monitoring, agricultural applications, urban life, 90 to 95% reliability is sufficient. But much more reliability is needed in military applications. Reliability means that all packets sent from the source must reach their destination and lost packets must be recovered by a secure schema. Reliability is calculated by the number of packets that reach the sink, not the reliability of individual packets. Reliability is directly related to energy efficiency. Sensor networks are a vital component of the Internet of Things (IoT) and are known as limited networks due to limited memory, computations, and energy capabilities.

One way to improve the reliability is to use local retransmission through interface caching. Data caching is an effective way for reducing the number of end-to-end re-transmissions, thereby reducing interference and overcoming variable channel conditions. Cache increases data access because it provides fast storage and retrieval of future information [1]. Caching techniques have a major impact on the transmission protocol proposed for WSN [2]. The development of the transport protocol should be independent of the other layers. A secure transport layer protocol is required in many wireless sensor network applications that provide different levels of reliability for different applications. Since congestion is one of the major causes of packet loss, congestion control mechanisms are a key component of the transport layer protocol. Congestion occurs when packets generated by sensor nodes exceed the network capacity. When congestion occurs in the network, the middle nodes destroy packets and this leads to the retransmission of packets and wasted energy in the network. Packet loss occurs not only due to congestion due to memory overload, but also for other reasons such as node mobility, node failure, collision, interference, and poor radio links. Hop-by-hop transmission is commonly used due to the short-range of sensor nodes in sensor networks. This increases the likelihood of packet loss and wasted energy to resend lost packets. To detect packet loss, explicit notification of lost packets has been proposed, which can be implemented as distributed (on sensor nodes) and centralized (on the sink). In the distributed model, sensor nodes use packet sequence numbers to identify lost packets. When a packet is lost, a middle node requests a re-transmission from its other neighbors. The sensor nodes detect the congestion from the buffer overflow. Therefore, to ensure the reliability of the network, the need for retrieval mechanisms such as retransmission and redundancy is felt.

Reliability can be divided into two levels [4]:

Packet or event confidence level

Hop-by-hop or end-to-end confidence level

data transmission in wireless sensor networks is in the case of a high lost packet rate due to link problems or buffer congestion. In this regard, mechanisms such as middle cache points and congestion control can improve the performance of the reliability of transmission protocols when the packet is lost. On the other hand, the issue of energy consumption in this type of networks has become an important parameter in their reliability

The contributions of this paper are as follows:

(1) Designing a reliable transport protocol using active cache management based on a deep learning algorithm and various cache management policies to improve cache performance.

(2) Providing a simulation and an analytical model to evaluate the performance of the cache-aware congestion control mechanism in the presence of lost packets in the WSN.

The rest of the paper is organized as follows: in section 2, the previously presented methods on sensor networks' reliability will be studied. Section 3 expresses the proposed method, section 4 evaluates the proposed method and studies its function, and finally, section 5 will give the overall conclusion.

2- Related Works

The data sending process between IoT devices is automatic according to the configuration at specific times (usually permanently and instantaneously), without demanding the "human-to-human" or "human-to-computer" inter-action. Wireless Sensor Networks (WSNs) can play a significant role in promoting to cast the cheap and straightforward network for connecting IoT devices

In [12], a new method has been presented for clustering wireless sensor nodes to reduce energy consumption named EAC. EAC is a clustering algorithm based on energy and distance; that means, sensor nodes are chosen as cluster head based on remaining energy. Meanwhile, non-cluster head nodes chose their cluster head based on distance from neighbor cluster heads. EAC algorithm increases the life span of network via balancing energy load among network nodes.

In [13], has presented a hierarchical clustering method for reducing energy consumption in wireless sensor network. This algorithm divides network to circles with different power levels in sinks and each circle has different nodes. Simulations and results obtained from using three scales of network life span, number of clusters and consuming energy of cluster heads showed that the efficiency of this method is better that LEACH in terms of energy consumption of cluster head, number of clusters and life span of network. In fact, this method reduces the number of dead node and energy consumption and increases the

life span of network. This algorithm includes three phases of launching, cluster launching and routing among clusters. In [14], another hierarchical algorithm has been presented for reducing energy consumption in wireless sensor network. The presented method in this papers uses a mechanism to prioritize clusters and packs of data. This protocol provides a route without congestion for optimal energy consumption for necessary data packs via prioritizing. Therefore, the best route always remains for transferred vital and necessary data. Therefore, the proposed algorithm in this paper minimizes the delay and consuming energy and maximizes the life span of network and operational power.

BLAC algorithm [15] used combination of battery level and other criteria like node density and rank for choosing cluster head. For balancing in energy consumption, the cluster head is taken frequently by each node. In BLAC, the cluster heads gather data from his cluster sensor nodes and sends them via GPRS links.

In 2012, some researchers [16] studied the manner of forming clustering and especially a primary schema entitled fuzzy logic cluster formation protocol where fuzzy logic is used in clustering process. Several changes on several parameters related to fuzzy logic and clustering reduces the energy consumption and therefore increases the life span of network.

In [17], the optimization algorithm of single cluster network and multi-cluster networks are proposed where node energy harvesting are allocated to strengthening nodes chosen as cluster head and therefore, it leads to more survival of network. As it was said, the energy production resource in sensors is battery and this resource has less capacity. None of the above works could minimize the energy consumption and therefore the life span of wireless sensor networks will be less such that some of sensors will be eliminated after some time.

In [18], the neural networks were used for dynamic management of power (maximizing life span of sensor nodes after placement and for scheduling the cycle of the duties of sensor nodes (determining which node should be slept and which one should be alive). In this method, next event time is a non-fixed series which are predicted by wavelet neural network precisely. The mentioned neural network is a three-layer network which uses Morlt wavelet transform in hidden layer. The nodes which are in deeper sleeping modes consume less energy while it causes more delay and high energy consumption for waking.

In [19], a self-organizing neural network was used for reducing and classifying similar patterns. They used SOM in a hierarchical network architecture (based on the cluster) where nodes are an organization in several clusters and cluster head or sinks of data combinations. This self-organizing neural network reduces the transferring of data and classifies similar patterns.

Wireless sensor networks are widely used to perform the automations in many applications. The WSN is used in both attended and unattended environment such as Internet of Things, smart phones, health monitoring, surveillance, volcano monitoring, boarder surveillance and more The IoT based WSN are emerging rapidly because of its versatility and economic nature [20,22].

In many applications of wireless sensor networks, providing reliability and healthy delivery of packet to the destination is of great importance. Reliability is one of the tasks of the transport layer in these networks, which gives the network the ability to deliver data sent to the receiver securely [23.24].

3- The Proposed Method

Because WSNs are cost-effective and modular, they can be used to secure smart cities by providing remote monitoring and sensing for a variety of critical scenarios. In [1], a new framework for remote sensing and monitoring in smart cities using WSNs is proposed. In their proposal, they suggested using Unmanned Aerial Vehicles to act as a data mule to offload the sensor nodes and transfer monitoring data securely to the remote control center for further analysis and decision-making. Additionally, the paper provides insights into the implementation challenges of the proposed framework.

Machine learning technique is used in the proposed method of this paper to train the network based on various factors, such as buffer capacity, number of hops, energy node, speed of node, popularity and number of successful deliveries. The machine learning algorithm is trained based on previous network routing data to generate an equation, which examines the probability used, whether the communication node is able to deliver the message to the intended destination. To better understand the formula of this proposed method in Table 1 the notation of the formula is explained. This value is then used to decide on the next hop for the buffered message. In the training phase, long-term short-term memory capabilities are used to learn network conditions. The schema of this proposed method is shown in Figure 1.

Table 1: The notation of the formula

Notation	Description
P_f	Probability of failure due to failure
P_c	Probability of failure due to congestion
P_n	Probability of failure due to noise
P_{sd}	Probability of success to the destination
P_e	Bit loss probability
P_p	The probability of packet loss

PEP_i^f Probability of progress of flow packets

$PER_{j,\ j+1}$ Probability of losing the packet on the link between nodes j and j + 1

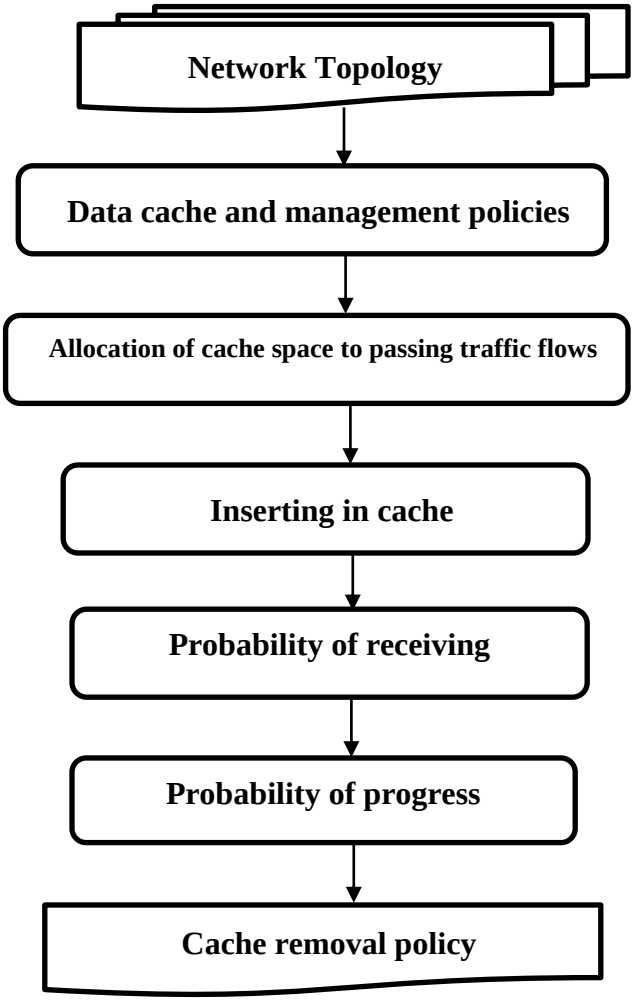


Fig. 1 Schema of the proposed method

- **Long Short Term Memory (LSTM) Cell**

LSTM is an artificial recurrent neural network (RNN) architecture [1] used in the field of deep learning. Unlike standard feed forward neural networks, LSTM has feedback connections. It can process not only single data points (such as images), but also entire sequences of data (such as speech or video). For example, LSTM is applicable to tasks such as un-segmented, connected handwriting recognition speech recognition and anomaly detection in network traffic or IDSs (intrusion detection systems). A common LSTM unit is composed of a cell, an input gate, an output gate and a forget gate. The cell remembers values over arbitrary time intervals and the three gates regulate the flow of information into and out of the cell. LSTM networks are well-suited to classifying, processing and making predictions based on time series data, since there can be lags of unknown duration between important events in a time series. LSTMs were developed to deal with the vanishing gradient problem that can be encountered when training traditional RNNs. Relative insensitivity to gap length is an advantage of LSTM over RNNs, hidden Markov models and other sequence learning methods in numerous applications.

Table 2: LSTM with a forget gate [2]

$$f_t = \sigma_g(W_f x_t + U_f h_{t-1} + b_f)$$

$$i_t = a_g(W_f x_t + U_i h_{t-1} + b_i)$$

$$o_t = a_g(W_o x_t + U_o h_{t-1} + b_o)$$

$$\tilde{c}_t = \sigma_c(W_c x_t + U_c h_{t-1} + b_c)$$

$$c_t = f_t \ o \ c_{t-1} + i_t \ o \ \tilde{c}_t$$

$$h_t = O_t \ o \ \sigma_h(c_t)$$

Variables

$x_t \in R^d$: input vector to the LSTM unit

$f_t \in (0,1)^h$: forget gates activation vector

$i_t \in (0,1)^h$: input/output gates activation vector

$o_t \in (0,1)^h$: output gates activation vector

$h_t \in (-1,1)^h$: hidden state vector also know as output of the LSTM unit

$\tilde{c}_t(-1,1)^h$: cell input activation vector

$c_t \in R^d$: cell state vector

$W \in R^{h*d}$, $U \in R^{h*h}$ and $b \in R^h$: weitht matrices and bias vector parameters

3-1- Input Parameters of the Training Phase

The proposed method uses machine learning techniques to select the next cache points. When a connection is established between two nodes and the buffer of one node contains a message to be transmitted, a decision must be made as to which node the message should be transmitted (as the next point selection). Normally, the message should only be sent from the sender to the adjacent recipient node, provided that the middle node has a sufficient probability of direct or indirect transmission to the destination node. Repeated message transmission can lead to packet loss and higher buffer overflow and higher power consumption. On the other hand, reducing frequent transmissions increases the number of unsuccessful delivery of messages. The probability of successful delivery depends on various factors that indicate the recent background and the ability of the nodes to deliver the message successfully. The probability of delivery in the next selected cache is

calculated using a trained ML model, which includes the following features: delivery probability, buffer capacity, successful deliveries, success rate, source and destination node speed, distance from message source, distance to message destination, number of hops to current node, and message life time. The message life time parameter provides the duration from the creation of the message to the present. The message is transmitted from the transmitter node, if $P_m > P_r$ this point is considered as the cache point of this flow. In this expression, P_m is the probability of final delivery at the node and has been calculated using ML techniques. P_r is the current reliability rate at the transmitter node.

In the initial phase of the learning phase of this dissertation, the aim is to collect data related to network behavior analysis and then more effective items on network traffic management are selected by creating a database in the preprocessing phase. Then, in the second phase, the prediction accuracy increased by using the deep learning model based on long-short-term memory feedback neural networks (LSTM), we present a deep learning model so that we can improve the learning depth by deepening the time windows (short-term daily and long-term annual). In the third phase, the output of the deep model is given to the extreme learning machine (ELM), which can calculate the estimated delivery time of each packet instantaneously according to other input data. In the proposed method, we intend to use a stack of LSTMs due to the high sensitivity of time series data. The output of these networks is the probability of success to the destination.

- **Calculation of the Probability of Final Delivery Based on Machine Learning**

LSTM network: In the proposed method, the LSTM recursive neural network model with multiple hidden layers has been studied. Where (x1,…., X16) are the input parameters. Each training data sample provides input values (x1,…, x16) for selecting the next cache point, and the obtained output determines the probability of successful delivery of sent message to the destination. Prior to training, a sample network has been provided, that is, it has been quantified at random values and then the deep network is learned by repetition in the training set, thus it provides the desired predictions about whether the delivery will be successful or not. To calculate different p values, the machine learning model must first be prepared, trained, or constructed based on the training scenario data. Relevant data is called training data, and a specific data input is a training example. Then, during the next point selection process, the trained LSTM network is used to calculate P(Y) based on the input parameters (x16, ..., x1, x2), which are obtained in real time. The probability of successful sending by each node is denoted by P_{su} and is calculated by the following equation:

$$P_{su} = 1 - P_{loss} \quad (1)$$

$$P_{loss} = P_f * P_c * P_n * P_{sd} \quad (2)$$

Where, P_f is the probability of failure due to failure, P_c is the probability of failure due to congestion, $P_n$represents the probability of failure due to noise, and P_{sd} denotes the probability of success to the destination.

- **Calculation of the Probability of Packet Loss Due to Noise**

Assume that the wireless network has a bit loss probability equal to P_e. If the average length of packets is S bits, the probability of packet loss P_p can be calculated as follows:

$$p_p = 1 - (1 - p_e)^s \quad (3)$$

The probability of packet loss due to channel noise P_n is equal to P_p.

- **Calculation of the Probability of Packet Loss Due to Failure**

Another cause of packet loss in wireless sensor network nodes is hardware failures of the node. Assume that P_f represents the probability of loss due to hardware failure of the node. The failure rate of a sensor node depends on its external and internal factors. External factors affecting a node can be considered through the MIL-HDBK documentation. But internal factors need to be included in the final formula somehow. In the first step, the components of the system must be thoroughly examined in order to provide a series or parallel model or other modes that can be considered for it. The series mode for a system is the state in which the failure of each component of the system causes the failure of the entire system, and the parallel mode of a system is the state in which if all the components of a system fail, the entire system will fail. It is clear that the essential components of a node (microcontroller, memory, battery, communication device, sensor component, and ADC) form the system series; because the failure of each component causes the failure of the entire system. But the optional components, including the actuators and the number of sensors, form a parallel system. The probability of the proper function of a part or the reliability in the general case is obtained from the following equation.

$$R(t) = P_s = P(t_f \geq t \quad (4)$$

Given the above relation, the function f(t) must be specified. Given that the lifespan of most electronic components follows an exponential relationship, with the reliability of sensor node (R(t)), the probability of packet loss can be obtained as follows:

$$P_f = 1 - R(t) \quad (5)$$

- **Calculation of the Probability of Packet Loss Due to Congestion**

Each node has a buffer or queue. Its length and size can be a simple and good sign for congestion. The size of the

buffer can be considered as a threshold. The proposed method uses a fixed threshold and if the buffer size exceeds the threshold, the congestion is detected. In some methods, the buffer size is periodically checked at the beginning of each period and the congestion is signaled instantaneously. The remaining length of the overall buffer size, or the difference between the remaining space and the traffic rate, can be used as a possible indicator of congestion. In the proposed method, by monitoring the average buffer queue length, a series of times is obtained that can use time delay neural network (TDNN) to predict its future values, which is actually the future status of the buffer of the cache point. For this purpose, we first design a suitable neural network and train it through the above time series. It then implements the trained neural network into the buffer point buffer to predict the future value of the average length of the buffer queue at a few time periods, called the forecast horizon, based on current and previous values of the average buffer queue length. Then, based on the result of this prediction, we introduce a mechanism called ML-RED, which operates on the basis of the RED algorithm, and the transmitters are notified before the congestion starts and reduce their transmission rate. One of the advantages of this method is that the determination of the cache points by the nodes is done locally and completely dynamically. This is done by comparing the transmission rate of each node with the required reliability interval for each stream. It should be noted that at the beginning of the network, before entering the learning stage, the cache points are randomly selected. In order to learn this, the cache points are selected or removed independently and locally by the middle nodes. Disclaimer of previous cache points only includes not inserting new entries and they are responsible for resending until deleted from the cache as a point cache. In the proposed method, congestion control is done globally and end-to-end and the selection of cache points is done locally by each node.

3-2- Data Cache and Management Policies

The PRM-DDCLAM (Deep PRM) protocol is a DTSN[1] protocol with sender side changes. This protocol uses both ACK and NACK messages that the recipient asks the sender to send through explicit acknowledgment request. The EAR signal is mounted on the data packet. After sending the EAR, the source launches the EAR timer. If the EAR timer expires before receiving the ACK/NACK, the source sends the EAR packet again. After adopting the EAR on the node of the receiver, a NACK, containing a bit map of the lost packets, is generated and sent to the sender. During the transport of such NACKs, cache points learn the lost packets and check if there are packets in the cache. If this is true, the cache points resend packets to the receiver and change the NACK bitmap before sending it to the sender. These implicit ACK and NACK notifications are a cache removal policy which are for removing all packets that have already been ACKed and also creating space for new incoming packets. Similarly, ML-PRM adapts a NACK repair mechanism so that cache points can output the NACK signal to speed up the repair process. In addition to receivers that can identify lost packets, cache points can also find lost packets and signal the node of the previous steps through the Repair Non- Acknowledgment Control Packet (RNACK), which includes a counter of lost packets. After receiving the RNACK, the node of the previous step, if it finds a copy in the cache, resends the lost packet to the destination. Otherwise, RNACK will be broadcast to the source. Assuming the route is always constant, RNACK transmission is not based on a timer, but occurs as soon as an out-of-row packet (outside the packet sequence) is detected. This capability further reduces the risk of packet loss by the accelerated recovery, thus it promotes instant transport.

3-3- Allocation of Cache Space to Passing Traffic Flows

In [16] and [17], the cache segmentation method has been used to manage the cache capacity. In this method, a cache point divides its cache capacity among the flows passing through the node in its route according to the cache segmentation policy in the network. To explain this method, suppose that ci represents the capacity of the node i and F_i^n is the total number of the flows passing through node i in its route. Using the cache segmentation policy, the weight w_i^f is assigned to each of these streams, which actually determines the part of the node cache i, which is assigned to flow f. By determining the weight of all the flows passing through node i, the true share of each flow from ci is determined by Equation (6):

$$p_i^f = \frac{w_i^f}{\sum_{j=1}^{F_i^n} w_i^j} \quad (6)$$

In this method, a packet of flow f can only be placed in the part of node i, which belongs to flow f, whose size is equal to $p_i^f \times c_i$

In the proposed method, after determining the optimal cache points of traffic flows passing through the network, considering the limited memory space of each middle node, the cache space of each node should be divided according to a specific policy between traffic flows passing through each node. The simplest cache segmentation policy is a uniform distribution policy in which the entire cache space is equally distributed between each traffic stream passing through it. This is certainly not a good policy. Our proposed solution in this section is to allocate the cache space of each node to the traffic flows passing through it based on various criteria such as: the requested reliability

[1] Distributed Transport for Sensor Networks

of the traffic flows and the distance of the node to the destination and the priority of the flow. In this way, traffic flows with high reliability and high distance of the nodes from the destination and have a higher flow priority, take up more space of the cache memory. The goal is to provide an exploratory algorithm based on effective parameters. To calculate the weighting coefficient of cache space allocation w_2^f, fuzzy systems with three inputs, the required reliability of traffic flows and the distance between the node and the destination and flow priority are used. As a result, the actual share of each flow from ci is determined by the following equation:

$$p_i^f = \frac{w_2^f w_i^f}{\sum_{j=1}^{F_i^n} w_i^j} \quad (7)$$

3-4- Inserting in Cache

Considering the limited resources in wireless sensor networks, it is crucial to provide designs that try to make optimal use of these resources. Transport layer protocols use cache in the middle nodes to counteract the inefficiency of the end-to-end retransmission method, enabling middle nodes to store packets received from different flows in their cache and resend them when needed. The purpose of storing packets in the middle nodes and locally retransmitting them is to minimize end-to-end retransmission, thereby reducing the number of transfers needed to send a packet from source to destination, energy consumption, and packet delay. To meet this goal, given the cache memory limitations, middle nodes must adopt appropriate policies to manage their cache space. Certainly one of the most important of these policies is the policy of selecting packets to be stored in the middle nodes.

A middle node decides to save a copy of the packet in its cache before sending it according to the packet selection policy. In an optimal packet selection policy, it should be tried that the packets with the experience of more difficult conditions in their downstream nodes as a result, lower probability of being received, have a higher chance of being stored in the middle node cache, so that they do not have to spend a lot of money to recover if they are lost along the route. On the other hand, a middle node in its packet selection policy should store packets with a lower probability of progress in the upstream nodes with a higher probability in its cache to prevent end-to-end retransmission. Undoubtedly, having a high cache capacity such that all incoming packets can be stored is the best option for middle nodes. However, given the limited resources (including memory) in sensor nodes, a middle node can only select a limited number of packets to store in its cache. On the other hand, providing a suitable policy for selecting packets depends on having a proper approach in determining the weight or in other words, the priority of each received packet in the middle nodes so that a suitable policy for selecting packets can be adopted through the weight of a packet. Therefore, in the following, we will examine the effective parameters in determining the weight of packets.

3-5- Effective Parameters

The purpose of examining the effective parameters in determining the weight of packets is to provide appropriate policies for selecting packets to be placed in the middle node cache. In such a way that these policies can serve the purpose of using cache in the middle nodes. To determine the effective parameters in packet selection, according to Figure (1), a middle node is considered as a destination for packets received from downstream nodes and, on the other hand, a source for the packets sent to the upstream nodes in the following route.

3-6- Probability of Receiving

In this section, we consider a middle node as the destination for packets received from downstream nodes. Therefore, we will examine the parameter affecting packet transport in the downstream nodes of a middle node. It should be noted that due to the nature of the message transmission in hop-by-hop form in a wireless sensor network, if a node sends a message to a destination in n farther hop, the probability of receiving the message by the destination is determined using Equation (8).

$$PoR_{\mathrm{n}} = \prod_{\mathrm{k}=0}^{\mathrm{n}-1}(1 - \mathrm{PER}_{\mathrm{k},\mathrm{k}+1}) \quad (8)$$

In an end-to-end retransmission scheme, if a message is lost along the route between sender and receiver, the source resends the message to the destination. In this case, the probability of receiving a message at least one time in the destination after sending for t times by the source is determined based on Equation (9):

$$\mathrm{r} = \sum_{\mathrm{k}=0}^{\mathrm{t}-1}(1 - \mathrm{PoR}_{\mathrm{n}})^{\mathrm{k}}.\mathrm{PoR}_{\mathrm{n}} \quad (9)$$

In Equation (2), $(1- P_oR_n)^k$ is the probability of not receiving a packet after k times of sending, and PoR_n is the probability of successfully receiving a packet after (k + 1) attempts. This equation shows that, after t times of sending a message by the source, it is expected that the message will be received with minimum reliability r at a destination with a distance of n hops. Lem (1) shows the geometric series introduced in Equation (10) as an equation of degree t.

$$\begin{aligned} \mathrm{r} &= \sum_{k=0}^{t-1}(1 - PoR_n)^k.PoR_n \\ &= PoR_n.\sum_{k=0}^{t-1}(1 - PoR_n)^k \\ &= \frac{PoR_n.(1-(1-PoR_n)^t)}{1-(1-PoR_n)} \\ &= 1-(1-PoR_n)^t. \end{aligned} \quad (10)$$

Equation (4), using Equation (3) obtained from Lemma (1), calculates the maximum number of transports required to achieve end-to-end reliability r:

$$r = 1 - (1 - PoR_n)^t$$

$$1 - r = (1 - PoR_n)^t$$

$$t = \log_{(1-PoR_n)}(1 - r) \quad (11)$$

Equation (11) shows that in an end-to-end message transport design, a maximum of t (including retransmission) end-to-end transports by the source is required to provide a certain reliability (r) in delivering a message. As mentioned earlier, the use of cache in middle nodes tries to minimize end-to-end retransmission. Therefore, packet selection policies for caching in the middle nodes should select packets to be stored in the cache among the received packets, which will cause the greatest reduction in the number of end-to-end retransmissions (t).

3-7- Probability of Progress

The policy of selecting packets in a mid-node in addition to considering the conditions experienced by the received packets in downstream nodes, should also consider the status of packets in the route. In fact, a middle node, as the source of packets that it sends to upstream nodes, should try to store packets with low probability of placement in the cache of the upstream nodes or reaching their destination before being sent. In this section, we use the progress probability parameter to examine the condition of the received packets in the upstream nodes of a middle node. That is, the probability of storing packets of a flow in the cache of the upstream nodes of a middle node or reaching the destination. Equation (12) calculates the probability of progress of flow packets such as f in the upstream nodes of the middle node simplifies the description of Equation (12).

$$PEP_i^f = 1 - \sum_{j=i}^{l_f-1} \left[\prod_{k=i}^{j-1} \left(1 - PER_{k,k+1} \right) . (1 - CP_k^f) \right] . PER_{j.j+1} \quad (12)$$

In this equation $\left[\prod_{k=i}^{j-1} \left(1 - PER_{k,k+1} \right) . (1 - CP_k^f) \right]$ represents the probability of reaching a packet of flow f is from node i to node j, in the case that none of the middle nodes between i and j are stored in the cache, and $PER_{j,\,j+1}$ is the probability of losing the packet on the link between nodes j and j + 1. Node j can be any of the nodes upstream of node i (except destination).

Therefore;

$\left\{ \sum_{j=i}^{l_f-1} \left[\prod_{k=i}^{j-1} \left(1 - PER_{k,k+1} \right) . (1 - CP_k^f) \right] . PER_{j.j+1} \right\}$ is the probability of non-progress of the flow packets f among all upstream nodes of node i. A packet selection policy used in middle nodes should store packets that have a low probability of progress in the middle nodes. Because these packets are likely to need to be resent and, the chances of upstream nodes being able to retrieve and resend these packets from their cache is low. On the other hand, storing packets in the middle node cache that have a high probability of progress will waste the cache capacity. Because these packets are most likely to be received by the destination or stored in the upstream middle nodes. In the first case, there is no need to resend the cached packets, and in the second case, if there is a need to resend, the middle nodes closer to the destination (due to receiving the lost message list sooner or faster timer expiration) will send the lost packets faster.

- **Local Variables and Packet Headers**

Implementing the proposed cache management system requires several local variables in the middle nodes, one field in the data packet header, and one field in the acknowledgment packet header.

To calculate the parameter of the probability of receiving each packet, we use the PoR field in the packet header. To explain the function of this method, let $PER_{i,\,i+1}$ be the probability of lost packets on the link between nodes i and i + 1. Considering Equation (2), node i multiplies the value in the PoR field by (1- $PER_{i,i+1}$) before sending the data packet to node i + 1 so that node i+1 is likely to receive this packet. Therefore, each middle node, given the PoR value of a packet, finds out the probability of receiving that packet again. The initial value of the PoR field is 1. To calculate the probability of progress of the packets of a flow, each node must store three local variables for each flow that passes through that node in its route. For example these variables for flow f are:

1. pep_flow (f): This variable specifies the probability of effective progress of the packets of flow f in the upstream nodes.
2. received_packet (f): This variable specifies the number of packets received from flow f.
3. cached_packet (f): This variable specifies the number of packets of flow f that have been successfully cached.

Calculating the effective progress probability parameter in addition to the variables mentioned will require a PEP field in the acknowledgement packet header. Given that the effective progress probability parameter for packets of a flow is calculated in the upstream nodes of a middle node. Therefore, we have mentioned the recursive form of Equation (12) in Equation (13) so that the acknowledgement packets that move in the opposite direction of the data packets from the destination to the source can calculate the probability of progress using the local variables of the middle nodes in the PEP field of their header. It should be noted that, like Equation (4-5), to calculate the effective progress probability, we use its opposite, that is, the probability of effective progress.

$$PEP_i^f = 1 - \left[PER_{i,i+1} + \left(1 - PER_{i,i+1} \right) NPEP_{i+1}^f \right] \quad (13)$$

$$NPEP_i^f = \left(1 - PEP_i^f \right) \left(1 - CP_i^f \right) \quad (14)$$

$$NPEP_h^f = 0 \quad (15)$$

PEP_i^f represents the probability of progress of the packets of flow f among the upstream nodes of node i and $NPEP_i^f$ is the probability that the packets of flow f will not progress among the upstream nodes of node i and will not be stored on the cache of node i. Specifically, the probability of non-progress of the packets of a flow in the upstream nodes of the destination (h) and non-caching on the destination node is zero. In each node, as the acknowledgment message is received and after calculating PEP_i^f, this value is placed in the local variable pep_flow (f) corresponding to the flow f, then $NPEP_i^f$ is calculated and an acknowledgment message is placed in the PEP field and sent to the next node in the direction of source. Therefore, after receiving an acknowledgment message from a flow such as f in a middle node such as i, the value $NPEP_{i+1}^f$ will be in its PEP field, and this value will be changed to $NPEP_i^f$ before sending for node i-1. Each middle node to calculate PEP_i^f requires the calculation of CP_i^f, which is obtained based on Equation (16):

$$CP_i^f = cached(f)/recived(f) \quad (16)$$

In this method, each middle node updates the probability of the progress of the packets by receiving an acknowledgment message from a flow such as f. The packet selection policy in the proposed cache management system, in addition to the parameters of probability of receipt and effective progress, also requires the parameter distance from the source. To calculate this parameter, we use the ttl field in the data packet header.

3-8- Cache Removal Policy

Any packet stored in the cache of a middle node will be deleted from the cache only by replacing or receiving an acknowledgment message. But if the cache does not have enough space to store the new packet and the input packet weighs more than the packet in the cache can be selected from the packets stored in the cache of the packet whose remaining lifespan is ending to be replaced.

4- Experimental Results

In this section, we will evaluate the proposed cache management system. The network topology intended for evaluation is a 10×10 grid topology, which is shown in the figure 2. In this topology, each of the nodes at the end of the communication links sends a flow of data to the corresponding nodes on the opposite side. Hence each of these nodes sends a data flow and receives a data flow. This approach allows four data flows to pass through each middle node. The direction of these flows and the number of flows passing through the middle nodes have been specified in the figure 2.

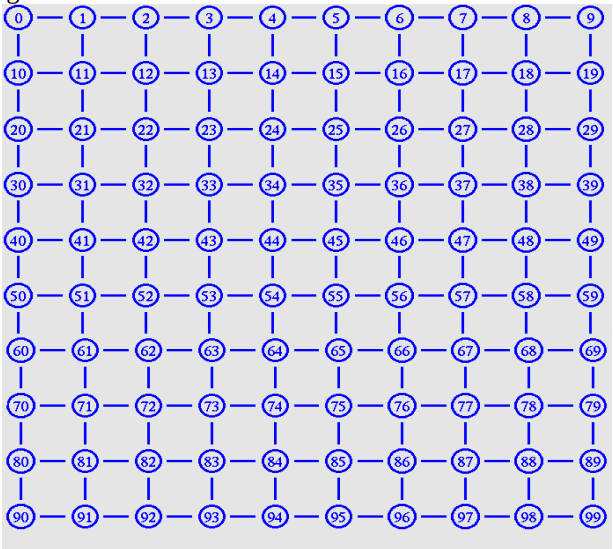


Fig. 2 10 × 10 grid topology for simulation

The simulation was used by ns-2 simulator to evaluate the proposed cache management system. This simulator, which is based on a discrete and object-oriented event simulation method, was designed and implemented by Berkeley University and, it is one of the best tools available to researchers to simulate wired and wireless networks.

In the scenario considered for the simulation, the source of each flow sends 250 packets to the destination, which due to the presence of 40 flows, a total of 10000 packets have been sent in each time of the simulation run in this network. The start time of each flow was randomly ranged from 1 to 1 second and the data transmission rate is fixed. The results of the evaluations have been presented for the proposed method and the comparative method presented in [24] as the Based Method.Two different scenarios have been used to compare the results of the two methods. In the first scenario, the error rate between nodes is considered between 0.2 and 0.6 with intervals of 0.05, while the cache size of the middle nodes has been considered to be 25. But in the second scenario, the error rate between the links is fixed and between 0.2 and 0.3, but the cache size of the middle nodes is between 5 to 40 with intervals of 5 units. In both scenarios, the goal was to achieve reliability between 0.8 and 0.9. The results of these two scenarios have been presented in the form of evaluation.

4-1- Transmission Cost

Given the energy constraints on sensor nodes and the direct relationship between energy consumption and the number of transmissions made by the nodes, a cache

management system should try to minimize the number of transmissions needed to send a packet from source to destination. We define the transmission cost as the average number of transmissions required to send a packet from source to destination. In this subsection, we will examine the transmission cost in the proposed methods. Figure 3 and Figure 4 show the transmission cost in comparative methods for different error rates and cache sizes.

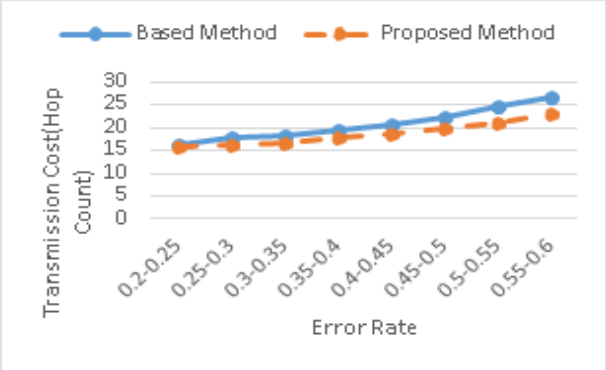


Fig. 3 Investigation of transmission costs in the proposed method and the based method with different error rates

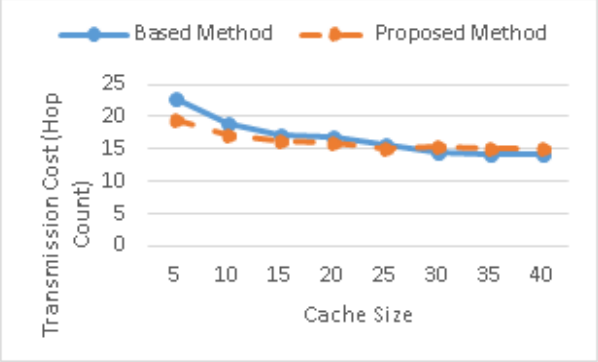


Fig. 4 Investigation of transmission costs in the proposed method and the based method with different cache sizes

4-2- End-to-end Delay

The delay calculated in this section is based on the definition of end-to-end delay. End-to-end delay is the time between the first sending of a packet and the successful receipt of the packet sent to the destination. Figure 5 and Figure 6 illustrates the end-to-end delay.

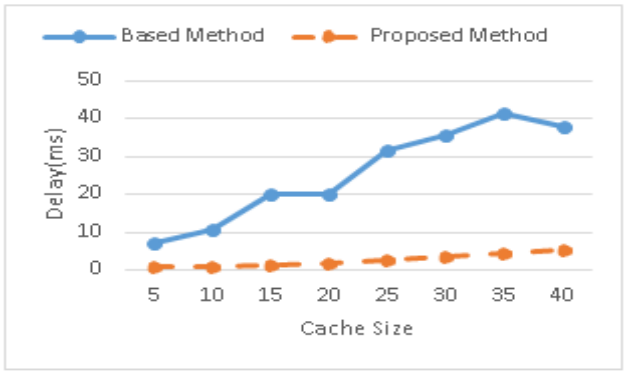


Fig. 5 Delay of sent packets at different error rates

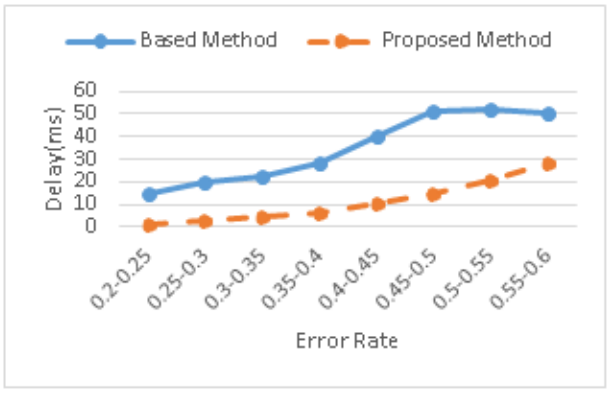


Fig. 6 Delay of sent packets with different sizes

4-3- Cache Use

Figure 7 and Figure 8 show the cache use in the middle nodes for different scenarios in the proposed methods.

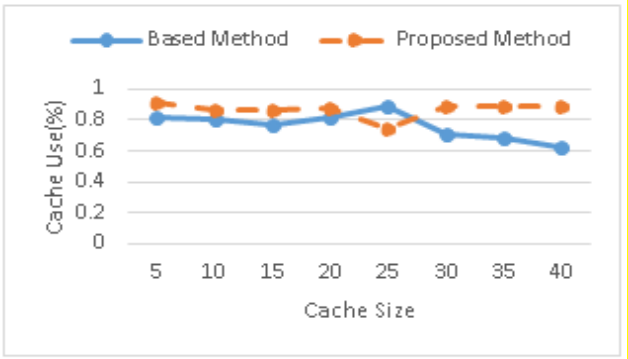


Fig. 7 Cache use with different error rates

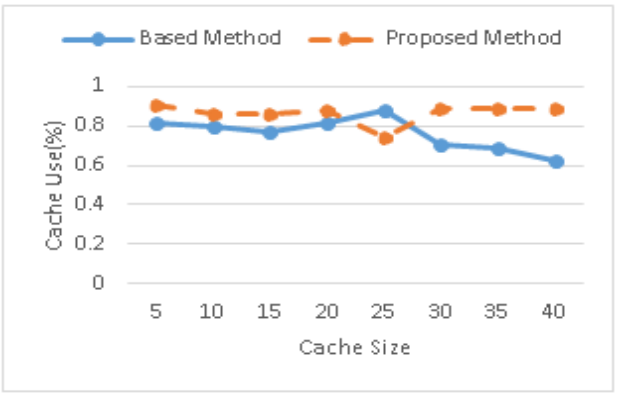


Fig. 8 Cache use with different cache sizes

4-4- Throughput

Throughput as another service quality parameter, plays an important role in determining the quality of the compared methods. Therefore, in this section, the throughput is examined. In Figure 9 and Figure 10, the throughput of each method is reported for different scenarios.

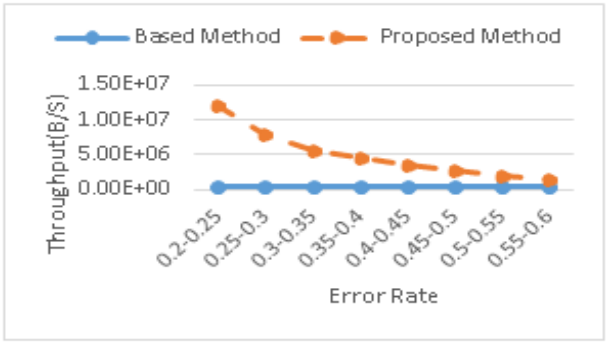


Fig. 9 Throughput with different error rates

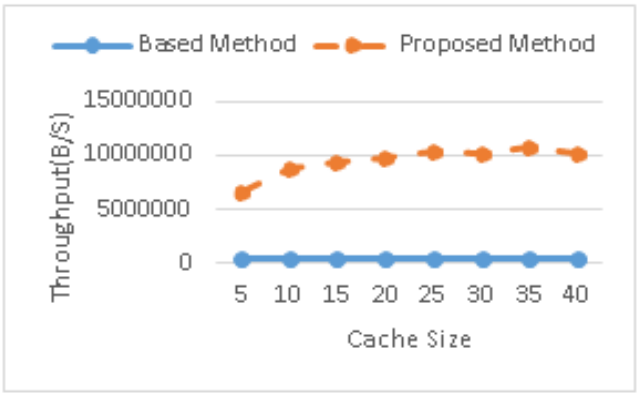


Fig. 10 Throughput with different cache sizes

5- Conclusion and Future Works

Wireless sensor networks are widely used to perform automation in many applications. The WSN is used in both attended and unattended environments such as the Internet of Things, smartphones, health monitoring, surveillance, volcano monitoring, border surveillance, and more The IoT-based WSN is emerging rapidly because of its versatility and economic nature.In many applications of wireless sensor networks, providing reliability and healthy delivery of the packet to the destination is of great importance. Reliability is one of the tasks of the transport layer in these networks, which gives the network the ability to deliver data sent to the receiver securely. In this paper, a protocol for providing statistical reliability for multiple traffic flows using dynamic caching capability is presented. In order to make optimal use of memory and energy, distributed dynamic caching methods have been considered. On the other hand, in many applications of wireless sensor networks, providing 100% reliability is not considered and statistical reliability is also required. In this dissertation, a new protocol for recovering lost packets in these networks has been presented, which has the following features:

• Suitable for applications that need to provide statistical reliability. Naturally, this protocol can also be used for applications with 100% reliability.

• Ability to support various flows with different characteristics (different packet lengths).

• Act in caching mode to balance energy consumption and memory.

• In calculating the probability of loss, all the factors that produce loss, including channel noise, node failure, and congestion, are considered as the input of the deep learning network.

• Caching points are dynamic and are dynamically determined according to the needs of traffic flows and network conditions.

• Has the ability to support different traffic classes.

• Can be extended to heterogeneous wireless sensor networks. Implementations of different scenarios in the results show that the proposed method works better in examining the evaluation criteria of transmission costs, end-to-end delays, cache use, and throughput.

References

[1]. A. Khalifeh et al., “Wireless sensor networks for smart cities: Network design, implementation and performance evaluation, Electron” vol. 10, no. 2, pp. 1–28, 2021, doi: 10.3390/electronics10020218.

[2]. A. Shankar, N. R. Sivakumar, M. Sivaram, A. Ambikapathy, T. K. Nguyen, and V. Dhasarathan, “Increasing fault tolerance ability and network lifetime with clustered pollination in wireless sensor networks,” J. Ambient Intell. Humaniz. Comput., vol. 12, no. 2, pp. 2285–2298, 2021, doi: 10.1007/s12652-020-02325-z.

[3]. Kabashkin, I., Reliability of Cluster-Based Nodes in Wireless Sensor Networks of Cyber Physical Systems. Procedia Computer Science, 2019. 151: p. 313-320.

[4]. W. Osamy, A. A. El-Sawy, and A. Salim, "CSOCA: Chicken Swarm Optimization Based Clustering Algorithm for Wireless Sensor Networks," IEEE Access, vol. 8, pp. 60676–60688, 2020, doi: 10.1109/ACCESS.2020.2983483.

[5]. S. K. S. L. Preeth, R. Dhanalakshmi, and P. M. Shakeel, "An intelligent approach for energy efficient trajectory design for mobile sink based IoT supported wireless sensor networks," Peer-to-Peer Netw. Appl., vol. 13, no. 6, pp. 2011–2022, 2020, doi: 10.1007/s12083-019-00798-0.

[6]. Zhou, Y., et al., Topology design and cross-layer optimization for wireless body sensor networks. Ad Hoc Networks, 2017. 59: p. 48-62.

[7]. R. Arya and S. C. Sharma, "Analysis and optimization of energy of Sensor node using ACO in wireless sensor network," Procedia Comput. Sci., vol. 45, no. C, pp. 681–686, 2015, doi: 10.1016/j.procs.2015.03.132.

[8]. Singh, A., Sharma, S. and Singh, J., 2021. Nature-inspired algorithms for wireless sensor networks: A comprehensive survey. Computer Science Review, 39, p.100342.

[9]. A. Singh, S. Sharma, and J. Singh, "Nature-inspired algorithms for Wireless Sensor Networks: A comprehensive survey," Comput. Sci. Rev., vol. 39, 2021, doi: 10.1016/j.cosrev.2020.100342.

[10]. H. Zhong, L. Shao, J. Cui, and Y. Xu, "An efficient and secure recoverable data aggregation scheme for heterogeneous wireless sensor networks," J. Parallel Distrib. Comput., vol. 111, pp. 1–12, 2018, doi: 10.1016/j.jpdc.2017.06.019.

[11]. N. Abbas and F. Yu, "Design and Implementation of a Video Surveillance System for Linear Wireless Multimedia Sensor Networks," 2018 3rd IEEE Int. Conf. Image, Vis. Comput. ICIVC 2018, pp. 524–527, 2018, doi: 10.1109/ICIVC.2018.8492776.

[12]. M. Mohamed-Lamine, "New clustering scheme for wireless sensor networks," 2013 8th Int. Work. Syst. Signal Process. Their Appl. WoSSPA 2013, pp. 487–491, 2013, doi: 10.1109/WoSSPA.2013.6602412.

[13]. Meenakshi, D. and Kumar, S., 2012. "Energy Efficient Hierarchical Clustering Routing Protocol for Wireless Sensor Networks". Springer, pp. 409-420.

[14]. N. Goel and G. Aujl, "Simulation and feasibility analysis: Hierarchical Energy Efficient Routing Protocol (HEERP) for Wireless Sensor Network," Int. Conf. Commun. Signal Process. ICCSP 2013 - Proc., pp. 1143–1148, 2013, doi: 10.1109/iccsp.2013.6577235.

[15]. T. Ducrocq, N. Mitton, and M. Hauspie, "Energy-based clustering for wireless sensor network lifetime optimization," IEEE Wirel. Commun. Netw. Conf. WCNC, pp. 968–973, 2013, doi: 10.1109/WCNC.2013.6554695.

[16]. R. Mhemed, N. Aslam, W. Phillips, and F. Comeau, "An energy efficient fuzzy logic cluster formation protocol in wireless sensor networks," Procedia Comput. Sci., vol. 10, pp. 255–262, 2012, doi: 10.1016/j.procs.2012.06.035.

[17]. P. Zhang, G. Xiao, and H. P. Tan, "Clustering algorithms for maximizing the lifetime of wireless sensor networks with energy-harvesting sensors," Comput. Networks, vol. 57, no. 14, pp. 2689–2704, 2013, doi: 10.1016/j.comnet.2013.06.003.

[18]. Y. Shen and X. Li, "Wavelet neural network approach for dynamic power management in wireless sensor networks," Proc. Int. Conf. Embed. Softw. Syst. ICESS 2008, pp. 376–381, 2008, doi: 10.1109/ICESS.2008.36.

[19]. Oldewurtel F, Mahonen P. Neural wireless sensor networks. In2006 International Conference on Systems and Networks Communications (ICSNC'06) 2006 Oct 6 (pp. 28-28). IEEE.

[20]. Chou LD, Li DC, Hong WY. Improving energy- efficient communications with a battery lifetime- aware mechanism in IEEE802. 16e wireless networks. Concurrency and computation: Practice and experience. 2013 Jan;25(1):94-111.

[21]. P. Mohanty and M. R. Kabat, "A hierarchical energy efficient reliable transport protocol for wireless sensor networks," Ain Shams Eng. J., vol. 5, no. 4, pp. 1141–1155, 2014, doi: 10.1016/j.asej.2014.05.009.

[22]. B. Sharma and T. C. Aseri, "A hybrid and dynamic reliable transport protocol for wireless sensor networks," Comput. Electr. Eng., vol. 48, pp. 298–311, 2015, doi: 10.1016/j.compeleceng.2015.01.007.

[23]. Tiglao, N.M.C. and A.M. Grilo. Transmission Window Optimization for Caching-Based Transport Protocols in Wireless Sensor Networks. 2015. Cham: Springer International Publishing.

[24]. Alipio MI, Tiglao NM. RT-CaCC: a reliable transport with cache-aware congestion control protocol in wireless sensor networks. IEEE Transactions on Wireless Communications. 2018 Apr 24;17(7):4607-19.

Journal of Information Systems and Telecommunication
Vol.11, No.2, April-June 2023, 135-148
http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

Representing a Novel Expanded Version of Shor's Algorithm and a Real-Time Experiment using IBM Q-Experience Platform

Sepehr Goodarzi[1] , Afshin Rezakhani[1 *], Mehdi Maleki[1]

[1]. Department of Computer Engineering, Ayatollah Boroujerdi University, Boroujerd, Iran

Received: 24 Dec 2021/ Revised: 16 Apr 2022/ Accepted: 17 May 2022

Abstract

The data are stored on the memory of the classical computer in small units of classical bits, which could be either 0 or 1. However, on a Quantum Computer, The Quantum States of each Quantum Bit (Qbit), would be every possible number between 0 and 1, including themselves. By placing the photons on a special state, which is a spot located at the middle of the two-dimensional space vectors $\binom{1}{0}$ and $\binom{1}{1}$ on the Unit Circle, which is called Superposition and we can take advantage of properties of this state when we place lots of vectors of N-dimensional spaces in superposition and we can do a parallelization and factorization for getting significant speedup. In fact, in Quantum Computing we are taking advantage of Quantum Dynamic Principles to process the data, which Classical Computers lack on, by considering the limitations of logical concepts behind them. Through this paper, we expand a quantum algorithm for the number of n Qbits in a new way and by implementing circuits using IBM-Q Experience, we are going to have some practical results, which are more obvious to be demonstrable. By expanding the Quantum Algorithms and using Linear Algebra, we can manage to achieve the goals at a higher level, the ones that Classical Computers are unable to perform, as machine learning problems with complicated models and by expanding the subject we can mention majors in different sciences like Chemistry (predicting the Structure of proteins with higher percentage accuracy in less period), Astronomy and so on.

Keywords: Quantum Computer; Quantum Dynamics; Unit Circle; N-dimensional Space; IBM-Q Experience.

1- Introduction

Nowadays, the act of factorization in mathematics is so popular among scientists and also engineers due to the extensive applications coming with it. Applications such as the optimization of the processing algorithms, which are exclusively written to be processed on Graphics Processing Units, which can run a lot faster than Central Processing Units due to their special parallelization and so on [1, 2]. Another capability of factorization is that it can help us solve differential matrix equations on a large-scale and would be practical in Linear Algebra [3], therefore it would be practical in computer science by considering the foundation of computers, which every classical bit is defined by matrices and the operation on them would be matrices and matrices are coming from Linear Algebra [4], therefore it is directly related to computer science. For example, it has been proved we can pull out more functional properties of some specific data across non-linear mappings, negative valued data processing, and also reviewing the data with only known relationships with them and all the three mentioned properties have been obtained by using the non-negative matrix factorization in a publication [5]. This subject is not related just to some specific branches of science and it is useful in many subjects. As an example, the simulation of materials is so important, given that if it would be possible to simulate the exact construction of a specific protein, then we would be able to cure many diseases since many of them are caused by the lack of some protein or shortages in the structure proteins [6]. By considering the amino acids, which are the main formational units of proteins, they can create many different types of proteins, according to the chemical structure they have and the types of chemical structure they can form. By expanding the data range, we are

✉ **Afshin Rezakhani**
rezakhani@abru.ac.ir

going to need much more power and resources to process the data, but we can use some factorization methods to factor and classify data, which would be easier for computers to process and simulate but sometimes it becomes impossible to do such calculations using the classical computers, the ones which are working with classical bits [7].

Classical computers are too weak to process a huge amount of data using some algorithms and it may take thousands of years to get the results, so it would be useless in that case. This occurs due to their classical properties and their physical limitations [8]. we are going to be more precise on this subject, we are going to check a paper, which has been published and it is concentrated on reviewing some machine learning methods to calculate the protein secondary structure prediction. Proteins are molecules, which carry out essential subordinates in almost all operations in the human body. They are made up of amino acid macromolecules and there are about 500 amino acids out there but 20 of them are coded by the genome to construct the necessary proteins for the human body. In summary, by considering a collection of n amino acids, 2n types of proteins with different constructions are possible to build. As a result, for 250 amino acids, we are going to have, 2250 proteins, which is a large number of species and it needs a very long time, so many powerful hardware and so much power source to calculate every possible state, which is not efficient [9].

Decades ago, a scientist presented another method of computation varying from classical computation and it is called Quantum Computation, which uses Qbits rather than Classical bits. As we know, classical bits can be in either two states of 0 or 1, which are presented by $\binom{1}{0}$ as 0 and $\binom{0}{1}$ as 1 and there are the only states, which a classical bit can be at a time, but in quantum computation a Qbit, which has the same functionality as the bits in Classical Computing, but it can be in more states at a time. A Qbit can be 0 or 1 or both the 0 and 1 together, which is a quantum property of the underlying atomic particles. This is called "Super Position" in Quantum Mechanics [10]. Scientists are using these properties to their advantage. Superposition means, that an Electron could be in both of the 0 and 1 states at once. According to the cause, Quantum Bits include Classical Bits as special states. These underlying atomic particles have more weird properties such as "Entanglement" and "supremacy" [11], which we are not going to talk about and the superposition is the most important part of our research. Another useful feature of a Quantum Computer is, that it is reversible, which works this way, because of its physical properties and limitations and we are going to take advantage of this property later on. There are four primary operations, which can be done on a single bit of information including, Identity, Negation, Constant-0 and Constant-1, which we are going to have more focus on, later in this paper. For an introduction, Identity and Negation are reversible, but Constant-0 and Constant-1 are irreversible and we are going to have to write them in a reversible way. By knowing a factor like being reversible or irreversible of an unknown function, we can do the factorization to get more properties of that function, which is very useful, as described earlier. In a classical computer, it can take 2 queries to process it by having one bit as an input to know if our unknown function is irreversible or not, whereas a quantum computer would do it on a single query, which is a massive speedup. As a result, for every 2 bits, we are going to need a single query, which is half the time and at the end, we are going to need $\sqrt{(2\text{^}n)}$ queries for n bits.

At first, we had an abstract of the whole paper. After that, we considered four sections in the introduction. In the first part of the introduction, we checked the importance and the capabilities of the factorization operation, then we reviewed the lack of classical computers in calculating the properties of unknown functions in order to factor and classify those properties. After that, by addressing the quantum mechanics and quantum computer properties, we claimed we are able to do a calculation to get the mentioned property on a quantum computer with half the time for getting the same result on a classical computer. Now we are reviewing the sections of the paper and then we are going to have the algorithm of doing the calculation on a quantum computer on paper and the result of an experiment on a real quantum computer using the cloud-base framework IBM has introduced for the researchers to have access to real quantum computers by aiming the real experiments on real quantum computers. At last, we are going to have a discussion and then the references. By using the mentioned methods and, at most of them, we must do the whole calculations of the existing entities of the function to get to know more about the features of the function in order to do some factorizations regarding to increase the pace of further calculations.

2- Related Works

Major In general, mathematical thinking and computation has been doing an important role in our lives and the advancement of various sciences. We are going to discuss a number of the related research jobs, which has been done in this area a few years ago and we are going to know the importance of the discussed subject in the following section. Our special subject in

the discussion would be the pros of mathematical factorization on algorithms and the approach to it.

A recent work, which has been done earlier, shows the benefits of Maximization-Factorization Statics on an advance in computer science, which is Blockchain Technology and also Internet-of-Things, which has been published in IEEE Internet of Things Journal. The pros include using less memory and power and also they are the less iteration to converge to the consensus solution and easiness to configure the complete mathematical model as per the requirement [12]. Another work has been published on PPoPP '21: Proceedings of the 26th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming, in which the authors have proposed a method of deriving parallel I/O lower bounds for the programs and they derive COnfLUX, an LU algorithm with the parallel I/O cost of N3/([EQUATION]) communicated elements per processor - only 1/3× over our established lower bound, which would be considered as a massive speedup [13]. Through another one, the authors show, that the D-Wave 2X can be effectively used as part of an unsupervised machine learning method. The used method takes a matrix as input and produces two low-rank matrices as output—one containing latent features in the data and another matrix describing how the features can be combined to approximately reproduce the input matrix. Despite the limited number of bits in the D-Wave hardware, this method is capable of handling a large input matrix. The D-Wave only limits the rank of the two output matrices. They applied their method to learn the features from a set of facial images and compare the performance of the D-Wave to two classical tools [14]. Another job, which has been published on IEEE Access, proposes two techniques for overcoming load-imbalance encountered when implementing so-called look-ahead mechanisms in relevant dense matrix factorizations for the solution of linear systems. The first technique promotes worker sharing (WS) between the two tasks, allowing the threads of the task that completes first to be reallocated for use by the costlier task. The second technique allows a fast task to alert the slower task of completion, enforcing the early termination (ET) of the second task, and a smooth transition of the factorization procedure into the next iteration [15]. In another research, a person named Reid Atcheson, who is a Ph.D. in Computational and Applied Mathematics, proposed a way to use non-Euclidean norms to formulate a QR-like factorization which can unlock interesting and potentially useful properties of non-Euclidean norms - for example, the ability of l1 norm to suppresses outliers or promote sparsity. At the end of his paper, he confirmed the results using python [16]. Through another work, the authors propose a novel solution to this problem: at the mathematical level, we reduce the computational requirement by exploiting the data sparsity structure of the matrix off-diagonal tiles utilizing low-rank approximations; and, at the programming-paradigm level, we integrate PaRSEC, a dynamic, task-based runtime to reach unparalleled levels of efficiency for solving extreme-scale linear algebra matrix operations. The paper has been published on PASC '20: Proceedings of the Platform for Advanced Scientific Computing Conference [17]. In another research, the authors have proposed to factorize the matrix using a “lattice HH-matrix” format that generalizes the BLR format by storing each of the blocks (both diagonals and off-diagonals) in the HH-matrix format. These blocks stored in the HH-matrix format are referred to as lattices. Thus, this lattice format aims to combine the parallel scalability of BLR factorization with the near-linear complexity of HH-matrix factorization. At the very first step, they compared factorization performances using the HH-matrix, BLR, and lattice HH-matrix formats under various conditions on a shared-memory computer. The performance results show that the lattice format has storage and computational complexities similar to those of the HH-matrix format, and hence a much lower cost of factorization than BLR [18]. In another research, the authors describe efficient algorithms for computing rank-revealing factorizations of matrices that are too large to fit in RAM, and must instead be stored on slow external memory devices such as solid-state or spinning disk hard drives (out-of-core or out-of-memory). They propose separate methods. The first is a blocked version of column pivoted Householder QR, organized as a "left-looking" method to minimize the number of write operations (which are more expensive than reading operations on a spinning disk drive). The second method results in a so-called UTV factorization which expresses a matrix A as A=UTV× where U and V are unitary, and T is triangular. This method is organized as an algorithm-by-blocks, in which floating-point operations overlap read and write operations [19]. In another work, the authors proposed a distributed high-performance parallel implementation of the BPMF using Gibbs sampling on shared and distributed architectures. They have shown by using efficient load balancing using work-stealing on a single node, and by using asynchronous communication in the distributed version they beat state-of-the-art implementations [20]. Through another paper, which has been published in IEEE Transactions on Services Computing, they addressed a privacy bug in clouds by presenting a novel outsourced scheme for NMF (O-NMF), which aims to lessen clients' computing burden and tackle security problems faced by outsourcing NMF [21]. Through another research, which has been published in 2017 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), Batch matrix operations address the case of

solving the same linear algebra problem for a very large number of very small matrices. They focused on implementing the batch Cholesky factorization in CUDA, in single-precision arithmetic, for NVIDIA GPUs and also the benefits of using noncanonical data layouts, where consecutive memory locations store elements with the same row and column index in a set of consecutive matrices [22]. In another work, the authors provide a comprehensive survey of mixed-precision numerical linear algebra routines, including the underlying concepts, theoretical background, and experimental results for both dense and sparse linear algebra problems [23]. Through another work, the authors provide techniques for Supernodal Sparse Cholesky factorization on a hybrid multicore platform consisting of a multicore CPU and GPU. The techniques are the subtree algorithm, pipelining and, multithreading. The subtree algorithm minimizes PCIe transmissions by storing an entire branch of the elimination tree in the GPU memory (the elimination tree is a tree data structure describing the workflow of the factorization) and also reduces the total kernel launch time by launching BLAS kernels in batches [24]. In another job, the authors highlight the necessary development of new instrumentation tools within the PaRSE task-based runtime system to leverage the performance of low-rank matrix computations. They demonstrate the benefits of these amenable tools while assessing the performance of TLR Cholesky factorization from data distribution, communication-reducing, and synchronization-reducing perspectives. The mentioned tool-assisted performance analysis results in three major contributions: a new hybrid data distribution, a new hierarchical TLR Cholesky algorithm, and a new performance model for tuning the tile size. The new TLR Cholesky factorization achieves an 8X performance speedup over existing implementations on massively parallel supercomputers, toward solving large-scale 3D climate and weather prediction applications [25]. And finally, in the last mentioning paper, the authors present a multithreaded method for Supernodal Sparse Cholesky factorization on a hybrid multicore platform consisting of a multicore CPU and GPU. The mentioned algorithm can utilize concurrency at different levels of the elimination tree by using multiple threads in both the CPU and the GPU. The elimination tree is a tree data structure describing the workflow of the factorization. The results on a platform consisting of an Intel multicore processor along with an Nvidia GPU indicate a significant improvement in performance and energy over a single-threaded Supernodal algorithm [26].

Table 1: Presenting the Related Works with Details

Number	***Publication***	***Description***	***Efficiency***
1	Kumar, G. et. al.,	Discussing an efficient statistical method with a proof-of-work consensus approach for cloud and fog computing	Less iteration to converge to the consensus solution and easiness to configure the complete mathematical model as per the requirement and also less energy and memory are needed
2	Kwasniewski, G. et. al.,	Presenting a method of deriving parallel I/O lower bounds for Dense linear algebra programs	Running on 1,024 nodes of Piz Daint, COnfLUX communicates 1.6× less than the second-best implementation and is expected to communicate 2.1× less on a full-scale and run on Summit
3	Lang, N. et. al.,	Proposing efficient algorithms for solving large-scale matrix differential equations	better performance of the proposed methods compared to earlier formulations
4	Likharev, K.K. et. al.,	The discussion of Fundamental limitations on the energy dissipated during one elementary logical operation	The limits due to classical and quantum statistics are shown to lie well below the earlier estimates, k B T and ηϑ, respectively
5	Malley, D.O. et. al.,	Representing a novel computational architecture and have attracted significant interest	The D-Wave 2X can be effectively used as part of an unsupervised machine learning method

6	Ostrouchov. et. al.,	Explaining the observed performance of sparse matrix factorization algorithms on parallel computers	Finishing with a parameterized model that is capable of reproducing the full range of behavior within these bounds, including the speedups observed in practice
7	Prijatmoko, D. et. al.,	Quantifying changes in body composition and compare methods for measuring body composition in alcoholic cirrhosis	With increasing severity of cirrhosis, total body water increased, whereas total body protein decreased with a significant decrease in serum albumin levels
8	Steffen, P. et. al.,	Updating the compiler to include a parallel backend, launching a large number of independent threads	Speedups ranging from 6.1× to 25.8× on an Nvidia GTX 280 through the CUDA libraries
9	MengTang. et. al.,	Presenting a multithreaded method for Supernodal sparse Cholesky factorization on a hybrid multicore platform consisting of a multicore CPU and GPU	Improvement in performance and energy over single-threaded Supernodal algorithm
10	MengTang. et. al.,	presenting techniques for Supernodal sparse Cholesky factorization on a hybrid multicore platform consisting of a multicore CPU and GPU	Improvement in performance and energy over CHOLMOD
11	Theurer, T. et. al.,	Introducing a rigorous resource theory framework for the quantification of superposition of a finite number of linear independent states	Establishing a strong structural connection between superposition and entanglement
12	VanderAa, T. et. al.,	Proposing a distributed high-performance parallel implementation of the BPMF using Gibbs sampling on shared and distributed architectures	Beating the state of the art implementations by using efficient load balancing and asynchronous communication
13	Wardah, W. et. al.,	Improving protein secondary structure prediction accuracy using neural networks	Producing better protein secondary structure prediction from higher architecture complexity
14	Windley, P. F. et. al.,	The problem of transposing a matrix in the store of a computer	Storing the data on the memory of a computer

15	Yamazaki, I. et. al.,	Parallelizing the LU factorization of a hierarchical low-rank matrix (HH-matrix) on a distributed-memory computer	Lower cost of factorization, which leads to a faster one
16	Zhang, D. et. al.,	Extend the original non-negative matrix factorization to kernel NMF	Extracting more efficient features, dealing with data through knowing the relations, processing negative values

3- Proposed Method

3-1- Problem Statement

As we all know, the four primary operations on a single bit of information are Identity, Negation, and Constants, including Constant-0, and Constant-1. As an intro, Identity and Negation are reversible but the Constants are irreversible. The proof is specified. A bit of information, which is either 0 or 1 and is stored on a Classical Computer and are considered as vectors. They are presented by $\binom{1}{0}$ and $\binom{0}{1}$, which are presenting the state 0 and 1 in order. The operations are also considered by matrices. By considering the multiplication of an operation into a bit, we are going to have a vector, which is the new state of the mentioned bit. Regardless of the state of the entry bit, the Constants Matrices and the results are going to be the same in every possible way, due to the essence of their construction, so they are irreversible and the vice versa applies to Identity and Negation. We can do some factorization by knowing the features of a function, which would give us the ability of faster processing. It can even make some impossibilities, possible due to the physical limitations of our processing units. To know if a function is reversible or not, it can take up to thousands of years to do the whole process on a classical computer. Imagine we have a single bit of entry information, it would take two queries for us to get to know if the function is reversible or not (by considering that we do not much information of our function, which would be so valuable in many areas in different sciences as Analytical Chemistry since it could help us generating new constructions of materials by solving useful equations). We should input 0 and 1 each time to get the results and then we can recognize if it is reversible or not by comparing the outputs of each query. For two bits, again we should test every possible state of each bit, which would be 4 queries. In summary, we should have two to the power of n (2n) for n bits of entry information. Now consider we have a function with thousands of entries and we want to know if it is reversible or not. In a simple word, we cannot calculate it, because it takes thousands of years to be processed and we are going to represent a way to calculate the mentioned factor of a function with n bits of entry information in a single query on a quantum computer, which would take infinite years of calculations on a classical one [27].

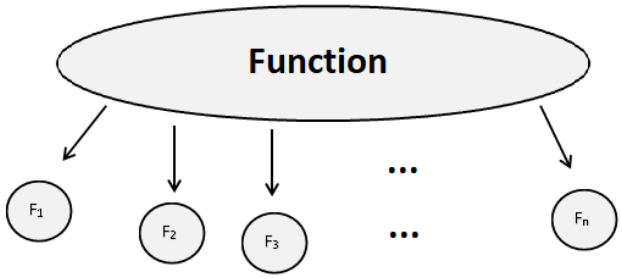


Fig. 1 Proposed beam former

3-2- Necessary Basics of Quantum Computing

In quantum computing, unlike the classical one, we have more than two states for each bit. A Quantum bit could be in the states of 0 and 1 and both and every possible number between them as states. According to this definition, the states of a classical bit would be a part of the states of a quantum bit, which is more recognizable in figure 2.

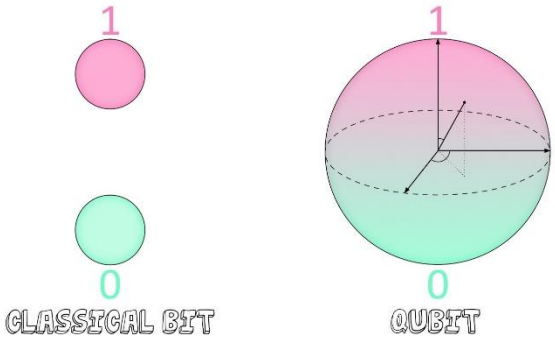


Fig. 2 The comparison of a classical bit (Cbit) with a quantum bit (Qbit)

3-3- Tensor Product

We represent the quantum state of 0 by |0⟩ and 1 by |1⟩. As we discussed earlier, we represent states of a bit using vectors, which is an m×1 matrix, in which m=2^n and we are able to represent it as the Tensor Products of n 2×1 vectors, which would be a lot helpful later on [28]. A Tensor Product of n vectors is:

$$\begin{pmatrix} a0 \\ a1 \\ ... \\ an \end{pmatrix} \otimes \begin{pmatrix} b0 \\ b1 \\ ... \\ bn \end{pmatrix} \otimes ... \otimes \begin{pmatrix} z0 \\ z1 \\ ... \\ zn \end{pmatrix} = \begin{pmatrix} a0 * b0 * ... * z0 \\ a1 * b1 * ... * z1 \\ ... \\ an * bn * ... * zn \end{pmatrix} \quad (1)$$

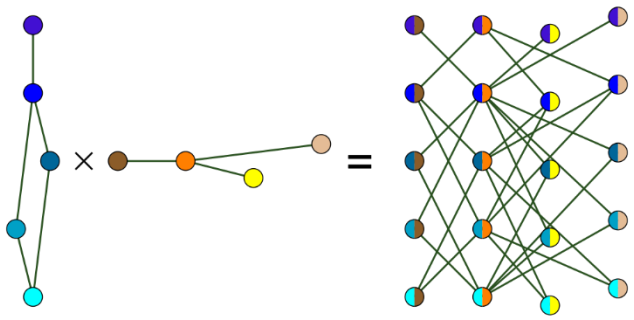


Fig. 3 The representation of Tensor Product using shapes (which is similar to the chemical structure of materials)

3-4- Quantum Gates

There are a number of quantum gates out there, but we need a few of them and we will discuss them [29].

3-4-1 CNOT Gate

CNOT gate is a 4×4 matrix and it changes the last two elements of the 4×1 matrix (Tensor Product of two vectors) [30]. It is represented by:

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad (2)$$

3-4-2 Bit-Flip Operator

It is the Negation in Classical Computing and is used to produce the symmetry of the input state [31]. The matrix is:

$$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad (3)$$

3-4-3 Hadamard Gate

This gate takes the input into a superposition, which is one of the four states:

$$\begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix}, \begin{pmatrix} \frac{-1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix}, \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix}, \begin{pmatrix} \frac{-1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix} \quad (4)$$

3-4-4 Identity and Constants

These are represented and called as the same as Classical Computing environment. Identity, Constant-0 and Constant-1 are represented in order:

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix} \quad (5)$$

3-5- Irreversible Quantity

Quantum Computers are irreversible devices, which means we do not have the ability to perform irreversible operations in the same way of reversible ones, as we were acting on classical computers [32]. It is not permitted by the physical reality according to the quantum mechanics and properties of a quantum computer. To solve this problem, we can have two Qbits as an entry instead of one.

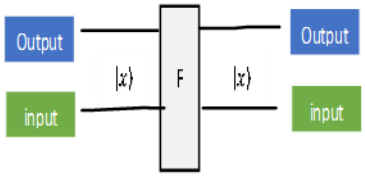


Fig. 4 The solution for having irreversible functions on a reversible computer

The input, which is labeled by output could be either 0 or 1 and there is not much difference between them, because the answer would be symmetry if we use the other one. In this particular subject, we use |0⟩. The important output to us would be the one, which has been labeled by output, which the function would operate on. Now we can reversibly use Constants.

3-6- Method

Now we are going to calculate an algorithm on a single Qbit using one query to get to know if the unknown

function is reversible or not in which we should have two queries on a classical computer [33]. We are going to need two Qbits as entries for each Qbit. At first, we are going to get them through a Bit-Flip operator and then a Hadamard Gate, and after that we will apply our unknown function and at the end another Hadamard Gate and finally measuring the results. So it would look like this:

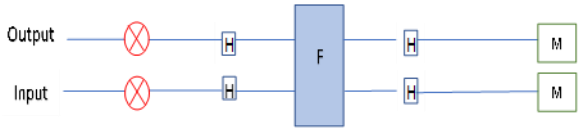


Fig. 5 The implemented Quantum Gates

4- Results and Discussion

At first, we apply the Bit-Flip and then the Hadamard Gate which gives the down results (by considering both the inputs at the state of $|0\rangle$):

Output[1]:

$[(0,1)(1,0)]^T \times [1,0]^T = [0,1]^T$, $[(1/\sqrt{2}, 1/\sqrt{2})(1/\sqrt{2}, -1/\sqrt{2})]^T \times [0,1]^T = [1/\sqrt{2} , -1/\sqrt{2}]^T$

(6)

Input:

$[(0,1)(1,0)]^T \times [1,0]^T = [0,1]^T$, $[(1/\sqrt{2}, 1/\sqrt{2})(1/\sqrt{2}, -1/\sqrt{2})]^T \times [0,1]^T = [1/\sqrt{2} , -1/\sqrt{2}]^T$

(7)

Now we apply each of the four primary gates and the Hadamard on them separately.

Constant-0:

Output:

$[1/\sqrt{2} , -1/\sqrt{2}]^T$ (No changes are applied) , $[(1/\sqrt{2}, 1/\sqrt{2})(1/\sqrt{2}, -1/\sqrt{2})]^T \times [1/\sqrt{2} , -1/\sqrt{2}]^T = [0,1]^T = |1\rangle$

(8)

Input:

$[1/\sqrt{2} , -1/\sqrt{2}]^T$ (No changes are applied) , $[(1/\sqrt{2}, 1/\sqrt{2})(1/\sqrt{2}, -1/\sqrt{2})]^T \times [1/\sqrt{2} , -1/\sqrt{2}]^T = [0,1]^T = |1\rangle$

(9)

So the result of measuring Constant-0 would be $|11\rangle$.

Constant-1:

Output:

$[(0,1)(1,0)]^T \times [1/\sqrt{2} , -1/\sqrt{2}]^T = [-1/\sqrt{2} , 1/\sqrt{2}]^T$,

$[(1/\sqrt{2}, 1/\sqrt{2})(1/\sqrt{2}, -1/\sqrt{2})]^T \times [-1/\sqrt{2} , 1/\sqrt{2}]^T = [0,1]^T = |1\rangle$

(10)

Input:

$[1/\sqrt{2} , -1/\sqrt{2}]^T$ (No changes are applied) , $[(1/\sqrt{2}, 1/\sqrt{2})(1/\sqrt{2}, -1/\sqrt{2})]^T \times [1/\sqrt{2} , -1/\sqrt{2}]^T = [0,-1]^T = |1\rangle$

(11)

As we see, the result of measuring Constant-1 would be $|11\rangle$ too.

Identity:

Output:

$[1/\sqrt{2} , -1/\sqrt{2}]^T$ (No changes are applied) , $[(1/\sqrt{2}, 1/\sqrt{2})(1/\sqrt{2}, -1/\sqrt{2})]^T \times [-1/\sqrt{2} , 1/\sqrt{2}]^T = [0,1]^T = |1\rangle$

(12)

Input:

$[(1,0,0,0)(0,1,0,0)(0,0,0,1)(0,0,1,0)] \times [[1/\sqrt{2}, -1/\sqrt{2}]^T \otimes [1/\sqrt{2}, -1/\sqrt{2}]^T]^T$

$= [1/\sqrt{2}, 1/\sqrt{2}]^T \otimes [1/\sqrt{2}, -1/\sqrt{2}]^T$, $[(1/\sqrt{2}, 1/\sqrt{2})(1/\sqrt{2}, -1/\sqrt{2})]^T \times [1/\sqrt{2}, 1/\sqrt{2}]^T = [0,1]^T = |0\rangle$

(13)

As a result, the measurement result of Identity was $|01\rangle$.

Negation:

Output:

$[(0,1)(1,0)]^T \times [1/\sqrt{2}, -1/\sqrt{2}]^T = [-1/\sqrt{2}, 1/\sqrt{2}]^T$, $[(1/\sqrt{2}, 1/\sqrt{2})(1/\sqrt{2}, -1/\sqrt{2})]^T \times [-1/\sqrt{2}, 1/\sqrt{2}]^T = [0,-1]^T = |1\rangle$

(14)

Input:

$[(1,0,0,0)(0,1,0,0)(0,0,0,1)(0,0,1,0)] \times [[1/\sqrt{2}, -1/\sqrt{2}]^T \otimes [1/\sqrt{2}, -1/\sqrt{2}]^T]^T$

$= [1/\sqrt{2}, 1/\sqrt{2}]^T \otimes [1/\sqrt{2}, -1/\sqrt{2}]^T$, , $[(1/\sqrt{2}, 1/\sqrt{2})(1/\sqrt{2}, -1/\sqrt{2})]^T \times [1/\sqrt{2}, 1/\sqrt{2}]^T = [0,1]^T = |0\rangle$

(15)

The result was the same as Identity, which is $|01\rangle$. As you saw, the results of the Identity and Negation were equal ($|01\rangle$), and the same happened for Constants ($|11\rangle$). It is now proved that we can run the whole process on a single query, in which we need two queries on a classical computer. For one entry Qbit if the result comes out $|01\rangle$, then the function is reversible, but if we get $|11\rangle$, then the function would be irreversible. We are going to discuss the n Qbits through the next table.

[1] A matrix can be represented in several ways. The best way to do this on a paper is to write it linearly. For example, the matrix $\begin{pmatrix} a11 & a12 & \dots & a1n \\ b21 & b22 & \dots & b2m \\ \dots & \dots & \dots & \dots \\ zn1 & zn2 & \dots & znm \end{pmatrix}$, which is a m × n matrix and it could be represented by:
[(a11,a12,…,a1m) (b21,b22,…,b2m)… (zn1,zn2,…,znm)]T, which saves a lot of space.

Table 2: Expanding the algorithm for n Qbits

Number of Qbits	*Reversible*	*Irreversible*
1	$\|01\rangle$	$\|11\rangle$
2	$\|0011\rangle$	$\|1111\rangle$
3	$\|000111\rangle$	$\|111111\rangle$
…	…	…
n	$\|00\ldots011\ldots1\rangle$	$\|11\ldots1\rangle$

By considering the calculations, we can say for n Qbits as entry, we are going to need 2×n Qbits of memory due to the properties, which discussed earlier, and then we would apply the algorithm and we will face two conditions, a state including 2×n of 1s which is $|11\ldots1\rangle$ and we can conclude that the function is irreversible, but if we face a state including n×0s and n×1s which would be $|00\ldots011\ldots1\rangle$, then the function is reversible. We got it in a single query.

5- Main Difference Between the Method and Shor's Algorithm

As you have observed, we have proven it is possible for the purposed method to function as well as Shor's algorithm, by the difference that other existing methods require much more complicated circuits than the method given in this paper. We will be comparing a few of them with the method brought up in this paper at the following table. The table is consisting of the best existing algorithms by considering required random access memory and the estimated time to implement the algorithm in order to get the hidden features of the given function, and as it is observable, the one brought up on this paper is going to get us going less than a minute by considering the implemented circuit and also the number of qubits, which we are going to need and in this case it would be two Qbits.

Table 3: Publications in this area

Publication	*Number of bits/Qbits required for the implementation of n bits/Qbits*	*Estimated time (s)*
Kumar, G. et. al.,	1024	(≈) > 150
Kwasniewski, G. et. al.,	512	(≈) > 70
Lang, N. et. al.,	128	(≈) > 20
Likharev, K.K. et. al.,	16	(≈) > 4
Our proposed method	2	(≈) > 1

6- Applying the Method on a Real Quantum Computer

IBM is a high-tech industry, which does an important role in the United States economy [34]. The concentration of such high-tech companies is on the combination of Science, Technology, Engineering, and Mathematics, which is called STEM for short [35]. It is one of the fewest industries, which has Quantum Computers. They provided a framework for scientists and engineers to do their research on real Quantum Computers online. It is called Qiskit, which has been written in python and is open source and free to use [36]. By creating an account on the IBM website, we got access to real Quantum Computers and have done a short process, due to the number of processes and also physical limitations, so we could do it on two Qbits, which has been considered for a single bit of entry. We have done it by defining the necessary gates and entries to get it into a lineup, which has been established by IBM and when it is our turn, after getting checked by their agents, they processed it on a real Quantum Computer and sent us back the results. We have calculated the results of each primary operation on a single Qbit, in which we need two of them and we have exported all the data we entered including the source codes and the graphical figure of the implemented gates and also we exported the results we got from the calculations including the plots of the probabilities and amplitudes on computational basis states and an extra 3D Q-sphere, which is the exact state of the Qbit and it is represented in a 3D plot but we are unable to put it on the paper, due to its 3D feature, but we put a 2D version of it on the paper.

6-1- Constant-0

The source code of the four designed circuits is available at:
https://github.com/sepehrgoodarzi6/The-four-primary-operations-on-a-Qbit.git
By measuring the results of the Qbits, we got the results showing on plots a 3D-sphere. According to the probability states and also amplitude, the measured probability of states of Qbits is a hundred percent $|11\rangle$[1] (which we have demonstrated in the previous section).

[1] There is no certainty in Quantum Mechanics and in fact, the probabilities are all we got, which are able to measure them so much faster than the certain results using classical computers and the point is that we are able to get to a point, which we can ensure we found the result we are looking for by repeating the calculation a few more time, which is still so much faster than any Classical Algorithm implemented on Classical Computers.

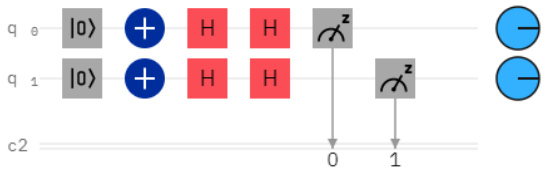


Fig. 8 The implemented circuit for the Constant-0 test

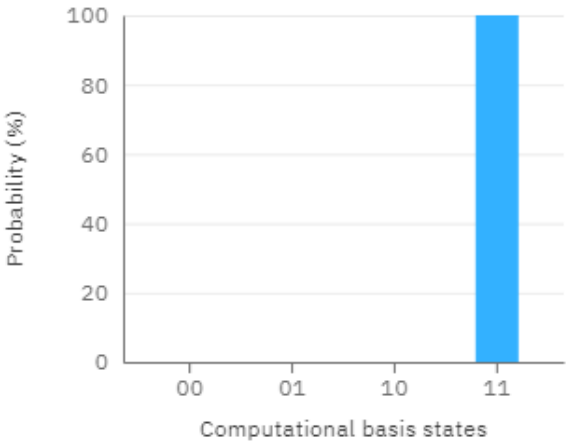


Fig. 9 The prediction of the probability states of Qbits of Constant-0 test

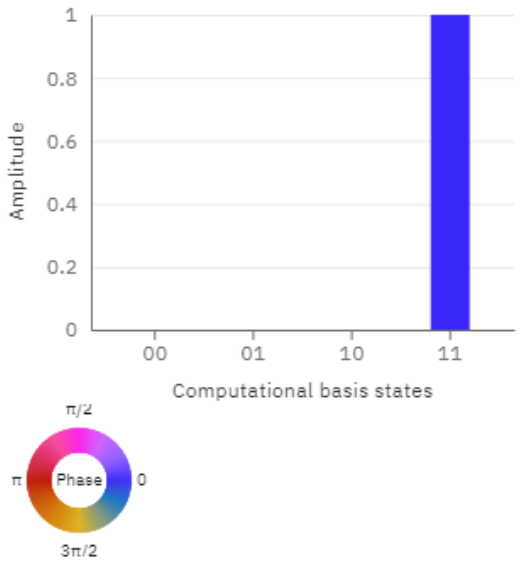


Fig. 10 The amplitude on computational basis states of Constant-0 test

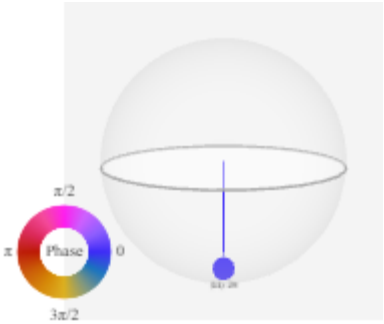


Fig. 11 2D version of the 3D-sphere of Constant-0 test

6-2- Constant-1

According to the probability states and also amplitude, the measured probability of states of Qbits is a hundred percent |11⟩, exactly the same as the results we got from testing Constant-0.

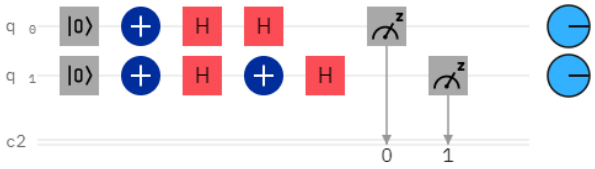


Fig. 12 The implemented circuit for the Constant-1 test

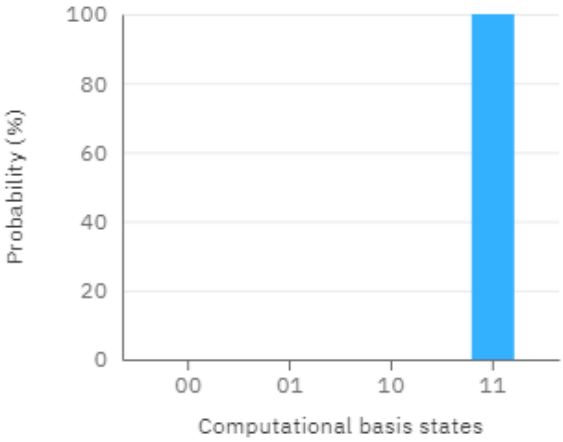


Fig. 13 The prediction of the probability states of Qbits of Constant-1 test

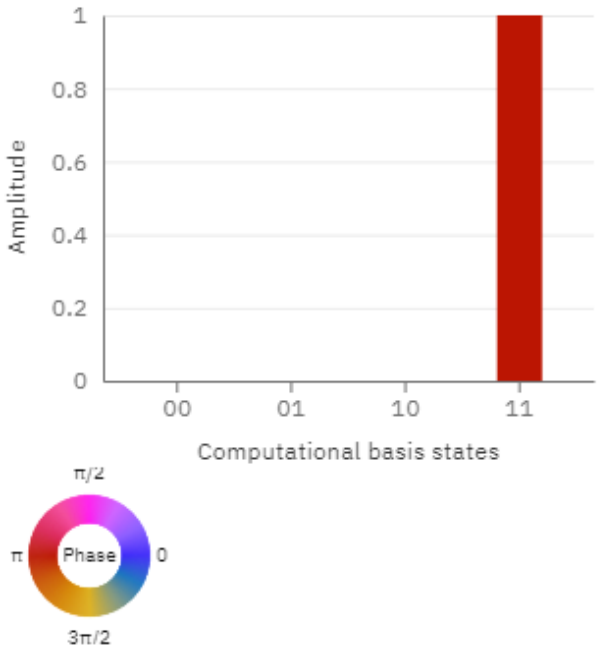


Fig. 14 The amplitude on computational basis states of Constant-1 test

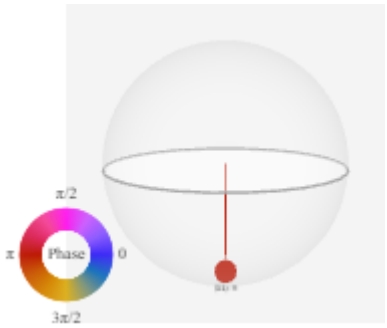


Fig. 15 2D version of the 3D-sphere of Constant-1 test

6-3- Identity

According to the probability states and also amplitude, the measured probability of states of Qbits is a hundred percent |10⟩, which varies from the results of the Constants.

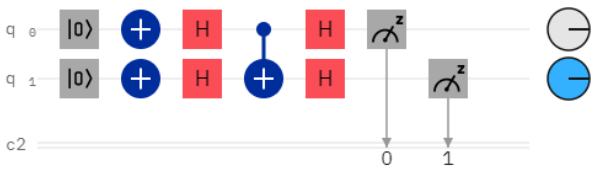


Fig. 16 The implemented circuit for the Identity test

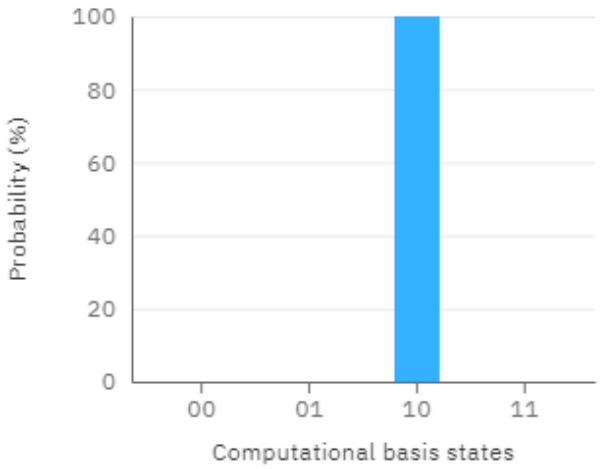


Fig. 17 The prediction of the probability states of Qbits of Identity test

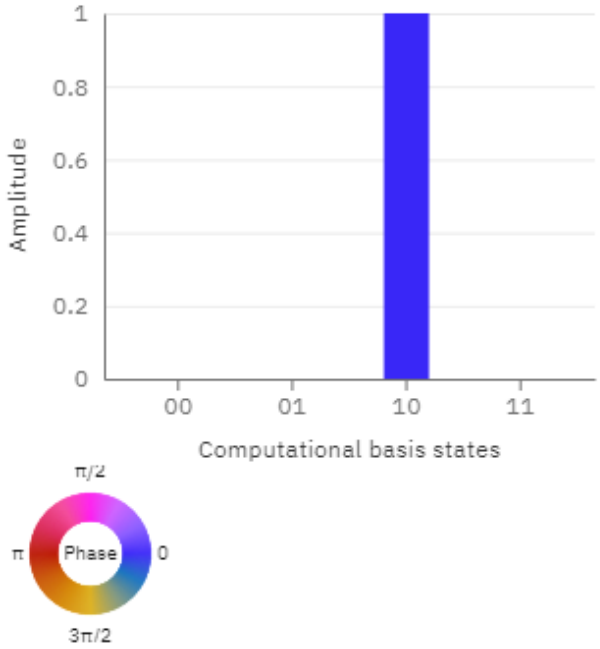


Fig. 18 The amplitude on computational basis states of Identity test

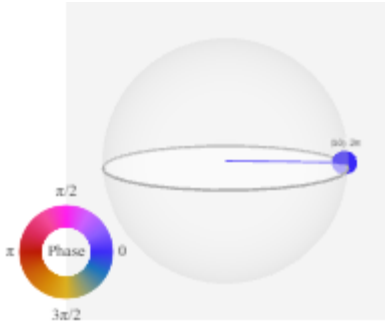


Fig. 19 2D version of the 3D-sphere of Identity test

6-4- Negation

According to the probability states and also amplitude, the measured probability of states of Qbits is a hundred percent |10⟩, which varies from the results of the Constants and is the same as Identity.

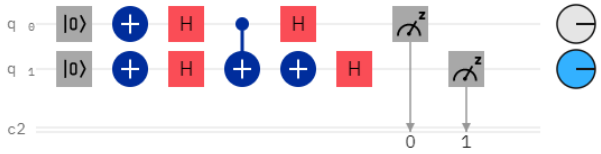


Fig. 20 The implemented circuit for the Negation test

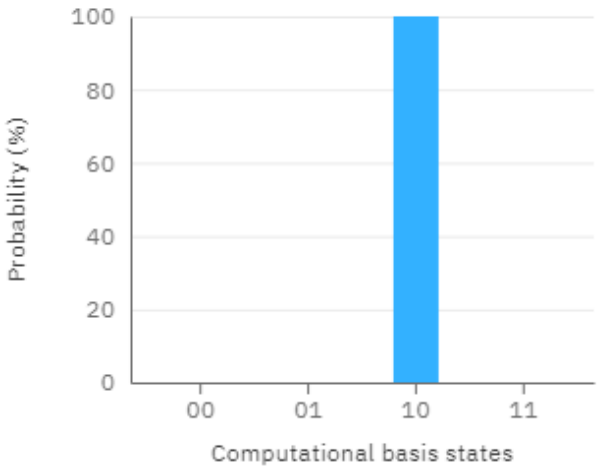


Fig. 21 The prediction of the probability states of Qbits of Negation test

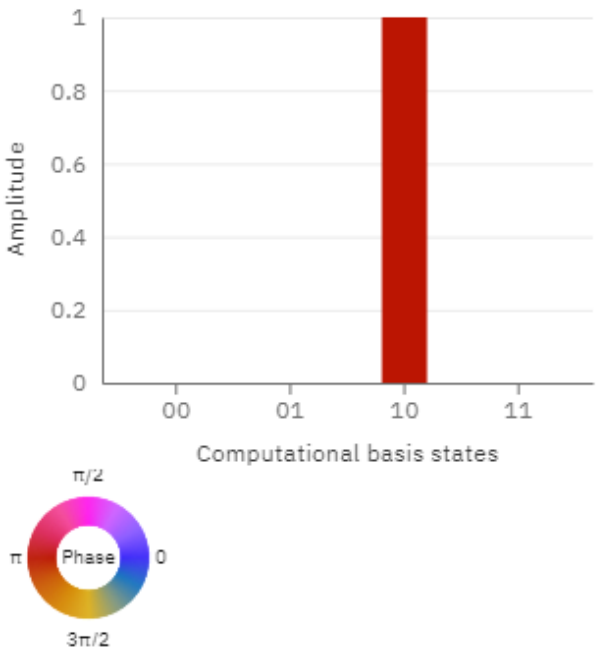


Fig. 22 The amplitude on computational basis states of Negation test

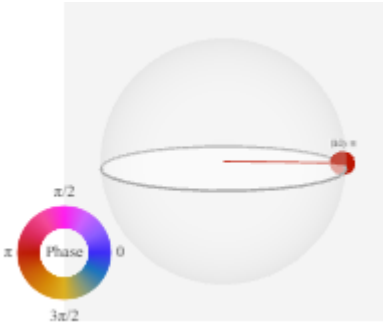


Fig. 23 2D version of the 3D-sphere of Negation test

6-5- Results Review

By measuring the results of the Qbits, we got the results showing on plots a 3D-sphere. According to the probability states and also amplitude, the measured probability of states of Qbits is a hundred percent |11⟩ (which we have demonstrated in the previous sections).
According to the probability states and also amplitude, the measured probability of states of Qbits is a hundred percent |11⟩, exactly the same as the results we got from testing Constant-0. By considering the probability states and also amplitude, the measured probability of states of Qbits of Negation is a hundred percent |10⟩, which varies from the results of the Constants.

According to the probability states and also amplitude, the measured probability of states of Qbits is a hundred percent |10⟩, which varies from the results of the Constants and is the same as Identity.
By having this information, we are going to be able to predict the results of these sorts of functions by testing two opponents of them.

7- Conclusions

In summary, we can model every single quantum bit of information existing all around the world, including the quantum world or many observable entities in a mathematical way, which would be Linear Algebra and by taking advantage of the principles of Quantum Dynamics, we can speed up the calculations by outperforming the best classical algorithms, which are implemented on the classical computers. By having these kinds of information, we are going to be able to predict the results of these sorts of functions by testing two opponents of them instead of calculating the whole entities, and by this method, we are going to be able to increase the pace of the calculations at a high rate. By speeding up the calculations, we can achieve the results of the most complex problems, which we are dealing with now, the ones we were not able to solve even in years and they took thousands of years like the ability to predict every possible Secondary Structure of Proteins and so on. But since the middle of the late 90th century, scientists have proved we can process them in a short amount of time if we have the real Quantum Computers with real Quantum Processors and now, we are achieving that goal. The only restriction of this method is that we cannot use it for Non-Quantum processing, and also they are not applicable in all subject of areas. In a few years, we are going to have a massive revolution in all the subject areas of Science and Technology.

References

[1] P. Steffen, R. Giegerich, and M. Giraud, "GPU parallelization of algebraic dynamic programming," in International Conference on Parallel Processing and Applied Mathematics, 2009: Springer, pp. 290-299.

[2] S. Asano, T. Maruyama, and Y. Yamaguchi, "Performance comparison of FPGA, GPU and CPU in image processing," in 2009 international conference on field programmable logic and applications, 2009: IEEE, pp. 126-131.

[3] N. Lang, H. Mena, and J. Saak, "On the benefits of the LDLT factorization for large-scale differential matrix equation solvers," Linear Algebra and its Applications, vol. 480, pp. 44-71, 2015.

[4] P. Windley, "Transposing matrices in a digital computer," The Computer Journal, vol. 2, no. 1, pp. 47-48, 1959.

[5] D. Zhang, Z.-H. Zhou, and S. Chen, "Non-negative matrix factorization on kernels," in Pacific Rim International Conference on Artificial Intelligence, 2006: Springer, pp. 404-412.

[6] D. Prijatmoko et al., "Early detection of protein depletion in alcoholic cirrhosis: role of body composition analysis," Gastroenterology, vol. 105, no. 6, pp. 1839-1845, 1993.

[7] L. S. Ostrouchov, M. Heath, and C. Romine, "Modeling speedup in parallel sparse matrix factorization," Oak Ridge National Lab., TN (USA), 1990.

[8] K. Likharev, "Classical and quantum limitations on energy consumption in computation," International Journal of Theoretical Physics, vol. 21, no. 3, pp. 311-326, 1982.

[9] W. Wardah, M. G. Khan, A. Sharma, and M. A. Rashid, "Protein secondary structure prediction using neural networks and deep learning: A review," Computational biology and chemistry, vol. 81, pp. 1-8, 2019.

[10] T. Theurer, N. Killoran, D. Egloff, and M. B. Plenio, "Resource theory of superposition," Physical review letters, vol. 119, no. 23, p. 230401, 2017.

[11] G. Gour and C. M. Scandolo, "Dynamical Entanglement," Physical Review Letters, vol. 125, no. 18, p. 180505, 2020.

[12] G. Kumar, R. Saha, M. K. Rai, R. Thomas, and T.-H. Kim, "Proof-of-work consensus approach in blockchain technology for cloud and fog computing using maximization-factorization statistics," IEEE Internet of Things Journal, vol. 6, no. 4, pp. 6835-6842, 2019.

[13] G. Kwasniewski, T. Ben-Nun, A. N. Ziogas, T. Schneider, M. Besta, and T. Hoefler, "On the parallel I/O optimality of linear algebra kernels: near-optimal LU factorization," in Proceedings of the 26th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming, 2021, pp. 463-464.

[14] D. O'Malley, V. V. Vesselinov, B. S. Alexandrov, and L. B. Alexandrov, "Nonnegative/binary matrix factorization with a d-wave quantum annealer," PloS one, vol. 13, no. 12, p. e0206653, 2018.

[15] S. Catalán, J. R. Herrero, E. S. Quintana-Ortí, R. Rodríguez-Sánchez, and R. Van De Geijn, "A case for malleable thread-level linear algebra libraries: The LU factorization with partial pivoting," IEEE access, vol. 7, pp. 17617-17633, 2019.

[16] R. Atcheson, "A Generalization of QR Factorization To Non-Euclidean Norms," arXiv preprint arXiv:2101.09830, 2021.

[17] Q. Cao et al., "Extreme-scale task-based cholesky factorization toward climate and weather prediction applications," in Proceedings of the Platform for Advanced Scientific Computing Conference, 2020, pp. 1-11.

[18] I. Yamazaki, A. Ida, R. Yokota, and J. Dongarra, "Distributed-memory lattice h-matrix factorization," The International Journal of High Performance Computing Applications, vol. 33, no. 5, pp. 1046-1063, 2019.

[19] N. Heavner, P.-G. Martinsson, and G. Quintana-Ortí, "Computing rank-revealing factorizations of matrices stored out-of-core," arXiv preprint arXiv:2002.06960, 2020.

[20] T. Vander Aa, I. Chakroun, and T. Haber, "Distributed Bayesian probabilistic matrix factorization," Procedia Computer Science, vol. 108, pp. 1030-1039, 2017.

[21] A. Fu, Z. Chen, Y. Mu, W. Susilo, Y. Sun, and J. Wu, "Cloud-based outsourcing for enabling privacy-preserving

large-scale non-negative matrix factorization," IEEE Transactions on Services Computing, 2019.
[22] M. Gates, J. Kurzak, P. Luszczek, Y. Pei, and J. Dongarra, "Autotuning batch Cholesky factorization in CUDA with interleaved layout of matrices," in 2017 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), 2017: IEEE, pp. 1408-1417.
[23] A. Abdelfattah et al., "A survey of numerical linear algebra methods utilizing mixed-precision arithmetic," The International Journal of High Performance Computing Applications, p. 10943420211003313, 2021.
[24] M. Tang, M. Gadou, S. Rennich, T. A. Davis, and S. Ranka, "Optimized sparse Cholesky factorization on hybrid multicore architectures," Journal of computational science, vol. 26, pp. 246-253, 2018.
[25] Q. Cao et al., "Performance analysis of tile low-rank cholesky factorization using parsec instrumentation tools," in 2019 IEEE/ACM International Workshop on Programming and Performance Visualization Tools (ProTools), 2019: IEEE, pp. 25-32.
[26] M. Tang, M. Gadou, and S. Ranka, "A Multithreaded Algorithm for Sparse Cholesky Factorization on Hybrid Multicore Architectures," Procedia Computer Science, vol. 108, pp. 616-625, 2017.
[27] M. Green, K. Glover, D. Limebeer, and J. Doyle, "AJ-Spectral Factorization Approach to H_∞," SIAM Journal on Control and Optimization, vol. 28, no. 6, pp. 1350-1371, 1990.
[28] P. Smolensky, "Tensor product variable binding and the representation of symbolic structures in connectionist systems," Artificial intelligence, vol. 46, no. 1-2, pp. 159-216, 1990.
[29] D. P. DiVincenzo, "Quantum gates and circuits," Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences, vol. 454, no. 1969, pp. 261-276, 1998.
[30] D. M. Zajac et al., "Resonantly driven CNOT gate for electron spins," Science, vol. 359, no. 6374, pp. 439-442, 2018.
[31] D. Riste et al., "Detecting bit-flip errors in a logical qubit using stabilizer measurements," Nature communications, vol. 6, no. 1, pp. 1-6, 2015.
[32] F. Benatti and R. Floreanini, Irreversible quantum dynamics. Springer Science & Business Media, 2003.
[33] S. Gulde et al., "Implementation of the Deutsch–Jozsa algorithm on an ion-trap quantum computer," Nature, vol. 421, no. 6918, pp. 48-50, 2003.
[34] M. Wolf and D. Terrell, "The high-tech industry, what is it and why it matters to our economic future," 2016.
[35] A. P. Carnevale, N. Smith, and M. Melton, "STEM: Science Technology Engineering Mathematics," Georgetown University Center on Education and the Workforce, 2011.
[36] A. Cross, "The IBM Q experience and QISKit open-source quantum computing software," in APS March Meeting Abstracts, 2018, vol. 2018, p. L58. 003.

Journal of Information Systems and Telecommunication
Vol.11, No.2, April-June 2023, 149-162
http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

A Novel Elite-Oriented Meta-Heuristic Algorithm: Qashqai Optimization Algorithm (QOA)

Mehdi Khadem[1], Abbas Toloie Eshlaghy[1*], Kiamars Fathi Hafshejani[2]

[1]. Department of Industrial Management, Science and Research Branch, Islamic Azad University, Tehran, Iran
[2]. Department of Industrial Management, South Tehran Branch, Islamic Azad University, Tehran, Iran

Received: 14 Mar 2022 / Revised: 15 Aug 2022 / Accepted: 19 Sep 2022

Abstract

Optimization problems are becoming more complicated, and their resource requirements are rising. Real-life optimization problems are often NP-hard and time or memory consuming. Nature has always been an excellent pattern for humans to pull out the best mechanisms and the best engineering to solve their problems. The concept of optimization seen in several natural processes, such as species evolution, swarm intelligence, social group behavior, the immune system, mating strategies, reproduction and foraging, and animals' cooperative hunting behavior.

This paper proposes a new Meta-Heuristic algorithm for solving NP-hard nonlinear optimization problems inspired by the intelligence, socially, and collaborative behavior of the Qashqai nomad's migration who have adjusted for many years. In the design of this algorithm uses population-based features, experts' opinions, and more to improve its performance in achieving the optimal global solution. The performance of this algorithm tested using the well-known optimization test functions and factory facility layout problems. It found that in many cases, the performance of the proposed algorithm was better than other known meta-heuristic algorithms in terms of convergence speed and quality of solutions.

The name of this algorithm chooses in honor of the Qashqai nomads, the famous tribes of southwest Iran, the Qashqai algorithm.

Keywords: Optimization; Meta-Heuristic algorithms; Qashqai Optimization Algorithm (QOA); Complexity; NP-hard problems; Swarm algorithms.

1- Introduction

The concept of optimization has expanded from engineering design to financial markets, stock markets, hospitality, tourism, and from our day-to-day activities to vacation planning and from computer science to industrial applications. An organization strives to maximize profits, minimize costs, and maximize its efficiency. Even when planning our vacation plans, we want to maximize utilization at the least cost (ideally free). [1]

Optimization problems categorize in terms of complexity into P problems, NP problems, NP-complete problems, and NP-hard problems. [2] The concept of problem complexity derived from computational complexity theory. Many of the critical issues are NP-hard, meaning that the time required to solve a sample in the worst-case grows exponentially with the size of the problem, so it isn't possible to solve such problems using exact methods in logical time. Many of our real-world problems are NP-hard, which means that a thorough and effective search is unlikely to meet our computational demand. [3] NP-hard problems include complex transportation network planning, data and computer network planning, human resources allocation, workshops, and machinery scheduling.

The approximate algorithms are dividing into three categories: heuristic, meta-heuristic, and hyper-heuristic. The basic principles of approximation algorithms are constructive heuristics and local search methods. [4]

The two main problems of the heuristic algorithms are their trapping at optimal local points and their early convergence. Heuristic algorithms presented to solve these problems. The main advantage of using meta-heuristic methods is the existence of limited assumptions in model formulation and no need for accurate search space information. At the same time, this is not the case in mathematical programming. Most nature-inspired or bio-

✉ Abbas Toloie Eshlaghy
toloie@gmail.com

inspired algorithms have been developed based on the successful evolutionary behavior of natural systems, learning from nature. Nature has solved complex problems for millions or even billions of years. In the environment, only the best and most sustainable solutions remain. [1]
The steps in this article are shown in Fig 1, respectively.

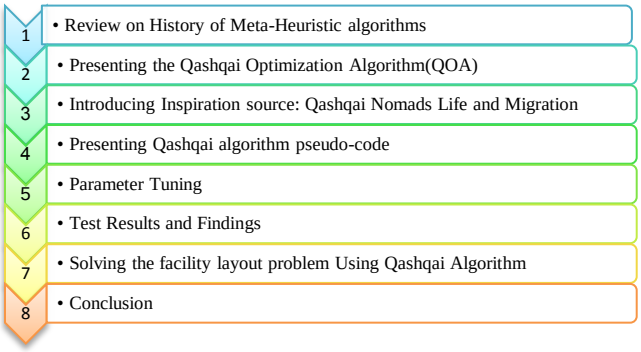


Fig. 1. Steps in this article

Eventually, the innovation of this research and the main objective is as follows:

- Presenting a novel Meta-heuristic algorithm with inspiration from Nomads' behavior, life, and Migration
- Ability to search the answer space extensively.
- It does not require gradient information.
- Requires few parameters to adjust.
- Find the optimal global answer or the near-optimal solution.
- It is elite-oriented and concatenates particular importance to swarm intelligence.

2- History of Meta-Heuristic Algorithms

A meta-heuristic optimization algorithm is a higher-level heuristic method that can be applied, especially with little information and with few modifications to search and find the optimal solution to various optimization problems. The use of meta-heuristic algorithms substantially increases the ability to find high-quality solutions to solve severe optimization problems. The characteristic of these algorithms is the use of exit from optimal local mechanisms. [5] The characteristic feature of meta-heuristic algorithms is to inspire biological and natural systems to solve complex optimization problems. The capabilities of these algorithms include the ability to search for vast spaces in low time efficiently, no need for the derivative of the objective function, the ability to evade from the local optimal points, low computational cost, and easy mathematics. These features have made these algorithms very attractive today. The development of meta-heuristic algorithms began in 1960. In recent decades, due to the ability and capability of nature-inspired algorithms
To solve various problems, the number of these algorithms has grown significantly and is growing. Fig 2 shows the cumulative number of meta-heuristic algorithms. [14]

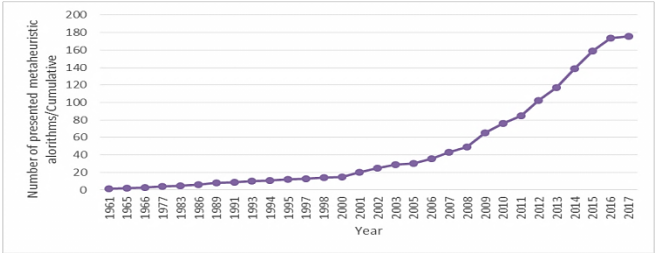


Fig. 2. The cumulative trend of presented Meta-Heuristic algorithms

Meta-Heuristic algorithms used in many areas of engineering design including structural optimization in electronics, aerodynamics, fluid dynamics, telecommunications, automotive, machine learning, data mining, computational biology, chemistry, and physics detection, signal and image processing, routing, scheduling, logistics and transportation, and supply chain management. Meta-heuristic algorithms are a subset of computational intelligence or soft computing, which is itself a subset of artificial intelligence. Fig 3 shows the place of meta-heuristic algorithms in artificial intelligence. [15]

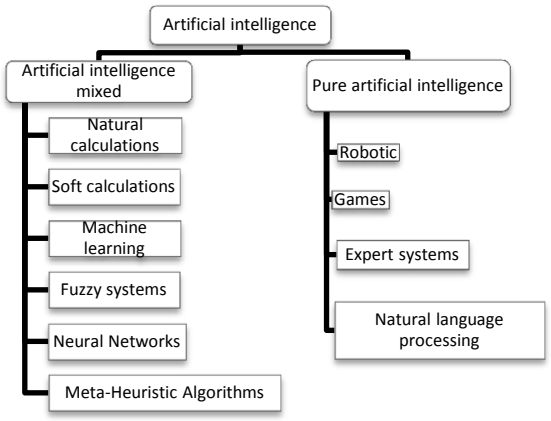


Fig. 3. The place of Meta-Heuristic algorithms in artificial intelligence

Fig 4 shows the general process of meta-heuristic algorithms to achieve the optimal solution.

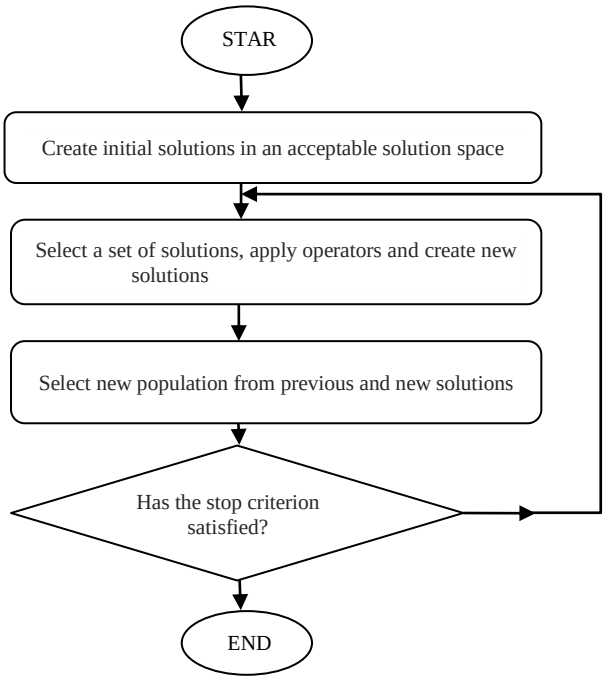


Fig. 4. General process of meta-heuristic algorithms

- **Research Gap**

The classification of the literature is addressed in Table 1. [14] Table 1. List of meta-heuristic algorithms with their specifications Table 1 shows that no metaheuristic algorithm inspired by nomadic life and migration has been proposed so far.

Therefore, the innovations of this research are:

- Study of nomadic life and migration and cooperative way of life and the role of nomadic elites in guiding migration.
- Presenting a novel meta-heuristic algorithm with high execution speed and achieving near-optimal solutions

Presenting an algorithm with low parameters that require low parameter tuning.

Table 1: List of meta-heuristic algorithms with their specifications

Row	Algorithm	Authors	Year	Brief Description	Ref
1	Red deer algorithm (RDA)	A. M. Fathollahi-Fard, M. Hajiaghaei-Keshteli, R. Tavakkoli-Moghaddam	2020	A new optimization algorithm inspired by red deer mating is developed. The Scottish red deer (Cervus Elaphus Scoticus) is a sub-species of red deer, which lives in the British Isles	[17]
2	Find-Fix-Finish-Exploit-Analyze (F3EA) meta-heuristic algorithm	A. H. Kashan, R. Tavakkoli-Moghaddam	2019	The F3EA algorithm is classified into the population based algorithm which simulates battleground and mimics the F3EA targeting process of object or installations selection for destruction in the warfare.	[18]
3	Tree Growth Algorithm (TGA)	A. Cheraghalipour, M. Hajiaghaei-Keshteli, M. M. Paydar	2018	The proposed algorithm is inspired by trees competition for acquiring light and foods	[19]
4	Whale Optimization Algorithm	Mirjalili, Seyedali Lewis, Andrew	2016	This algorithm imitates the hunting habits of whales.	[20]
5	Dragonfly Algorithm	Mirjalili, Seyedali	2016	Mimics from dragonflies behavior such as foraging and avoiding dangers.	[21]
6	African Buffalo Optimization	Odili, Julius Beneoluchi Kahar, Mohd Nizam Mohmad Anwar, Shahid	2015	Mimics foraging and organizational skills of African buffalos.	[22]
7	Ant Lion	Mirjalili, Seyedali	2015	Inspired by ant lions hunting behavior.	[23]
8	Ions Motion Algorithm	Javidy, Behzad Hatamlou, Mirjalili, Seyedali	2015	This algorithm imitates the movement of ions.	[24]
9	Monarch Butterfly Optimization	Wang, Gai-Ge Zhao, Xinchao Deb, Suash	2015	Monarch butterflies' massive journey from North America to California and Mexico inspire it.	[25]
10	Artificial Ecosystem Algorithm	Adham, Manal T Bentley, Peter J	2014	Inspired by characteristics of the ecosystem.	[26]
11	Grey Wolf Optimizer	Mirjalili, Seyedali	2014	Mimics the hierarchical leadership and hunting	[27]

Row	Algorithm	Authors	Year	Brief Description	Ref
		Mirjalili, Seyed Mohammad Lewis, Andrew		behavior of grey wolves.	
12	Keshtel Algorithm	Hajiaghaei-Keshteli, Aminnayeri, MJASC	2014	Based on the food searching strategy of a bird called a keshtel	[28]
13	Black Holes Algorithm	Hatamlou, Abdolreza	2013	Imitates the black hole's features.	[29]
14	Electro-magnetism Optimization	Cuevas, Erik Oliva, Diego Zaldivar, Daniel Pérez-Cisneros, Sossa, Humberto	2012	Electromagnetic problems inspire it.	[۳۰]
15	Flower Pollination Algorithm	Yang, Xin-She	2012	The pollination process of flowers inspires it.	[31]
16	Krill Herd	Gandomi, Amir Hossein Alavi, Amir Hossein	2012	Emulates the krills behavior in herding and searching food.	[32]
17	Bat Algorithm	Xin-She Yang	2010	Imitates the echolocation behavior of bats.	[33]
18	Cuckoo Search Algorithm	Yang, Xin-She and Deb, Suash	2009	Mimics the behavior of cuckoo in brood parasitic and levy flight.	[34]
19	Firefly Algorithm	Yang, Xin-She	2009	Fireflies flashing light behavior inspired it.	[35]
20	Imperialist Competitive Algorithm	Atashpaz-Gargari, Esmaeil Lucas, Caro	2007	Mimics the behavior of the imperialists in expanding their colonies.	[36]
21	Gravitational Search Algorithm	Webster, Barry Bernhard, Philip J	2003	The gravitational force inspires it.	[37]
22	Shuffled Frog Leaping Algorithm	Eusuff, Muzaffar M Lansey, Kevin E	2003	The collaborative behavior of frogs inspires it in search of food.	[38]
23	Honey-bees Mating Optimization Algorithm	Abbass, HA	2001	Honey bee's mating behavior inspires it.	[39]
24	Differential Evolution	Storn, Rainer Price, Kenneth	1997	It is inspired by the information exchange feature of chromosomes to generate better offspring.	[40]
25	Particle Swarm Optimization	Eberhart, Russell Kennedy, James	1995	The intelligent movements of bird swarms inspire it.	[41]
26	Cultural Algorithms	Reynolds, Robert G	1994	It is a Kind of evolutionary computation that has a knowledge component in addition to the population component.	[42]
27	Genetic Algorithm	Goldberg, David E	1989	The Darwinian evolution theory inspires it.	[43]
28	Tabu Search	Glover, Fred	1986	Enhancement of local search by essential rules modifications.	[44]

Dhiman and Kumar algorithms classify meta-heuristics into five categories of evolutionary algorithms, physics-based algorithms, swarm-based algorithms, biology-based algorithms, and nature-inspired algorithms. [6] Dalwani and Agrawal have inspired nature-based meta-heuristic algorithms into five categories of evolutionary algorithms, swarm-based algorithms, physics-based algorithms, biology-based algorithms, and other nature-inspired algorithms. [7] Memari and Ahmad classified the meta-heuristic algorithms into three categories of single-solution algorithms, population-based algorithms, and hybrid algorithms. [8] Birattari and Pokotti classify meta-heuristics algorithms into four continuous and discontinuous, population-based and single- solutions, memory-based and memory-free, single-neighbor, and multiple-neighbor, static and dynamic objective function, inspired by nature, and without inspiration. They were divided by nature. [9] Rajporehit, Sharma, and colleagues classified the meta-heuristic algorithms into three categories: evolutionary, logical search, and other nature-inspired algorithms. [10]

Most meta-heuristic algorithms inspired by natural species problem-solving strategies.

Features of algorithms inspired by species of insects, mammals, birds, and fishes include movement, routing, feeding, mating, reproduction, mass hunting, territorial protection, and risk aversion. The lifestyles of many species such as bacteria, frogs, dragonflies, fireflies, shrimp bunch, have also been inspired. Algorithms inspired by plant roots, photosynthesis, and pollination presented. Some of the natural Events, such as galaxies, river formation, chemical reactions, cloud formation, and crystals, gravitational forces, have also inspired to design the meta-heuristic algorithms. Algorithms inspired by sports competitions, training, learning, and imperialist competition also introduced as the smartest species inspired by the processes of human societies. Fig 5 shows

most of the articles presented in the field of meta-heuristic algorithms based on the source. [14]

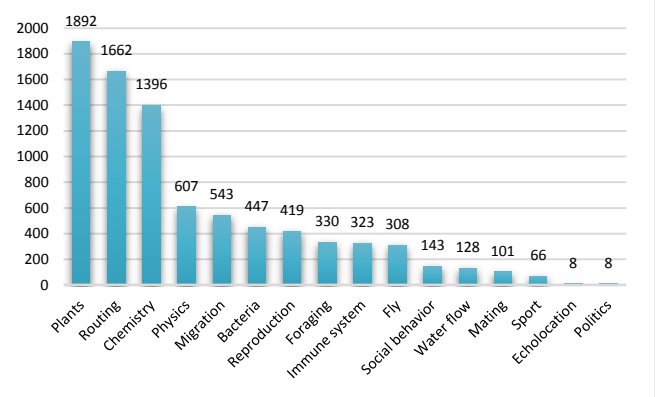


Fig. 5. Most of the papers in the field of Meta-Heuristic algorithms based on inspiration source (WOS source)

Due to the complexity of problems and the changing real world from 2000 onwards, we see the introduction of new Meta-Heuristic algorithms to solve various problems. Also, according to the NFL[1] theory proposed by Wolpert and McReady in 1997, there is no meta-heuristic algorithm suitable for solving all problems. For this reason, in this article, we have developed a new meta-heuristic algorithm inspired by the intelligent, cooperative, and social behavior of Qashqai nomads in their life and migration.

3- Presenting the Novel Qashqai Optimization Algorithm (QOA)

Organizing, designing, and testing the Qashqai Nomads' meta-heuristic algorithm was implemented in seven steps as follows:

3-1- Introducing Inspiration Source: Qashqai Nomads Life and Migration

The history of the nomads' life in Iran goes back about 8,000 years. Since the nomadic livelihoods depend on livestock, they have to migrate between the areas of Qeshlaq (winter quarters) and Yilagh (Summer quarters) to avoid the overwhelming heat and cold needed to provide forage.
Livestock, pasture, and migration are the three main pillars of immigrant human life. The nomad's immigration based on a timing plan for the tribal power structure. It is subject to the opinions of the tribal elders and Experts as well as the climatic and weather conditions, thus making it the best time for nomadic families and livestock to move from Qeshlaq to Yilagh areas. It is determined based on local experience and knowledge gained over many years.

Concerning the components and delicacies of the nomads' lives and migration and to the swarm intelligence of the nomads' movement over the many years that have resulted from the collective experience, perseverance, and collaboration of the tribe members, these experiences are transmitted from generation to generation intuitively and systematically. The name of this algorithm chosen in honor of the Qashqai tribe from the famous tribes of the southwest of Iran. [45-48, 55-57]

Fig. 6. Qashqai Nomads

3-2- The Basics of Algorithm Design

An algorithm is a problem-solving procedure involving a finite set of commands that must define as step-by-step and carefully. Each algorithm has a specific starting point and stop point and must consider all possible scenarios in solving different problem scenarios. An algorithm to solve the problem requires two sources of time and memory space. The behavioral analysis indices of the algorithm are the rate of efficiency, robustness in computing, rate of convergence, intelligence, Self-Adaptiveness, and error bound.
The time complexity function of the algorithm is a function such as f (n) that determines the time required to execute an algorithm based on n. The Big-O symbol is one of the universal symbols in the complexity analysis of algorithms. This symbol uses to calculate the complexity of an algorithm's time or memory space. [11]
The process of designing and implementing meta-heuristic algorithms has three successive stages. The first step is preparation, in which we need to get a detailed understanding of the problem we want to solve. The next step is called construction, the most important of which is to choose a solution strategy, to define performance metrics, and to design an algorithm for the solution strategy. The last step is implementation, in which utilization of the algorithm developed in the previous step, including parameter adjustment, performance analysis, and finally reporting of results, should be performed.

[1] No Free Lunch

3-3- Method of Designing the Qashqai Optimization Algorithm (QOA)

3-3-1- Creating an Initial Population (Initialization)

Suppose tribe has n members, each having a starting point (from Yilagh to Qeshlaq and vice versa). The starting point of each tribe member is random in the problem space.

3-3-2- Elite Selection (Elitism)

A practical variant of the general process of constructing a new population is to allow the best organism(s) from the current generation to carry over to the next, unaltered. This strategy is known as elitist selection and guarantees that the solution quality obtained by the Meta-heuristic algorithm will not decrease from one generation to the next. [49]
Usually, the tribes have their territory and are governed by Ilkhan or Il Big, under their leadership and management. Clan elders have rich experience of the best and the least risky paths to their memory and rely more on it in choosing paths. However, younger members of the tribe have a shorter experience and mind, so they are less likely to refer to their memory and rely more on their previous position. In contrast, the elders of the tribe are less likely to take their next place as their next move. This item has inspired us to update the algorithm's new locations of movement.

3-3-3- How to Update New Locations (Positions)

In this algorithm, the best cost function selects as Ilkhan (The head of the clan), and Formula one is used to updating a new place.

$$x_i^{t+1} = C_1 * \frac{fitness(pop(i)) - m_1}{m_2 - m_1} * x_i{}^t + C_2 * \frac{m_2 - fitness(pop(i))}{m_2 - m_1} * rand[varmin, varmax]$$

Formula One

The rest of the population update according to Formula two. m_1 is the best solution in each iteration, and m_2 is the worst solution.

$$x_i^{t+1} = C_1 * \frac{m_2 - fitness(pop(i))}{m_2 - m_1} * x_i{}^t + C_2 * \frac{fitness(pop(i)) - m_1}{m_2 - m_1} * rand[varmin, varmax]$$

Formula Two

Table 2 represents the parameters of the Qashqai optimization algorithm (QOA).

Table 2: Parameters of the Qashqai optimization algorithm (QOA)

Parameter	***Description***
Varmax	Maximum number of tribe members
x_i^t	The position of the ith member in iteration of t
x_i^{t+1}	The position of the ith member in the iteration of t+1
Pop(i)	The ith member of the tribe population
$fitness(pop(i))$	The fitnes function of a member of the i
	population of the tribe
Varmin	Minimum number of tribe members
C1, C2	Algorithm parameters
m_1	The best solution to each iteration
m_2	The worst solution (answer) of each iteration

3-3-4- Migration Route

A set of best-traversed points that equivalent to the best solutions in the algorithm for the general path of the migration.

3-3-5- Strategy to Prevent the Optimal Solution from Worsening

In this algorithm, an approach adopted to prevent the optimal solution from getting worse, so that if the optimal solution of the algorithm in one iteration were worse than the previous iteration of the algorithm, the worst of the current iteration would be replaced by the optimal point of the previous iteration. It will prevent the answer from getting worse.

3-3-6- Diversification and Intensification Strategy

The critical components of any meta-heuristic algorithm are intensification and diversification, or exploitation, and exploration. In the proposed algorithm, whatever greater focus on the previous position, have more exploitation. The less attention to the previous situation, result in more exploration.

3-3-7- Algorithm Stopping Conditions

Different conditions can consider for terminating the algorithm, such as specified execution time, a specified number of iteration, no response improvement.

3-4- Qashqai Optimization Algorithm (QOA) Pseudo-code

Table 3 shows the pseudo-code of the Qashqai optimization algorithm (QOA).

Table 3: Qashqai optimization algorithm (QOA) pseudo-code

Qashqai Optimization Algorithm (QOA) Pseudo Code
Result: Find The best solution Objective min or max $f(x), X = (x_1, x_2, \dots, x_d)^T$ Generate initial population, of n members of tribes(or nomads) Find the best solution g_* in the population in each iteration While(t<Max Iteration) or (stop criterion) do For i=1:n (all n members of each tribe's) m_1 ← The best solution(it) m_2 ← Worst solution(it) Update Position if pop(i) is the best solution of each Iteration then

$$x_i^{t+1} = C_1 * \frac{fitness(pop(i)) - m_1}{m_2 - m_1} * x_i{}^t + C_2 * \frac{m_2 - fitness(pop(i))}{m_2 - m_1} * rand[varmin, varmax]$$

else

$$x_i^{t+1} = C_1 * \frac{m_2 - fitness(pop(i))}{m_2 - m_1} * x_i{}^t + C_2 * \frac{fitness(pop(i)) - m_1}{m_2 - m_1} * rand[varmin, varmax]$$

end if

Evaluate new solutions

If new solutions are better, update them in the population

end for

Find the current best solution g_*

end while

3-5- Flowchart of the Qashqai optimization Algorithm (QOA).

Fig 7 shows a diagram of the Qashqai optimization algorithm (QOA).

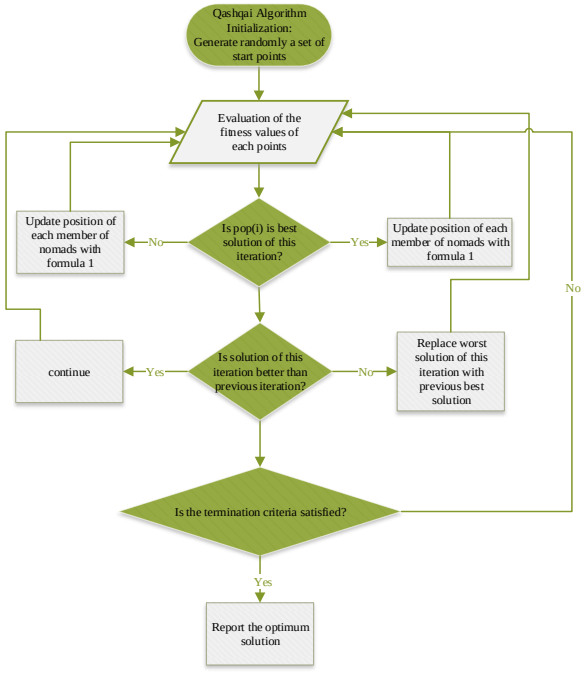


Fig. 7. Flowchart of Qashqai Optimization Algorithm (QOA)

3-6- Qashqai Optimization Algorithm (QOA) Parameter Tuning

Because input parameters influence the output of meta-heuristic algorithms, in order to adjust the parameters, the Taguchi method and Minitab software have been used. For example, according to Table 4, in five levels for the parameters of MaxIt, Npop, C1, and C2, the parameter adjustment of the Qashqai meta-heuristic algorithm has been made.

Table 4: Tuning the parameters of the Qashqai optimization algorithm (QOA)

Row	Parameter	Level 1	Level 2	Level 3	Level 4	Level 5
1	MaxIt	50	100	150	200	250
2	Npop	50	100	150	200	250
3	C1	0.1	0.5	1	2	3
4	C2	0.1	0.5	1	2	3

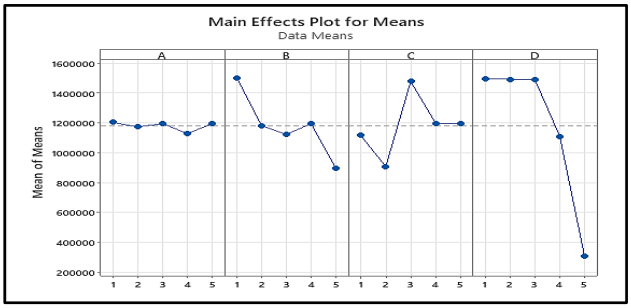


Fig. 8. Average diagram of means for each level of Qashqai optimization algorithm parameters.

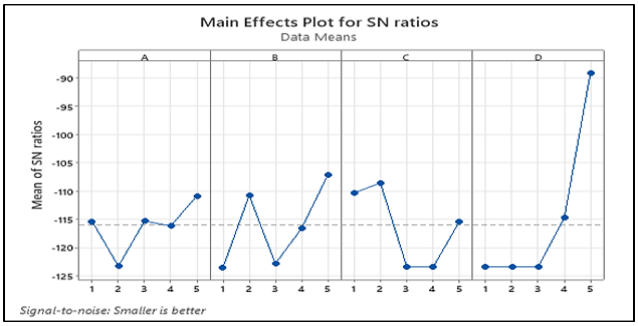


Fig. 9. Graph of the average S/N for each level of Qashqai optimization algorithm parameters.

Fig 8 and 9 show the analysis of the results of the parameter setting Taguchi method using Minitab software, according to which MaxIt = 200 or 250, Npop = 250, C1 = 0.5 and C2 = 3 have the best performance.

3-7- Test Results and Findings

After coding and implementing the proposed meta-heuristic algorithm using the eleven well-known optimization problems listed in Table 5, the algorithm runs with specific repetitions, and the responses and computational times of the proposed algorithm are recorded and compared with genetic algorithms, particle swarm, and differential evolution.

Table 5: Famous Optimization Test Functions [16]

Function name	**Function formula**	**Function domain**	**Optimal point (Minimum Function)**	**The value of the function at the minimum point**
Sphere	$f(x) = \sum_{i=1}^{d} x_i^2$	$x_i \in [-5.12, 5.12]$	$x_i^* = (0,0,\dots,0)$	$f(x^*) = 0$
Rastrigin	$f(x) = 10d + \sum_{i=1}^{d} [x_i^2 - 10\cos(2\pi x_i)]$	$x_i \in [-5.12, 5.12]$	$x_i^* = (0,0,\dots,0)$	$f(x^*) = 0$
Rosenbrock	$f(x) = \sum_{i=1}^{d-1} [100(x_{i+1} - x_i^2)^2 - (x_i - 1)^2]$	$x_i \in [-2.048, 2.048]$	$x_i^* = (1,1,\dots,1)$	$f(x^*) = 0$
Griewank	$f(x) = \sum_{i=1}^{d} \frac{x_i^2}{4000} - \prod_{i=1}^{d} \cos(\frac{x_i}{\sqrt{i}}) + 1$	$x_i \in [-600, 600]$	$x_i^* = (0,0,\dots,0)$	$f(x^*) = 0$
Ackley	$f(x) = -a\exp\left(-b\sqrt{\frac{1}{d}\sum_{i=1}^{d} x_i^2}\right) - \exp\left(\frac{1}{d}\sqrt{\sum_{i=1}^{d} \cos(cx_i)}\right) + a$	$x_i \in [-32.768, 32.768]$	$x_i^* = (0,0,\dots,0)$	$f(x^*) = 0$
EggHolder	$f(x) = -(x_2 + 47)\sin\left(\sqrt{\left\lvert x_2 + \frac{x_1}{2} + 47\right\rvert}\right) - x_1 \sin(\sqrt{\lvert x_1 - (x_2 + 47)\rvert})$	$x_i \in [-5.12, 5.12]$	$x_i^* = (512, 404.2319)$	$f(x^*) = -959.6407$
Michalewicz	$f(x) = -\sum_{i=1}^{d} \sin(x_i)\sin^{2m}(\frac{ix_i^2}{\pi})$	$x_i \in [0, \pi]$	$d = 2; x_i^* = (2.20, 1.57)$	$f(x^*) = -1.8013$
Six-Hump Camel	$f(x) = \left(4 - 2.1x_1^2 + \frac{x_1^4}{3}\right)x_1^2 + x_1x_2 + (-4 + 4x_2^2)x_2^2$	$-5 \le xi \le 5, i = 1, 2.$	$x_1^* = (0.0898, -0.7126), x_2^* = (0.0898, -0.7126)$	$f(x^*) = 0$
Levy	$f(x) = \sin^2(\pi w_1) + \sum_{i=1}^{d-1} (w_i - 1)^2[1 + 10\sin^2(\pi w_i + 1)] + (w_d - 1)^2[1 + \sin^2(2\pi w_d)], where\ w_i = 1 + (x_i - 1)/4$	$x_i \in [-10, 10]$	$x_i^* = (1,1,\dots,1)$	$f(x^*) = 0$
Rotated Hyper-Ellipsoid	$f(x) = \sum_{i=1}^{d} \sum_{j=1}^{i} x_j^2$	$x_i \in [-65.536, 65.536]$	$x_i^* = (0,0,\dots,0)$	$f(x^*) = 0$
Shubert	$f(x) = \prod_{i=1}^{n} (\sum_{j=1}^{5} \cos(j+1)x_i + j))$	$x_i = [-10, 10]$	18 global minima	$f(x) = -186.7309$

Fig 10 shows the shapes of these famous optimization test functions.

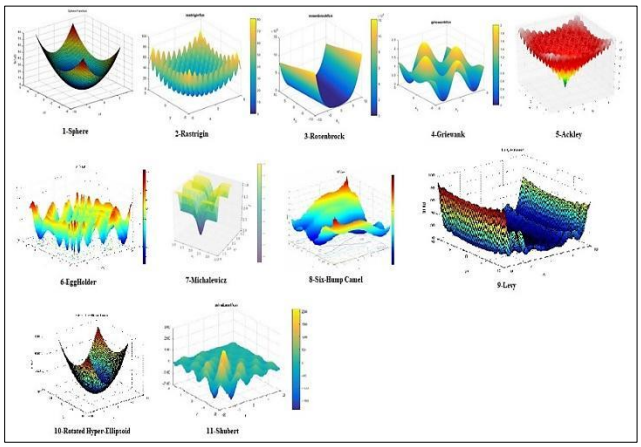


Fig. 10. Shapes of test functions

After recording the results of 30 consecutive runs of the Qashqai optimization algorithm (QOA) and genetic algorithm (GA), particle swarm optimization (PSO), and differential evolution (DE), the hypothesis test, according to Table 6, was performed to compare the cost function and speed of the algorithm.

Table 6: Testing the assumptions

TEST ASSUMPTIONS OF COST FUNCTION AND COMPUTATIONAL TIME	H_0 :THE GENETIC ALGORITHM(GA) OVERCOMES THE QASHQAI OPTIMIZATION ALGORITHM(QOA).	$H_0: \mu > \mu_0$
	H_1 :The genetic algorithm(GA) does not overcome the Qashqai optimization algorithm(QOA).	$H_1: \mu \leq \mu_0$
	H_0 :The particle swarm optimization (PSO) algorithm overcomes the Qashqai optimization algorithm(QOA).	$H_0: \mu > \mu_0$
	H_1 :The particle swarm optimization algorithm(PSO) does not overcome the Qashqai optimization algorithm(QOA).	$H_1: \mu \leq \mu_0$
	H_0 :The Differential evolution(DE) algorithm overcomes the Qashqai optimization algorithm(QOA).	$H_0: \mu > \mu_0$
	H_1 :The Differential evolution(DE) algorithm does not overcome the Qashqai optimization algorithm(QOA).	$H_1: \mu \leq \mu_0$

The Mann-Whitney test use to compare the two independent samples. Table 7 shows the results of the comparison of the two genetic and Qashqai algorithms for the Sphere function. Genetic algorithms, particle swarm, and differential evolution use to compare the results of the cost function and computational time of the proposed algorithm. A computer with the specifications of Table 8 uses to perform the calculations. The average results of 30 consecutive runs of the Qashqai optimization algorithm and the genetic algorithm and the cost function and computation time of the algorithm present in Table 9 and Table 10. It should be explained that nPop = 10, nVar = 5 and MaxIt = 100 are taken into account.

Table 7: Comparison of two genetic and Qashqai optimization algorithms (Sphere function)

Test Statistics[a]

	CostFunction
Mann-Whitney U	.000
Wilcoxon W	465.000
Z	-6.653
Asymp. Sig. (2-tailed)	.000

a. Grouping Variable: Group

Table 8: Specifications of the computer used to compare the results

System	
Processor:	Intel(R) Core(TM) i5-2450M CPU @ 2.50GHz 2.50 GHz
Installed memory (RAM):	6.00 GB (5.85 GB usable)
System type:	64-bit Operating System, x64-based processor
Pen and Touch:	No Pen or Touch Input is available for this Display

Table 9: Comparison of the average results of 30 times implementation of the Qashqai optimization algorithm (QOA) and genetic algorithm (GA) on test functions (cost function)

Function name	**Cost Function (Average)**		**P-Value**	**Standard error**	**Test result assumption**
	QOA	**GA**			
Sphere	2.01E-290	1.90E-02	0	α=0.05	H_0 is rejected
Rastrigin	0	1.94E+00	0	α=0.05	H_0 is rejected
Rosenbrock	3.98E+00	7.13E+00	0.008	α=0.05	H_0 is rejected
Griewank	0	4.53E-02	0	α=0.05	H_0 is rejected
Ackley	8.88E-16	2.58E-01	0	α=0.05	H_0 is rejected
EggHolder	8.24-E+02	-7.38E+02	0.156	α=0.05	H_0 is not rejected
Michalewicz	-1.51E+00	-1.75E+00	0	α=0.05	H_0 is rejected
Six-Hump Camel	4.77E-01	7.57E-03	0	α=0.05	H_0 is rejected
Levy	3.89E-01	1.07E-01	0	α=0.05	H_0 is rejected
Rotated Hyper-Ellipsoid	4.65E-55	1.44E+00	0	α=0.05	H_0 is rejected
Shubert	-1.37 E+02	-1.79E+02	0	α=0.05	H_0 is rejected

Table 10: Comparison of the average results of 30 times the Qashqai optimization algorithm(QOA) and Genetic algorithm on the test functions (computational time)

Function name	**Computational Time (Average)**		**P-Value**	**Standard error**	**Test result assumption**
	QOA	**GA**			
Sphere	2.87E-01	1.86E+00	٠	α=0.05	H_0 is rejected
Rastrigin	2.74E-01	1.83E+00	0	α=0.05	H_0 is rejected
Rosenbrock	2.76E-01	1.68E+00	0	α=0.05	H_0 is rejected
Griewank	2.77E-01	1.70E+00	0	α=0.05	H_0 is rejected
Ackley	2.80E-01	1.72E+00	0	α=0.05	H_0 is rejected
EggHolder	2.70E-01	2.50E+00	0	α=0.05	H_0 is rejected
Michalewicz	2.33E-01	1.78E+00	0	α=0.05	H_0 is rejected
Six-Hump Camel	2.89E-01	1.62E+00	0	α=0.05	H_0 is rejected
Levy	3.37E-01	1.57E+00	0	α=0.05	H_0 is rejected
Rotated Hyper-Ellipsoid	2.86E-01	1.60E+00	0	α=0.05	H_0 is rejected
Shubert	2.48E-01	1.58E+00	0	α=0.05	H_0 is rejected

To obtain the following tables, we continued the above procedure to compare the particle swarm algorithm (PSO) and the Qashqai optimization algorithm (QOA). The average of 30 consecutive runs of the Qashqai optimization algorithm and particle swarm algorithm and recording the algorithm's cost and computational time function are shown in Tables 11 and Tables 12.

Table 11: Comparison of the average results of 30 times implementation of the Qashqai optimization algorithm (QOA) and particle swarm algorithm (PSO) on test functions (cost function)

Function name	**Cost Function (Average)**		**P-Value**	**standard error**	**Test result assumption**
	QOA	**PSO**			
Sphere	2.01E-290	2.34E-07	0	α=0.05	H_0 is rejected
Rastrigin	0	4.41E+00	0	α=0.05	H_0 is rejected
Rosenbrock	3.98E+00	2.00E+00	0	α=0.05	H_0 is rejected
Griewank	0	2.27E-02	0	α=0.05	H_0 is rejected
Ackley	8.88E-16	5.71E-02	0	α=0.05	H_0 is rejected
EggHolder	8/24-E+02	-5.56E+02	0	α=0.05	H_0 is rejected
Michalewicz	-1.51E+00	-1.80E+00	0	α=0.05	H_0 is rejected
Six-Hump Camel	4.77E-01	4.69E-08	0	α=0.05	H_0 is rejected
Levy	3/89E-01	1.88E-01	0	α=0.05	H_0 is rejected
Rotated Hyper-Ellipsoid	4.65E-55	2.68E-06	0	α=0.05	H_0 is rejected
Shubert	-1.37 E+02	-1.87E+02	0	α=0.05	H_0 is rejected

Table 12: Comparison of the average results of 30 times the Qashqai optimization algorithm(QOA) and particle swarm algorithm (PSO) on the test functions (computational time)

Function name	**Computational Time (Average)**		**P-Value**	**standard error**	**Test result assumption**
	QOA	PSO			
Sphere	2.87E-01	2.56E-01	0	α=0.05	H_0 is rejected
Rastrigin	2.74E-01	2.99E-01	0.081	α=0.05	H_0 is not rejected
Rosenbrock	2.76E-01	2.57E-01	0	α=0.05	H_0 is rejected
Griewank	2.77E-01	2.53E-01	0	α=0.05	H_0 is rejected
Ackley	2.80E-01	2.51E-01	0	α=0.05	H_0 is rejected
EggHolder	2.70E-01	2.46E-01	0	α=0.05	H_0 is rejected
Michalewicz	2.33E-01	2.51E-01	0	α=0.05	H_0 is rejected
Six-Hump Camel	2.89E-01	2.41E-01	0	α=0.05	H_0 is rejected
Levy	3.37E-01	2.54E-01	0	α=0.05	H_0 is rejected
Rotated Hyper-Ellipsoid	2.86E-01	2.48E-01	0	α=0.05	H_0 is rejected
Shubert	2.48E-01	2.28E-02	0.69	α=0.05	H_0 is not rejected

We continued the above procedure to obtain the following tables to compare the Differential Evolution (DE) and the Qashqai optimization algorithm (QOA). The average of 30 consecutive runs of the Qashqai algorithm and Differential Evolution (DE) algorithm and recording the algorithm's cost and computational time function are shown in Table 13 and Table 14.

Table 13: Comparison of the average results of 30 times implementation of the Qashqai optimization algorithm (QOA) and Differential Evolution (DE) algorithm on test functions (cost function)

Function name	**Cost Function (Average)**		**P-Value**	**Standard error**	**Test result assumption**
	QOA	**DE**			
Sphere	2.01E-290	1.67E-04	0	α=0.05	H_0 is rejected
Rastrigin	0	9.18E-01	0	α=0.05	H_0 is rejected
Rosenbrock	3.98E+00	2.23E+00	0	α=0.05	H_0 is rejected
Griewank	0	1.28E-01	0	α=0.05	H_0 is rejected
Ackley	8.88E-16	2.44E-01	0	α=0.05	H_0 is rejected
EggHolder	-8.24E+02	-8.13E+02	0.941	α=0.05	H_0 is not rejected

Function name	Cost Function (Average)		P-Value	Standard error	Test result assumption
	QOA	DE			
Michalewicz	1.51E+00 -	-1.80E+00	0	α=0.05	H_0 is rejected
Six-Hump Camel	4.77E-01	1.15E-05	0	α=0.05	H_0 is rejected
Levy	3/89E-01	5.81E-07	0	α=0.05	H_0 is rejected
Rotated Hyper-Ellipsoid	4.65E-55	2.24E-04	0	α=0.05	H_0 is rejected
Shubert	-1.37 E+02	-1.84E+02	0	α=0.05	H_0 is rejected

Table 14: Comparison of the average results of 30 times the Qashqai optimization algorithm(QOA) and Differential Evolution (DE) algorithm on the test functions (computational time)

Function name	Computational Time (Average)		P-Value	Standard error	Test result assumption
	QOA	PSO			
Sphere	2.87E-01	2.64E-01	٠	α=0.05	H_0 is rejected
Rastrigin	2.74E-01	2.58E-01	0.001	α=0.05	H_0 is rejected
Rosenbrock	2.76E-01	2.40E-01	0	α=0.05	H_0 is rejected
Griewank	2.77E-01	2.44E-01	0	α=0.05	H_0 is rejected
Ackley	2.80E-01	3.05E-01	0	α=0.05	H_0 is rejected
EggHolder	2.70E-01	2.62E-01	0.09	α=0.05	H_0 is not rejected
Michalewicz	2.33E-01	2.37E-01	0	α=0.05	H_0 is rejected
Six-Hump Camel	2.89E-01	2.41E-01	0	α=0.05	H_0 is rejected
Levy	3.37E-01	2.57E-01	0	α=0.05	H_0 is rejected
Rotated Hyper-Ellipsoid	2.86E-01	2.38E-01	0	α=0.05	H_0 is rejected
Shubert	2.48E-01	2.40E-01	0	α=0.05	H_0 is not rejected

Table 15 summarizes the results of Tables 9 to 14.

Table 15: Summary of the results of the Qashqai optimization algorithm (QOA) and PSO and DE algorithm 11 test functions

Statistical hypothesis testing	Cost Function (Number of H0 is rejected)	Computational Time (Number of H0 is rejected)
(GA)vs (QOA)	10	11
(PSO)vs (QOA)	11	9
(DE)vs (QOA)	10	10

Therefore, we can conclude that genetic (GA), particle swarm optimization (PSO), and differential evolution (DE) algorithms cannot overcome the Qashqai optimization algorithm (QOA) in terms of optimal solution quality and computational time.Fig11 shows the performance of the Qashqai optimization algorithm (QOA) compared to the other meta-heuristic algorithms, such as genetic algorithm (GA), particle swarm optimization (PSO), and differential evolution (DE) algorithm.

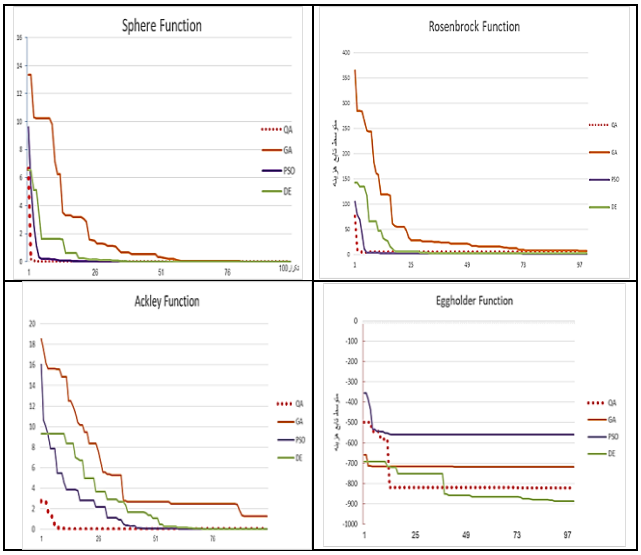


Fig. 11. Comparison of the performance results of the Qashqai Optimization Algorithm (QOA) on the test function

4- Solving the Facility Layout Problem using Qashqai Optimization Algorithm (QOA)

The facility layout problem (FLP) is one of the most critical issues in manufacturing plants. Proper layout of equipment will significantly reduce the cost of transporting the material and thus the total cost of production. Armour and Buffa first introduced this problem as a mathematical model. [12]

Researchers have always sought heuristic and meta-heuristic approaches to solving this problem, given the NP-hardness of the problem. [13, 50-54] The mathematical model of facility layout problem according to Formula Three.

$$\text{Min} \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n}\sum_{l=1}^{n} F_{ik}D_{jl}X_{ij}X_{kl}$$

Subject to

$$\sum_{i=1}^{n} X_{ij} = 1 \qquad 1 \le j \le n$$

$$\sum_{j=1}^{n} X_{ij} = 1 \qquad 1 \le i \le n$$

$$X_{ij} \in \{0,1\} \quad 1 \le i,j \le n$$

Formula Three

Table 16 represents the Indices and parameters of the facility layout problem.

Table 16: Parameters of the facility layout problem

Indices and parameters	Description
i ,k	Departments in Layout
j , l	Candidate places for departments in layout
n	Number of departments and locations
Djl	Distance between Departments l and j
Fik	The distance between the places i and k
Xij	Equals one if department i is assigned to location j and otherwise equals zero

Table 18 exhibits the results of the facility layout problem with the assumed problem information in Table 17. It should regard that nPop = 10 and MaxIt = 100 take into account.

Table 17: Specifications of the facility layout problems

The problem	Workshop length	Workshop width	Machine length	Machine width
P1	W=100	H=80	h=[12.22.17.25.25.12.25.25]	w=[18.14.11.20.24.12.24.23]
P2	W=150	H=100	h=[13.14.16.28.29.21.22.18]	w=[24.18.20.15.26.30.22.27]
P3	W=200	H=150	h=[24.22.32.35.33.18.24.25]	w=[25.22.15.20.24.15.24.28]
P4	W=220	H=180	h=[27.8.9.15.16.27.27.16]	w=[24.15.10.11.12.16.12.10]
P5	W=200	H=240	h=[23.26.19.22.27.30.25.16]	w=[15.7.20.11.12.28.25.21]

Table 18: Results of solving facility layout problem with Qashqai Optimization Algorithm (QOA) and Particle Swarm optimization (PSO)

The problem	**QOA**		**PSO**	
	Answers average	**Mean Standard deviation of responses**	**Answers average**	**Mean Standard deviation of responses**
P1	1.22E+05	1.01E+04	1.23E+05	1.23E+05
P2	1.29E+05	1.25E+04	1.29E+05	1.41E+04
P3	1.32E+05	1.46E+04	1.34E+05	1.90E+04
P4	1.01E+05	7.12E+03	1.02E+05	8.32E+03
P5	1.15E+05	7.09E+03	1.18E+05	7.24E+03

The results of Table 18 shows that the responses of the Qashqai optimization algorithm (QOA) for solving facility layout problems (FLP) are as good as the algorithm of particle swarm optimization (PSO) and have a lower standard deviation.

- ## Results and Discussion

In this article, we designed a new meta-heuristic algorithm inspired by collaborative migration, collective intelligence, and nomadic elitism. Pseudo-code of this algorithm showed in Table 3. After that, we implemented the algorithm in MATLAB software. We applied a computer with this configuration: CPU 2.5 GHz, processor core i5-2450, 6.00 GB RAM, and 64-bit operating system. Finally, the proposed algorithm was implemented 30 times using the well-known optimization functions given in Table 5.
Comparisons concerning the hypothesis tests listed in Table 6 indicate that genetic (GA), particle swarm optimization (PSO), and differential evolution (DE) algorithms cannot overcome the Qashqai optimization algorithm (QOA) in terms of optimal solution quality and computational time. (See Tables 9 to 15). Fig11 shows the performance of the Qashqai optimization algorithm (QOA) compared to the other meta-heuristic algorithms, such as genetic algorithm (GA), particle swarm optimization (PSO), and differential evolution (DE) algorithm. Moreover, the facility layout problem was solved using Qashqai Optimization Algorithm (QOA). The results show that the responses of the Qashqai optimization algorithm (QOA) for solving facility layout problems (FLP) are as good as the algorithm of particle swarm optimization (PSO) and have a lower standard deviation. (c.f Table 18)

- ## Managerial Insights and Practical Implications

This research focuses on presenting a novel metaheuristic algorithm inspired by nomads' cooperative life and migration. We proposed that this metaheuristic algorithm used low parameters and did not need gradient information. Moreover, it is elite-oriented, concatenates particular importance to swarm intelligence, and can evade local optimal points. Low parameters in an algorithm require low parameter tuning and reduce computational complexity and time. This algorithm showed excellent convergence speed to the optimal solutions. The proposed algorithm could be applicability in solving a vast scope of problems such as supply chain, factory facility layout, Flow Shop Scheduling, vehicle routing, JIT sequencing, economic load dispatch problem, assembly sequence planning, maintenance scheduling, robot path planning, cloud computing, image processing & segmentation, data clustering, feature selection, association rules mining, ANFIS training, neural network training, etc.

5- Conclusion and Outlook

Nature-inspired Meta-heuristic algorithms are increasingly introducing. The range of applications of these highly diverse, high performance, low-cost algorithms has opened a new horizon for humanity in computing and solving

complex problems in a reasonable time. In this paper, the literature review of the meta-heuristic algorithms presented from the beginning have reviewed. Then the Qashqai optimization algorithm (QOA) has been proposed to solve the optimization problems. The Qashqai algorithm does not require gradient information due to the use of random search. One of the exciting parts of the proposed algorithm is that it well simulates the behavior of nomads and can adjust the parameters C_1 and C_2 using experimental studies. These parameters considered to be the advantage of the Qashqai meta-heuristic algorithm over other meta-heuristic algorithms.

The results of the algorithm implementation for eleven well-known optimization functions showed that genetic(GA), particle swarm optimization(PSO), and differential evolution(DE) algorithms do not overcome the Qashqai optimization algorithm (QOA) both in terms of convergence to the optimal solution and computational speed.

As well, the results of the facility layout problem(FLP) using the Qashqai optimization algorithm (QOA) show that the responses of this algorithm are as good as the particle swarm algorithm and have a less standard deviation.

Some of the advantages of the Qashqai optimization algorithm (QOA) are as follows:

- Ability to search the answer space extensively.
- Ability to solve a vast span of optimization problems.
- Find the optimal global answer or the near-optimal solution.
- Crossing local optimization.
- The high-speed execution of the algorithm.
- The simplicity of the algorithm structure and implementation for solving complex optimization problems.
- The excellent convergence speed to the optimal solutions.
- The ability to evade local optimal points.
- The low standard deviation of the final solutions.
- It does not require gradient information.
- Requires few parameters to adjust.
- It is elite-oriented and concatenates particular importance to swarm intelligence.

The application of the Qashqai algorithm in various combinatorial optimization problems is one of the possible areas of research. Furthermore, it suggests to present an optimal strategy for the initial distribution of nomads and adjust the algorithm parameters for future research. For future research, suggested that the Qashqai optimization algorithm (QOA) utilized to solve various problems of continuous and discrete optimization, data clustering, feature selection problem, distribution of power generation and systems, design and optimization of communication networks, digital signals processing, and pattern recognition, design Automated and robotic systems, forecasting of economic models.

References

[1] S.-C. Chu, P.-W. Tsai, and J.-S. Pan, "Cat swarm optimization," in Pacific Rim international conference on artificial intelligence, 2006, pp. 854-858: Springer.

[2] J. van Leeuwen and J. Leeuwen, Algorithms and complexity. Elsevier, 1990.

[3] F. W. Glover and G. A. Kochenberger, Handbook of metaheuristics. Springer Science & Business Media, 2006.

[4] C. Blum, A. Roli, and M. Sampels, Hybrid metaheuristics: an emerging approach to optimization. Springer, 2008.

[5] X.-S. Yang, Engineering optimization: an introduction with metaheuristic applications. John Wiley & Sons, 2010.

[6] G. Dhiman and V. Kumar, "Spotted hyena optimizer: a novel bio-inspired based metaheuristic technique for engineering applications," Advances in Engineering Software, vol. 114, pp. 48-70, 2017.

[7] S. Dalwani and A. Agarwal, "Review on classification of nature inspired approach," International Journal of Computer & Mathematical Sciences, IJCMS, ISSN 2347, vol. 8527, 2018.

[8] A. Memari, R. Ahmad, and A. R. A. Rahim, "Journal of Soft Computing and Decision Support Systems," Journal of Soft Computing and Decision, vol. 4, no. 6, 2017.

[9] M. Birattari, L. Paquete, T. Stützle, and K. Varrentrapp, "Classification of metaheuristics and design of experiments for the analysis of components," Teknik Rapor, AIDA-01-05, 2001.

[10] A. Askarzadeh, "Bird mating optimizer: an optimization algorithm inspired by bird mating strategies," Communications in Nonlinear Science and Numerical Simulation, vol. 19, no. 4, pp. 1213-1228, 2014.

[11] A. Mohr, "Quantum computing in complexity theory and theory of computation," Carbondale, IL, vol. 194, 2014.

[12] G. C. Armour and E. S. Buffa, "A heuristic algorithm and simulation approach to relative location of facilities," Management science, vol. 9, no. 2, pp. 294-309, 1963.

[13] B. Alatas, "ACROA: artificial chemical reaction optimization algorithm for global optimization," Expert Systems with Applications, vol. 38, no. 10, pp. 13170-13180, 2011.

[14] M. Khadem, A. Toloie Eshlaghy, and K. Fathi Hafshejani, "Nature-inspired metaheuristic algorithms: literature review and presenting a novel classification," Journal of Applied Research on Industrial Engineering, 2021.

[15] M. E. Mohammad Pour Zarandi, Nonlinear optimization. Tehran University. (In Persian). https://www.adinehbook.com/gp/product/9640364754, 2013.

[16] "Virtual Library of Simulation Experiments: Test Functions and Datasets Optimization Test Problems." [Online]. Available: https://www.sfu.ca/~ssurjano/optimization.html.

[17] A. M. Fathollahi-Fard, M. Hajiaghaei-Keshteli, and R. Tavakkoli-Moghaddam, "Red deer algorithm (RDA): a new nature-inspired meta-heuristic," Soft Computing, vol. 24, no. 19, pp. 14637-14665, 2020.

[18] A. H. Kashan, R. Tavakkoli-Moghaddam, and M. Gen, "Find-Fix-Finish-Exploit-Analyze (F3EA) meta-heuristic algorithm: An effective algorithm with new evolutionary operators for global optimization," Computers & Industrial Engineering, vol. 128, pp. 192-218, 2019.

[19] A. Cheraghalipour, M. Hajiaghaei-Keshteli, and M. M. Paydar, "Tree Growth Algorithm (TGA): A novel approach for solving optimization problems," Engineering Applications of Artificial Intelligence, vol. 72, pp. 393-414, 2018.

[20]S. Mirjalili and A. Lewis, "The whale optimization algorithm," Advances in engineering software, vol. 95, pp. 51-67, 2016.
[21] S. Mirjalili, "Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems," Neural computing and applications, vol. 27, no. 4, pp. 1053-1073, 2016.
[22] J. B. Odili, M. N. M. Kahar, and S. Anwar, "African buffalo optimization: a swarm-intelligence technique," Procedia Computer Science, vol. 76, pp. 443-448, 2015.
[23] S. Mirjalili, "The ant lion optimizer," Advances in engineering software, vol. 83, pp. 80-98, 2015.
[24] B. Javidy, A. Hatamlou, and S. Mirjalili, "Ions motion algorithm for solving optimization problems," Applied Soft Computing, vol. 32, pp. 72-79, 2015.
[25] G.-G. Wang, X. Zhao, and S. Deb, "A novel monarch butterfly optimization with greedy strategy and self-adaptive," in 2015 Second International Conference on Soft Computing and Machine Intelligence (ISCMI), 2015, pp. 45-50: IEEE.
[26] M. T. Adham and P. J. Bentley, "An artificial ecosystem algorithm applied to static and dynamic travelling salesman problems," in 2014 IEEE International Conference on Evolvable Systems, 2014, pp. 149-156: IEEE.
[27] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," Advances in engineering software, vol. 69, pp. 46-61, 2014.
[28] M. Hajiaghaei-Keshteli and M. Aminnayeri, "Solving the integrated scheduling of production and rail transportation problem by Keshtel algorithm," Applied Soft Computing, vol. 25, pp. 184-203, 2014.
[29] A. Hatamlou, "Black hole: A new heuristic optimization approach for data clustering," Information sciences, vol. 222, pp. 175-184, 2013.
[30] E. Cuevas, D. Oliva, D. Zaldivar, M. Pérez-Cisneros, and H. Sossa, "Circle detection using electro-magnetism optimization," Information Sciences, vol. 182, no. 1, pp. 40-55, 2012.
[31]X.-S. Yang, "Flower pollination algorithm for global optimization," in International conference on unconventional computing and natural computation, 2012, pp. 240-249: Springer.
[32]A. H. Gandomi and A. H. Alavi, "Krill herd: a new bio-inspired optimization algorithm," Communications in nonlinear science and numerical simulation, vol. 17, no. 12, pp. 4831-4845, 2012.
[33] X.-S. Yang, "A new metaheuristic bat-inspired algorithm," in Nature inspired cooperative strategies for optimization (NICSO 2010): Springer, 2010, pp. 65-74.
[34] X.-S. Yang and S. Deb, "Cuckoo search via Lévy flights," in 2009 World congress on nature & biologically inspired computing (NaBIC), 2009, pp. 210-214: Ieee.
[35] X.-S. Yang and S. Deb, "Eagle strategy using Lévy walk and firefly algorithms for stochastic optimization," in Nature inspired cooperative strategies for optimization (NICSO 2010): Springer, 2010, pp. 101-111.
[36] E. Atashpaz-Gargari and C. Lucas, "Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition," in 2007 IEEE congress on evolutionary computation, 2007, pp. 4661-4667: Ieee.
[37] B. Webster and P. J. Bernhard, "A local search optimization algorithm based on natural principles of gravitation," 2003.
[38] M. M. Eusuff and K. E. Lansey, "Water distribution network design using the shuffled frog leaping algorithm," in Journal of Water Resources planning and management, , vol. 129, pp. 210-225, 2003.
[39] H. A. Abbass, "MBO: Marriage in honey bees optimization-A haplometrosis polygynous swarming approach," in Proceedings of the 2001 congress on evolutionary computation (IEEE Cat. No. 01TH8546), 2001, vol. 1, pp. 207-214: IEEE.
[40] R. Storn and K. Price, "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," Journal of global optimization, vol. 11, no. 4, pp. 341-359, 1997.
[41] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in MHS'95. Proceedings of the sixth international symposium on micro machine and human science, 1995, pp. 39-43: Ieee.
[42] R. G. Reynolds, "An introduction to cultural algorithms," in Proceedings of the third annual conference on evolutionary programming, 1994, vol. 24, pp. 131-139: World Scientific.
[43] D. E. Goldberg, "Genetic algorithms in search, optimization, and machine learning. Addison," Reading, 1989.
[44] F. Glover, "Future paths for integer programming and links to artificial intelligence," Computers & operations research, vol. 13, no. 5, pp. 533-549, 1986.
[45] M. A. Hematalikeikha and M. Alinaghizadeh, "Educational and practical approach to the study of native architecture-case study: Study of Qashqai tribe housing as one example of a sustainable native culture of Iran," Procedia-Social and Behavioral Sciences, vol. 51, pp. 373-379, 2012.
[46] M. GHARAKHLOU, "A study of cultural changes among the Qashqai tribes in Iran," 2006.
[47] M. Yazdanpanah and M. Rostami, "Who Are the Qashqai People?."
[48] P. Oberling, "The Qashqā'i Nomads of Fārs," in The Qashqā'i Nomads of Fārs: De Gruyter Mouton, 2017.
[49] S. Baluja and R. Caruana, "Removing the genetics from the standard genetic algorithm," in Machine Learning Proceedings 1995: Elsevier, 1995, pp. 38-46.
[50] G. L. Cravo and A. R. S. Amaral, "A GRASP algorithm for solving large-scale single row facility layout problems," Computers & Operations Research, vol. 106, pp. 49-61, 2019.
[51] S. H. A. Rahmati, V. Hajipour, and S. T. A. Niaki, "A soft-computing Pareto-based meta-heuristic algorithm for a multi-objective multi-server facility location problem," Applied soft computing, vol. 13, no. 4, pp. 1728-1740, 2013.
[52] A. Drira, H. Pierreval, and S. Hajri-Gabouj, "Facility layout problems: A literature analysis," IFAC Proceedings Volumes, vol. 39, no. 3, pp. 389-400, 2006.
[53] H. M. Dbouk, K. Ghorayeb, H. Kassem, H. Hayek, R. Torrens, and O. Wells, "Facility placement layout optimization," Journal of Petroleum Science and Engineering, vol. 207, p. 109079, 2021.
[54] X. Zhun, X. Liyun, and L. Xufeng, "An Improved Pigeon-inspired Optimization Algorithm for Solving Dynamic Facility Layout Problem with Uncertain Demand," Procedia CIRP, vol. 104, pp. 1203-1208, 2021.
[55] J. Torres, "QASHQAI PEOPLE: MEETING AUTHENTIC NOMADS OF IRAN", https://againstthecompass.com/en/qashqai-people-iranian-nomads (Last updated on Aug. 25, 2022).
[56] "Iran Nomads Tour Living with the Qashqai Tribes," https://surfiran.com/iran-tour/iran-nomad-living-qashqai-tribes. (accessed Nov. 7, 2020).
[57] "About Qashqai Nomads", https://surfiran.com/iran-tour/iran-nomad-living-qashqai-tribes. (accessed Oct. 2, 2021).

Journal of Information Systems and Telecommunication
Vol.11, No.2, April-June 2023, 163-175
http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

An analysis of Covid-19 pandemic outbreak on Economy using Neural Network and Random Forest

Md. Nahid Hasan[1*], Tanvir Ahmed[1], Md. Ashik[1], Md. Jahid Hasan[1], Tahaziba Azmin[1], Jia Uddin[2*]

[1].Department of Computer Science and Engineering, Brac University, 66 Mohakhali, Dhaka-1212
[2].Artifical Intelligence and Big Data Department, Endicott College, Woosong University, Daejeon, South Korea

Received: 23 Feb 2022 / Revised: 15 Jun 2022 / Accepted: 19 Jul 2022

Abstract

The pandemic disease outbreaks are causing a significant financial crisis affecting the worldwide economy. Machine learning techniques are urgently required to detect, predict and analyze the economy for early economic planning and growth. Consequently, in this paper, we use machine learning classifiers and regressors to construct an early warning model to tackle economic recession due to the cause of covid-19 pandemic outbreak. A publicly available database created by the National Bureau of Economic Research (NBER) is used to validate the model, which contains information about national revenue, employment rate, and workers' earnings of the USA over 239 days (1 January 2020 to 12 May 2020). Different techniques such as missing value imputation, k-fold cross validation have been used to pre-process the dataset. Machine learning classifiers- Multi-layer Perceptron- Neural Network (MLP-NN) and Random Forest (RF) have been used to predict recession. Additionally, machine learning regressors-Long Short-Term Memory (LSTM) and Random Forest (RF) have been used to detect how much recession a country is facing as a result of positive test cases of covid-19 pandemic. Experimental results demonstrate that the MLP-NN and RF classifiers have exhibited average 88.33% and 85% of recession (where 95%, 81%, 89% and 85%, 81%, 89% for revenue, employment rate and workers earnings, respectively) and average 90.67% and 93.67% of prediction accuracy for LSTM and RF regressors (where 92%, 90%, 90%, and 95%, 93%, 93% respectively).

Keywords: Multi-Layer Perceptron (MLP); Long Short-Term Memory (LSTM), Random Forest, Economic Recession, Machine learning (ML), Covid-19.

1- Introduction

An economic forecast directly affects the financial institutions during a pandemic outbreak caused by contagious organisms. Any wrong decision taken during this time may have significant adverse effects. As a result, detection of different economic sectors plays a vital role in finance. Over the decades, multiple pandemic outbreaks have taken place. For instance, the Spanish Flu (1918-1919), which is estimated to have infected and killed millions of people causing a severe impact in economic sectors. Asian Flu (1957-1958) which has also killed millions of people worldwide, depleting the economy across the globe. SARS coronavirus (2003) caused respiratory illness and killed 774 people creating a huge negative impact on human health as well as the economy. Swine Flu (2009) caused the death of 150000 to 575000 people, depleting the stock markets, tourism, food as well as transportation industries incurring a considerable loss. And lastly the novel Coronavirus (COVID-19) (2019-present) causing the death of 6.3 million people till now and destroying the economy throughout the world [1]. Significant numbers of countries have already gone through a financial crisis after a pandemic outbreak. For example, the SARS outbreak hit many Asian countries' economies and took millions of lives [2]. The economic costs from a global disease go beyond the direct damages incurred in disease-inflicted countries' affected sectors. The disease spreads rapidly through countries across networks linked to worldwide travel. Any economic shock to one country spreads quickly to other countries through expanded trade and financial ties related to globalization. Infectious diseases are likely to increase the global cost. As a result, it is essential to take the early initiative to

✉ **Jia Uddin**
jia.uddin@wsu.ac.kr

revive the economy. This paper provides a method for predicting the economy during a pandemic outbreak. In particular, the techniques that have been used are examined, the experiments that have been conducted are reviewed, and directions of future work from the perspective of machine learning are considered [3]. This model will capture crucial linkages across different economic sectors, thereby comprehensively measuring disease-related costs.

Millions of livelihoods around the globe are affected by the economic instability created by the pandemic. Everyone fights a longstanding battle against microorganism, which causes the pandemic and affects people's livelihood throughout the world. However, such pandemics are a problem of an entirely different nature and demand an unparalleled response scale. To respond to the imminent challenges to the long-term impact on the country's economy, machine learning techniques are used. Machine learning techniques can detect economic crises which help the public and private sector leaders, as well as the policymakers, plan a better solution to combat the financial crisis. Previously, Keogh-Brown et al. described that due to SARS virus outbreak 3.7% loss in GDP (US$ billion), 23.1% loss in export and trade (US$ billion), and 0.86% loss in tourism (US$ billion) sector in Hong-Kong was recorded [4]. Canada loses 3.2-6.4% in GDP (US$ billion), 1% in Growth (US$ billion), 5.2% in export and trade (US$ billion), 0.03% in tourism (US$ billion) and 6.25% in Airline (US$ billion) [4].

On the other hand, in 2020, in the USA, due to the Covid-19 outbreak, different economic sectors were affected. Those sectors are affected by increasing covid-19 positive test cases daily, which causes a national economic recession. Earnings received by workers will decrease by increasing positive test cases compared with January 2020. Falling workers' wages suggests recession which causes unemployment and downward pressure on wages. Manufacturing sector workers earnings decreased 41%, Retail Trade workers' earnings decreased 36%, Transportation and Warehousing decreased 30% and Health Care, and Social Assistance decreased 31%.

Employment level is also affected by this pandemic situation. Employment level or employment rate is defined as the number of people engaged in productive economic activities. Manufacturing employees' level decreased 41%, Retail Trade employees' level decreased 38%, Transportation and Warehousing decreased 32% and Health Care, and Social Assistance decreased 31% compared with January 2020.

These sector-wise downturns of employment rate and worker wages reflect the USA's national economic recession. Moreover, net Revenue for all businesses is also decreasing simultaneously. The maximum employment rate was 1% at the end of January 2020. Still, with the increase of positive test cases, it gradually decreased and within the first week of May 2020, the rate decreased 39% compared with January 2020. Also, workers' wages decreased 38% within the first week of May 2020. However, revenue for all small businesses started increasing from the last week of January 2020, and the maximum value was 14%. Suddenly, it started decreasing, and at the end of March 2020, revenue decreased 49% compared with January 2020.

The paper aims at defining robust financial crisis predictors. A link between a covid-19 pandemic and its impact on people's revenue, earnings, and employment is found. The chosen data-set consisting of different sectors revenue, earnings, and employment helped to find how a pandemic affects the economy. In comparison, this paper's uniqueness lies in the variables commonly correlated with the novel covid-19 pandemic. Overall, this paper contributes to decision-makers and leaders detecting the economic crisis during or after a pandemic and introducing measures to eliminate or dampen a crisis entirely.

2- Literature Review

A number of researches have been conducted due to the rise and fall of the economy during the pandemic. These studies relate how a pandemic can influence a country's economy, culture, people and others.

Smith et al. has shown the UK's economy's impact based on the Computable General Equilibrium (CGE) model [5]. This research estimates that only pandemic influenza can minimize GDP by 0.3%, 0.4%, and 0.6% for mild, moderate, and severe cases, respectively. Additionally, losses of different sectors are 1.5% in domestic output, 2% in household consumption, 3% in exports, 2.5% in imports, 2% in government consumption as calculated from data-set 2003 supply chain in the UK. This paper presented that a large portion of the economy was damaged due to the pandemic.

Keogh-Brown et al. suggest macro-economic outcome after SARS outbreak, through affected countries and their economic sectors evaluated by economic indicators [4]. Researchers calculated losses of various sectors such as GDP, growth, exports and trade, tourism, food and travel. In Hong-Kong, 3.7% loss in GDP, 4.75% loss in Growth, 23.1% loss in export and trade, and 0.86% loss in tourism is recorded according to their databases. This research gives a clear picture of affected sectors of individual countries and why stakeholders could develop solutions to be aware of the next pandemic outbreak.

Fernandez-Delgado et al. compared several classification algorithms to predict the economic crisis [6]. The results showed that multiple models demonstrated different detection results. Hence, deciding which one to be selected was a challenge. Lastly, the random forest family

algorithm showed the best results in the early warning of the economic crisis.

In recent years, bankruptcy forecasts have been made by machine-learning using standard statistical approaches [7]. To obtain improved failure–detection solutions of problems, analysis has been done using the mathematical and machine learning techniques [8]. In particular, the datasets, financial partnerships, country of origin, and the timeline of the analysis were used, and the results and implementation were compared with several different detection accuracy techniques [3-8].

This paper aims to detect and forecast the financial status of institutions or individuals by using machine learning algorithms [3]. If this can be done, human sufferings can be reduced by taking the best decision and the financial crisis can be overcome. Here, tools such as neural networks, decision trees, etc., have been extensively studied to predict financial crises. Use MLP, Random Forest (RF) [36] to apply machine learning methods, such as pattern classification techniques, single classification techniques and soft classification techniques. We have seen some problems that are not widely discussed in the literature in the observation. When different datasets are used for different training and evaluation sets and cross-validation, more accurate results can be obtained.

It is difficult to make crucial decisions due to the vulnerability of the emerging coronavirus outbreak. The deep learning and fuzzy detection approach are proposed by Fong et al. on different outcomes of Coronavirus and its effect [9]. The current events and possible actions were described using the Composite Monte-Carlo simulation system. Fong et al. addressed the daunting challenges of predicting the fate of an outbreak correctly effectively using the availability of dataset, the layout for picking the best predictive model and finely tuning each model's parameter [9].

The statistical data were benchmarked in a paper by Bluwstein et al. (2020), used in many machine learning models, such as decision trees, random forests, large-scale random trees, SVM [38] and artificial neural networks [10]. The finding of the paper, except for human decision-making agencies, all machine learning models performed better than logistic regression [10]. The best performing machine learning model (a tree that is too random) can correctly predict the global financial crisis of 2007-2008 and provide outstanding signals in countries and regions with different economic realities and achievements [10].

In [11], Nyman gives an actual forecasting scenario where they use a small number of financial variables for the detection of the economic recession. Two estimation techniques- ordinary least squares regression and random forest machine learning were utilized. Random forests noisy, non-linear, high-dimensional detection can be tackled using machine-learning models [12-13]. The author obtains qualitatively similar results for the UK and USA through the random forest algorithm's predictive power is more efficient for the USA. Finally, the author says the machine learning approach is very efficient for forecasting horizons and providing better information.

Car, Z. et al. [14] built a model that can detect the spread of covid-19 infection to predict its impact. They used a 51- day data set to train a multi-layer perceptron (MLP) neural network, and chose to use a grid search algorithm for hyperparameter optimization. After performing k-fold cross-validation, they found that the accuracy of positive confirmed cases was 94%, the accuracy of recovered patients was 78%, and the accuracy of deceased patients was 98%. This is an excellent model for the detection of covid-19 infection.

To date, many models have been developed with different domain knowledge to predict the economic crisis. Logistic regression (Ohlson 1980) and factor analysis are some of the conventional statistical approaches (West 1985) [15-16]. Some of Artificial Intelligence methods such as artificial neural network ANN (Atiya 2001), Support vector machine (SVM) (Min and Lee 2005) (Shin et al. 2005), Bayesian networks (Sarkar and Sriram 2001) (Sun and Shenoy 2007) and many integrated machine learning techniques (Fedorova) et al. 2013) (Abellán and Mantas, 2014), several hybrid methods are widely used to predict economic crises [17-22].

3- Proposed Methodology

This section discusses constructing the models to determine the impact of economic indicators in a country due to a pandemic. Firstly, we start with a description describing where we have collected datasets, structure, etc. Then, how we process the data-set to fit into the models. Then, the implementation of algorithms for detecting recession due to this covid-19 pandemic. Figure 1 presents detailed architecture of the proposed model.

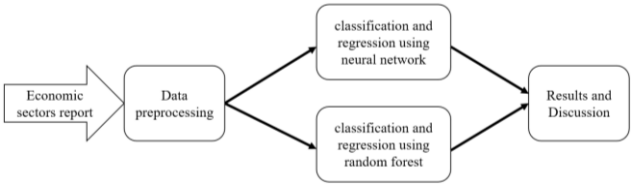


Fig. 1 Architecture of proposed model

3-1- Data Description

The Opportunity Insights Team built a publicly available database [23]. The database tracks economic activity at a granular level in real-time using anonymized data from private companies. They build the database using private sector data to show the Economic Impacts of COVID-19.

They studied how COVID-19 affects the economy by examining heterogeneity in its influences. The data-set used in this research is collected from a publicly available archive, operated by the Opportunity Insights Economic Tracker. The archive contains data from leading private companies from credit card providers to payroll firms to offer a real-time image of indicators such as employment rates, consumer spending, and job postings across counties, industries, and income groups. The data-set is split into employment rate, worker's earnings, and national revenue of various economic sectors. During this research, the dataset contained the data of 239 days for COVID-19 positive cases, small business revenue, and employment rates among low-income workers. Moreover, the chosen data-set, as published, is organized as time-series data. Besides, this dataset contains data points of 239 days, which is split into five folds.

3-2- Data Preprocessing

The dataset has three different sorts of information: employment rate, worker's earnings, and national revenue of various economic sectors. Firstly, the dataset was time-series data, so we converted it to fit the model. As we are classifying, is there any recession due to the impact of covid-19? If any value from the employment rate of all sectors is negative, then there exists a recession; otherwise, not. For this purpose, we are adding one extra feature into the database depending on the overall employment rate, which will be the attribute column for classifying recession in employment rate and the same for others.

In the real-world data-set, missing data points are not unexpected. In this paper, used data-sets even have missing data points. We used statistical imputation to fill up the gaps of missing data points to reduce biases. Although there are varieties of techniques for imputation, we have used mean values for each feature. Replacing missing data points with mean is simple; besides, it does not introduce many biases in the data.

$$Mean = \frac{x_1+x_2+x_3+\cdots+x_n}{n} \quad (1)$$

Each data set is divided into five random folds used to train and validate the model. Here, folding is used to train the model (k-1) times, and the rest will be used to verify the model, where k = 5. This process continued unless each fold was used as a test set for the model, which gave us the advantage of having a generalized model. In addition, because the data points are limited; therefore, this K-fold cross-validation technique has helped us use each data point as a training set, and a test set [24]. To determine the cross-validation result, we calculate the average of the R^2 scores.

$$R^2 = \frac{1}{5}\sum_{n=1}^{5} R_k^2 \quad (2)$$

3-3- Implementing Classifier in Economic Recession Detection Model

This section discusses how we detect economic recession due to pandemic outbreaks. We first use MLP (multilayer perceptron) to detect the issue first and then Random Forest classifier. Both algorithms help us determine the economic recession a country might face due to this covid-19 effect.

3-3-1 Multi-layer Perceptron Classifier

In Artificial Neural Network, Multi-layer Perceptron (MLP) is a fully connected feed-forward neural network that mimics the human brain to build up a machine learning model. Multi-layer Perception (MLP) is a class of Feed-forward Artificial Neural Network (ANN). MLP has three kinds of a layer, such as input layer *(i)*, which has an equal number of neurons, same as data-set features, hidden layer $j_0\ to\ j_n$ and output layer *(k)* consist of a single neuron. Signals flow from left to right layer by layer for computing output of each neuron, which is named as Forward Propagation of Function Signals [25]. Again, to minimize errors of the network, an error signal in the output neuron propagates backward layer by layer referred to as Backward Propagation of Error Signal.

To generate a network that can produce a higher accurate result, choosing hyper-parameters is a crucial factor. For the different values of hyper-parameters, the network will show different performance. For instance, if a hyper-parameter like the number of iterations is excessive then the network might face an over-fitting problem. Additionally, depending on the learning rate, the network will converge slowly or quickly. We applied the Grid Search Algorithm to find best-fitted hyper-parameters such as weights, learning rate, number of hidden layers, etc., for network convergence [26]. Besides, we set a few hyper-parameters constant such as stochastic gradient descent, which is a solver for weight optimization, a maximum number of iterations is 100, activation function *(relu)*, an initial learning rate 0.1. However, Summation of inputs and connected links weights will pass through *relu* activation function. *Rectified Linear Unit (relu)* function shows better convergence performance and is computationally efficient since it maps only the value *max (0, z)* where z is $z>0$. This function is also differentiable. Hence, the total equation of output from the *i-th* neuron is,

$$O(j) = relu\left(\sum_i^n w_{j_i}\, y_i + b_i\right) \quad (3)$$

Here, *relu* is the activation function, W_{ji} is the weights of connected links, y_i are inputs, and b_i is the bias in this

case it is 1. This process of computing output is the same for all hidden layer and output layer neurons.
An activation function decides whether it will fire the output of that particular neuron or not. Now, η is the learning rate, which decides how quickly the network will converge and we put $\eta=0.1$ so that machine can learn slowly and perform with higher accuracy in the long run. Hence updated *Delta rule*,

$$w_{new} = w_{old} + (0.1 * \delta_k * y_j) \quad (4)$$

Where δ_k is Local Gradient, and y_j is the input of *k-th* neuron or output from *j-th* neuron, w_{old} is the previous iteration's weight. After that, equations for calculating δ is different for hidden and output layers. If *k-th* is a neuron of the output layer,

$$\delta_k = \emptyset'(\sum_k w_{kj} y_i)(\partial_k - y_k) \quad (5)$$

Here, δ_k is the desired output from the *k-th* neuron, and ϕ' is the first derivative of the activation function. Again δ_j for *j-th* neuron of a hidden layer,

$$\delta_j = \emptyset'(\sum_j w_{ji} y_i)(\sum_{c \in k} \delta_c w_{cj}) \quad (6)$$

Where *c* is the set of next hidden/output layer neurons and y_i is a set of input layer neurons or hidden layer neurons, but it is the previous layer of *j-th* neuron's layer. In this way, MLP model training will continue until it meets stopping criteria. When the Average square error (loss function) change is sufficiently small for per epoch, then the training process will stop. Another measure is, after each period, the MLP model will be tested for Generalization, and if this generalization performance is suitable only then, training will stop.
After putting the values mentioned above, we have got a satisfactory result from the network. However, to check this network's robustness, we have applied the k-fold *(k=5)* cross-validation technique. After applying this technique, some of the results dropped, and some showed more accuracy. Nevertheless, this difference is shallow, so this network fulfilled our exception for detecting the issue that we wanted to solve. Finally, this is a simplex network having the ability to solve a complex problem like detecting financial crisis with lower training time.

3-3-2 Random Forest Classifier

The random forest classifier is a collection of projected trees, where each tree is subordinate to independently evaluated random vectors, with comparable transport within the random forest with one another tree. It, too accomplishes the proper speed required and productive parameterization within the process. The random forest classifier bootstraps random tests where the expectation with the most elevated vote from all trees is chosen. The distinction of each tree is ensured due to the taking after qualities. To begin with, each tree training within the test employs random subsets from the beginning training tests. Besides, the ideal part is chosen from the unpruned tree nodes' arbitrarily chosen features. Thirdly, each tree develops without limits and ought not to be pruned at all. In this model, we have implemented a random forest classifier for classification that uses an ensemble learning approach to detection [27], which uses several decision trees during the training process and average individual tree detection outputs. Random Forest efficiently runs on massive data-sets, can handle thousands of input variables without variable deletion, produces significant variable for forecasts, creates an internal unbiased measure of generalization error as forest growth increases, has an adequate method to estimate lost data, and maintains accuracy where a large proportion of data is lacking [12,28]. The chosen RFC's key objective is the power of the individual decision tree and the relation between base trees [12]. Random forest classifier comprises various individual classification trees, where each tree may be a classifier given diverse weighted classification. The output of the classification determined the overall classification. It builds each tree by part number of features for each part without pruning [29].

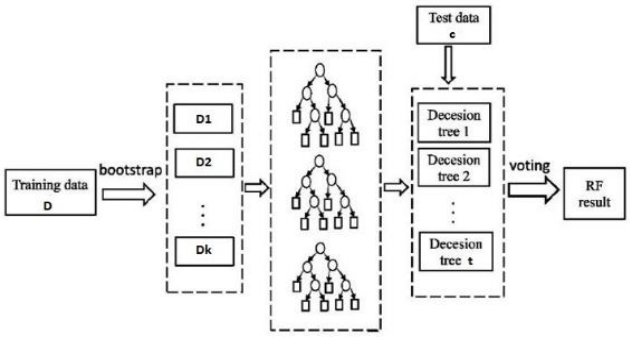


Fig. 2 Random Forest model

The classifier of Random Forest functions as consequently [30]: we choose C subsets from which to construct the training set T and initial training data D using bootstrap sampling

$$T = D_1, D_2, D_3, \dots D_c \quad (7)$$

The algorithm automatically generates the decision tree for C models with a random vector θc for each subset:

$$M = M_1, M_2, M_3, \dots M_c \quad (8)$$

The random vectors θc, $c = 1,2.....C$ are both distinct and distributed identically. Each decision tree evolves freely

without pruning so that all the trees are combined to get a forest.

$$r(T,c), c = 1,2,3 \dots C \quad (9)$$

For determining the classification of a new input variable t, the final vote of each established decision tree model is accurate. The result for classification is:

$$R(t) = \sum_{i=1}^{c} I(ri(t) = Z) \quad (10)$$

Where R(t) denotes the product of the classification is; ri(t) denotes the outcome of the decision tree classification; Z is the target group, and I *(ri(t)=Z)* stands for the characteristic function.

n-estimators are the number of trees in the forest. Having a few distinctive forms of trees with different depths and sizes boosts the generalization of the n-estimators' trees that the algorithm needs to construct. If the estimator is n, it gives n other decision trees. Using n-estimators, we build several trees within a Random Forest before aggregating the detection. We want to make the computational expense while taking the trees as it delays code to run.

Using the criterion Gini-impurity, we determine the feature of a tree that has to split the parameters. It measures the quality of each split. We use a max-depth to see how much further the tree has to be expanded to each node until we get to the leaf node. We set the max-depth to run the option for the risks of over-fitting this model. We observe the impact of the max-features hyper-parameter. We realize that the random forest selects a few random tests to explore the main break from the functions. We can see that the execution of the first increments demonstrates that the number of max features increases.

Nonetheless, the training score keeps expanding after a certain point. However, the test score saturates and begins diminishing the conclusion, implying that the show starts to over-fit. Ideally, the general execution of the demonstration near six of the highest highlights is the most notable. In general, the ideal number of total features tends to be similar to this value. To decide whether the algorithm will avoid further splitting, we use min samples to define the minimum number of records present in each node. If the split number is less than n, there will be no further split. We use max-leaf nodes to expand the tree in a best-first manner resulting in a relative reduction in impurity. We use max-leaf nodes to grow the tree in a best-first way resulting in a relative decrease in impurity. The random state makes it simple for others to imitate results if given the same training data and parameters. For sampling data points, we use the bootstrap process.

The algorithm randomly selects many rows with replacement to construct the trees using bootstraps once we provide the Random Forest Classifier model's training data. If the bootstrap option is set to False, there will be no random sampling, and the entire data set will be used to build the trees. We use oob-score as it is much quicker because it gathers all of the trees' observations and finds the highest score on each observation base's trees that did not use that observation to train. Oob-score is a cross-validation technique similar to a leave-one-out validation technique in which a model's generalized approximate output is trained on n-1 data samples. We set n-jobs to -1 will often lead to faster processing. If we use -1, there is no limit on how much computing resources the code can use n-jobs helps the program know how many processors it can use. The default value of 1 means that only one processor can be used. We set the logging output to be verbose, which gives us continuous feedback on what the model is doing as it is processed. This parameter defines the verbosity of the construction method of the tree. We use false Warm Start for recursive feature collection, and false Warm Start suggests that other features will gain in value as we drop such features, and it will be repeated used. It is often used in regression models with backwards exclusion and is not often used in classification models.

3-4- Implementing Regression in Economic Recession Detection Model

This section covers how much economic recession a country might face due to a pandemic outbreak in an individual economic sector. First, we will discuss how RNN implementation helped then Random Forest regressor.

3-4-1 Long Term Short-Term Memory

In this model, we used the long-term short-term memory (LSTM) method to estimate the economic impact of the COVID-19 outbreak on different regions of the United States [31,37]. When dealing with time series, we used LSTM modelling, an in-depth learning method that is useful when trying to model time series. In the LSTM unit, there are four functions in this model, which are 3 *Sigmoids* (f, i and o below) and *Tanh* (c function below). It is mentioned that the coefficient of deviation may be a general feature of all functions in the learning model, which can be set or measured in advance during the training process. In order to help it adjust to the various situations of each case, the bias can be used to model calibration. A common LSTM cell is formed of three gates-*input gate, output gate, and forget gate* [32].

In the LSTM structure, the first layer is called the forget gate, which selects the information to forget. It can produce any value between 0 (completely forget) to 1 (usefully) [33].

$$f_t = \sigma(W_f(h_{t-1}, x_t) + b_f) \quad (11)$$

The next step of the algorithm is to determine the new input which needs to be added.

$$i_t = \sigma(W_i(h_{t-1}, x_t) + b_i) \quad (12)$$

After that, the algorithm determines the new candidate value of neural cells.

$$\check{C} = \tanh(W_c(h_{t-1}, x_t) + b_c) \quad (13)$$

After computing above equations, it states the new cell states by computing C_t.

$$c_t = f_t \times C_{t-1} + i_t \times \check{C}_t \quad (14)$$

Lastly, the output function will predict the value. This layer is called detection layer and the detection will then

$$o_t = \sigma(W_0(h_{t-1}, x_t) + b_0) \quad (15)$$

$$h_t = o_t \times \tanh(C_t) \quad (16)$$

We add the LSTM layer and later integrate a few Dropout layers to prevent overfitting. We integrate the LSTM layer with 50 units which is the dimensionality of the output space. We define the dropout layers 0.2. This indicates that it will reduce the number of layers by 20%. After that, we merge the dense layer, which determines the output of 1 unit. Next, we used a very common optimizer, the Adam optimizer. Then, we fit the LSTM model to run at 100 epochs and 32 batch sizes.

We will conduct a k-fold on the data to improve results' complexity. K -fold is a cross-validation technique that requires the creation of various models on subsets of the data set. This approach would be very beneficial in achieving the desired predictive precision standard.

3-4-2 Random Forest Regressor

The random forest regression (RFR) incorporates a vast collection of decision trees trained together to produce a more precise and reliable final forecast [12]. A regression tree is ordered from the leaf to the root node based on some parameters. RFR consists of a supervised learning algorithm for predicting output target feature average by bootstrap aggregation or bagging of independently built decision trees [34]. Bootstrap aggregation or bagging is used for lowering variance error sources of independently built decision trees. We first randomly select points from the training data set of employment rate, worker income and revenue of different sectors. Then we created a bootstrap sample of the random data with replacement and created the root node, eventually forming a decision tree. After that, we created 200 different decision trees from it. To estimate an economic recession, each decision tree forecasts a data point value for worker income, employment rate and revenue and finally assigns the average new data point over all the expected values. Forest-random regression trees use a greedy top-down method to define ideal recursive divisions in binary nodes.

$$Min(SSE) = (\sum_{i=1}^{n}(y_i - y_s)^2) \quad (17)$$

Here SSE=Sum Squared error, y_t = output target feature data, and y_s = terminal node output target feature mean
And y_s is calculated by,

$$y_s = \frac{1}{m}\sum_{i=1}^{m}(y_t) \quad (18)$$

Here y_s is the mean terminal node, m=number of observations in the terminal node, and y_t output target feature data.

In a Random Forest regression algorithm, tree bagging consists of predicting the output feature of an independently built decision tree by calculating the arithmetic mean,

$$y_p = \frac{1}{k}\sum_{i=1}^{k}(y_s) \quad (19)$$

Here y_p = mean output target feature detection, *k*=number of independently built decision trees, and y_s =independently built decision trees output feature detection.

Unlike other machine learning techniques, Random Forest regressor only needs to set two parameters to construct a detection model: the number of regression trees and the random state, which is 200 regression trees and the random state is set at 0. For accuracy, we calculated the mean absolute error, which gives the average of error in detection and mean squared and root mean squared error to check how close the detection is to the actual value. Lastly, we cross-validation to check how well random forest regression performs on test data.

4- Result and Discussion

In this section, we discuss the achieved results in detail by following the described methodology. We will also analyze essential factors for achieving these results.

4-1- Result of Classifiers

We choose to use two different classifiers to detect a country's recession due to a pandemic. Now we will discuss results found from multi-layer perceptron and random forest algorithm described in section -4.1. The formulas for calculating Table 1 values for MLP and Random Forest classifiers are equations no. 20, 21, and 22.

$$Accuracy = \frac{Correctly\ predicted\ class * 100\%}{Total\ testing\ class} \quad (20)$$

$$Sensitivity = \frac{TP}{TP+FP} \quad (21)$$

In equation 21, *TP = true positive* and *FN = false negative.*

$$Specificity = \frac{TN}{TN+FP} \quad (22)$$

In equation 22, *TN = true negative* and *FP = false positive.*

4-1-1 Multi-layer Perceptron

The MLP classifier demonstrated an accuracy of 81% for the employment level, 89% for worker earnings, and 95% for national revenue as shown in Table 1. To achieve these results, we have used the initial learning rate (alpha) 0.1, *relu* activation function, three hidden layers with different units of neurons, and others.

Table 1: Results from MLP classifier (CV=cross validation

Section	Accuracy	Sensitivity	Specificity	Accuracy (cv)
Employment level	81%	100%	0%	82%
Worker Earnings	89%	100%	0%	88%
National Revenue	95%	100%	67%	87%

Sensitivity for all sectors was 100%. Nevertheless, specificity was 0%, 0%, and 67% for employment level, worker earnings, and national revenue, respectively. However, applying k-fold cross validation where k=5, suddenly, the accuracy showed different results. We get the mean value of 5 folds, 82% for employment level, 88% for worker earnings, and 87% for national revenue, respectively, as displayed in Table 1.

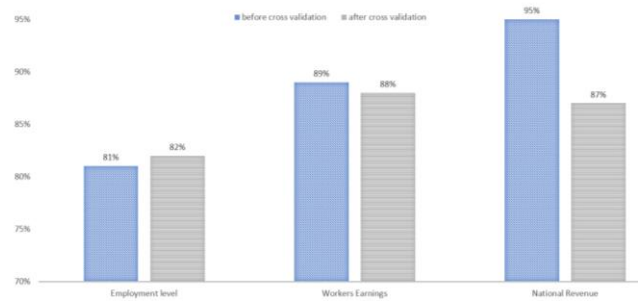


Fig. 3 Results from MLP classifier before and after cross validation (CV)

Figure 3 demonstrates accuracy before and after cross-validation of all three sectors. For employment level, accuracy increased after cross-validation, but accuracy decreased for worker earnings and national revenue.

4-1-2 Random Forest Classifier

By using the Random Forest classifier, we got an accuracy of 81% for employment level, 89% for worker earnings, and 85% for national revenue shown in Table 2. We have used n-jobs equal to -1 for faster processing, logging output as verbose for continuous feedback, n-estimator is 100 and used criterion as *gini* for the Gini impurity.

Table 2: Results from Random Forest classifier

Section	Accuracy	Sensitivity	Specificity	Accuracy(cv)
Employment level	81%	100%	0%	98%
Worker Earnings	89%	100%	0%	99%
National Revenue	85%	100%	67%	98%

Sensitivity for all sectors was 100%. Nevertheless, specificity was 0%, 0%, and 67% for employment level, worker earnings, and national revenue, respectively. However, applying k-fold cross validation where k=5, suddenly, the accuracy showed different results. We get the mean value of 5 folds, 98% for employment level, 99% for worker earnings, and 98% for national revenue, respectively, as displayed in Table 2.

Figure 4 exhibits accuracy before and after cross-validation of all three sectors. For all three sectors, accuracy increased after cross-validation.

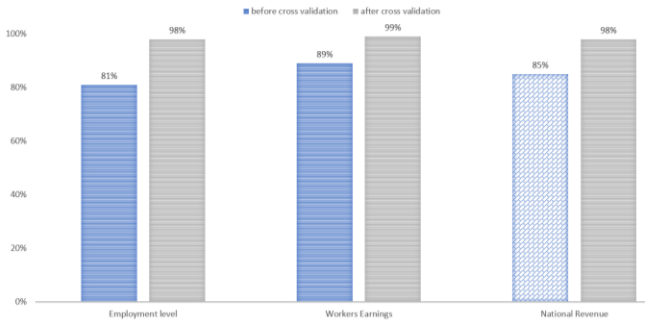


Fig. 4 Results from Random Forest classifier before and after cross validation (CV)

4-2- Result of Regressors

We used two different regressors, long short-term memory and random forest, to determine how much loss a country might face for a pandemic effect. This section will discuss experimental results found from them described in section 4.2. The formulas for calculating Table 1 values for LSTM and Random Forest regressors are equations no. 23,24, 25 and 26.

$$Accuracy = \frac{Correctly\ predicted\ class * 100\%}{Total\ testing\ class} \quad (23)$$

$$Mean\ absolute\ error = (\sum_{i=1}^{n} y_i - k_i)/n \quad (24)$$

In the equation 24, y_i=predicted value k_i=true value and n=total number of data points

$$Mean\ Squared\ error = (\sum_{i=1}^{n} (y_i - y_s)^2)/n \quad (25)$$

In the equation 25, y_s= predicted value y_i= true value and n = total number of data points

$$Root\ Mean\ Squared\ error = \sqrt{(\sum_{i=1}^{n} (y_i - y_s)^2)/n} \quad (26)$$

In the equation 26, y_s= predicted value y_i = true value and n = total number of data points.

4-2-1 Long Term Short-Term Memory

From Table 3, we got an accuracy of 90.04% for employment level, 90.33% for worker earnings, and 92.62% for national revenue using the LSTM. We get 2.63% mean absolute error, 0.12% mean squared error, and 3.46% root mean squared error for national revenue. Again, 4.87% mean absolute error, 0.24% mean squared error, and 4.89% root means squared error for employment level. Finally, 4.68% mean absolute error, 0.22% mean squared error, and 4.72% root means squared error for worker earnings.

Table 3: Results from LSTM regression

Section	*Accuracy*	*MAE*	*MSE*	*RMSE*
Employment level	90.04%	4.87%	0.24%	4.89%
Worker Earnings	90.33%	4.68%	0.22%	4.72%
National Revenue	92.62%	2.63%	0.12%	3.46%

The visual representation of actual and predicted data using LSTM regression are in Figures 5, 6 and 7.

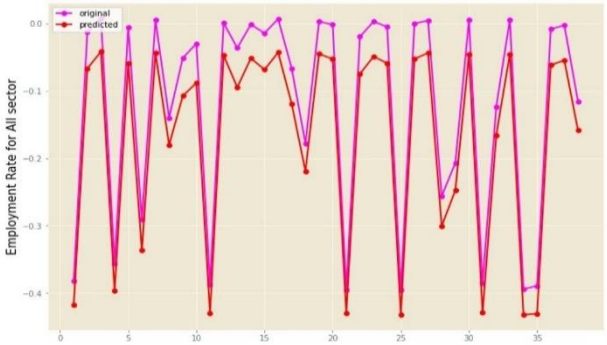


Fig. 5 Comparison of real and modeled (LSTM) data of employment rate of all sectors.

Figure 5 illustrates actual (original labeled) and predicted (predicted labeled) data we found using LSTM for employment rate or level. This figure, the x-axis, shows the number of days, and the y-axis shows the employment rate for all sectors. The predicted rates are 90.04% accurate.

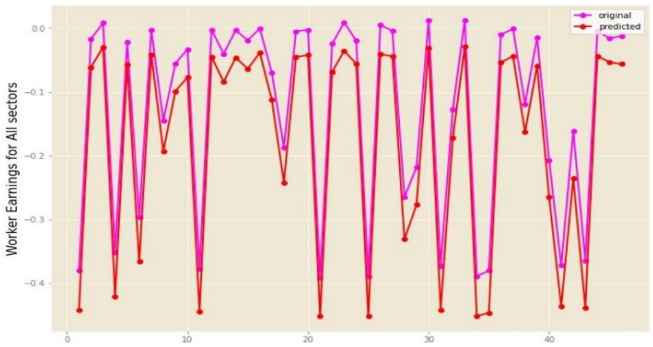


Fig. 6 Comparison of real and modeled (LSTM) data of worker earnings of all sectors

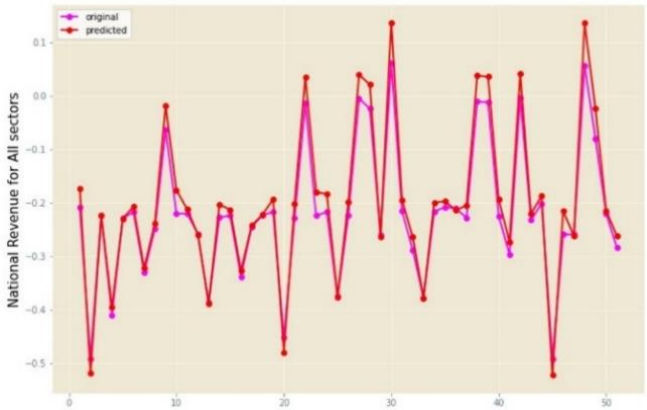


Fig. 7 Comparison of real and modeled (LSTM) data of national revenue all sectors.

Figure 6 represents actual (original labeled) and predicted (predicted labeled) data we found using LSTM for worker earnings. This figure, the x-axis, shows the number of days, and the y-axis shows worker earnings for all sectors. The predicted earnings are 90.33% accurate.

Figure 7 shows actual (original labeled) and predicted (predicted labeled) data we found using LSTM for national revenue. This figure, the x-axis, shows the number of days, and the y-axis shows the national revenue for all sectors. The predicted rates are 92.62% accurate.

4-2-2 Random Forest Regression

From Table 4, we get an accuracy of 93.16% for workers' earnings, 93.18% for employment level, and 95.53% for national revenue using the random forest regression. We get 3.63% mean absolute error, 0.16% mean squared error, and 4.05% root means squared error for employment level. Again, 3.65% mean absolute error, 0.15% mean squared error, and 3.97% root means squared error for worker earnings. Finally, 2.02% mean absolute error, 0.07% mean squared error, and 2.69% root mean squared error for national revenue. From Table 4, it is clear that accuracy

for the employment level, worker earnings, and national revenue is consistent, and the error percentage is low.

Table 4: Results from Random Forest regression

Section	*Accuracy*	*MAE*	*MSE*	*RMSE*
Employment level	93.18%	3.63%	0.16%	4.05%
Worker Earnings	93.16%	3.65%	0.15%	3.97%
National Revenue	95.53%	2.02%	0.07%	2.69%

The visual representation of actual and predicted data using random forest regression are in Figure 8, 9 and 10.

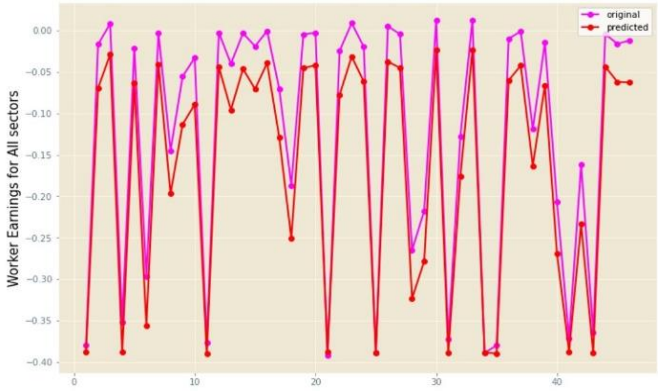


Fig. 8 Comparison of real and modeled (Random Forest Regression) data of worker earnings of all sectors.

Figure 8 illustrates actual (original labeled) and predicted (predicted labeled) data we found using random forest regression for employment rate or level. This figure, the x-axis, shows the number of days, and the y-axis shows the employment rate for all sectors. The predicted rates are 93.18% accurate.

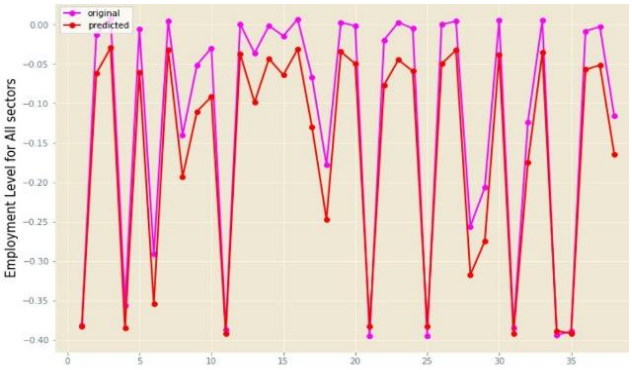


Fig. 9 Comparison of real and modeled (Random Forest Regression) data of employment rate of all sectors.

Figure 9 represents actual (original labeled) and predicted (predicted labeled) data we found using random forest regression for worker earnings. This figure, the x-axis, shows the number of days, and the y-axis shows worker earnings for all sectors. The predicted earnings are 93.18% accurate.

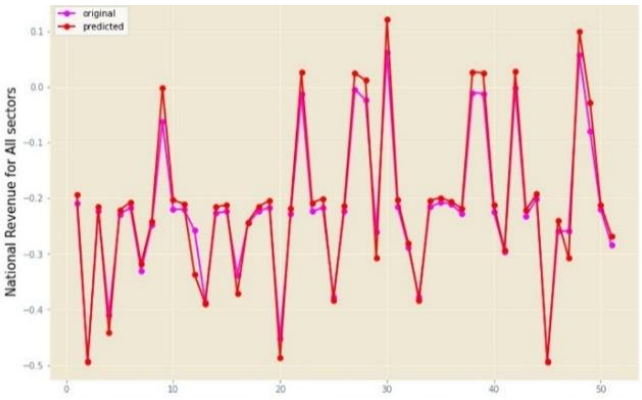


Fig. 10 Comparison of real and modeled (Random Forest Regression) data of national revenue all sectors.

Figure 10 shows actual (original labeled) and predicted (predicted labeled) data we found using random forest regression for national revenue. This figure, the x-axis, shows the number of days, and the y-axis shows the national revenue for all sectors. The predicted rates are 95.53% accurate.

4-3- Discussion

The goal of this model is to provide approximately financial downfall due to the pandemic by analyzing eight months (239 days) data of different economic sectors. The models demonstrated that it is possible to achieve a quality model using economic indicators as inputs through multi-layer perceptron (MLP), long short-term memory (LSTM) RNN, and Random Forest classifier and Regressor. The revenue, employment level, and worker earnings models use the same learning rate. When more data is available, a new detection can be produced with the latest data set by the neural network and random forest algorithms. The findings demonstrate the potential to use these algorithms in the future to model almost the same phenomenon. We have developed this model solution on all available data, which is incredibly constrained. Although the volume of data is minimal, in all three situations, we have superior accuracy.

In the MLP classifier, we got 81% accuracy for employment level, 89% for worker earnings, and 95% for national revenue. Although applying k-fold, employment level and worker earnings model accuracy increased, national revenue model accuracy decreased from 95% to 88%. On the other hand, LSTM revenue got 92.62%,

LSTM employment rate got 90.04%, and LSTM earnings got 90.33%, which is appreciable with this limited amount of data. Including one year of training data and validation data to improve these models to decrease the data limitation problem. Besides, to improve this model for higher accuracy in the future by enclosing bidirectional LSTM, recurrent neural networks, and other appropriate algorithms in the current model.

For the global pandemic situation like COVID-19, we used different economic sectors reports as inputs. We correctly predicted the economic recession outbreak, including revenue, employment rate, and earnings data set. Like the LSTM, MLP, we use random forest classifier and regression to learn national revenue, employment level, and workers' earnings. This model has done an excellent job of predicting the most recent data. The model gives us a positive indicator for foreseeing the economic recession. Though we have used limited data for this detection, we got excellent accuracy in all the cases.

The advantage of the random forest classifier is its tall precision for multi-class classification, which is of the highest need [12]. In addition, as the random forest classifier builds different decision trees, and the ultimate result is assessed depending on the voting of these trees, the issue of overfitting happening in a single choice tree approach is killed. The random selection of feature vectors and random choice of features during learning makes the Random Forest classifier and Regressor solid and productive for any dataset [12]. Forecast analysis from [35] appears to have an accuracy of up to 90% in foreseeing classes with an ensemble approach. Using the random forest classifier model, we got an accuracy of 81% for employment level, 89% for worker earnings, and 85% for national revenue.

On the other hand, using the Random Forest regression model, we got an accuracy of 95.53% for national revenue, 93.18% for employment level, and 93.16% for worker earnings. After applying k-fold cross validation where k=5, suddenly the accuracy level showed different results for random forest classifier. Taking the mean value of 5 folds, we get 98.94% for employment level, 99.04% for worker earnings, and 98.43% for national revenue, prepared with a limited amount of data. Random Forest Classifiers and regression are helpful tools for economists and practitioners dealing with forecasting economic recession detection.

We did provide a warning model system for increasing awareness of an upcoming shock event. Economic crisis detection is essential for practitioners and policymakers since it provides an in-depth understanding of economic linkage breakdowns after a crisis. We proposed a system where we selected some crucial indicators that can be used to predict the economic crisis. As we know that economic conditions are continually changing, and during a pandemic, it changes drastically. So economic downfall detection will remain an open research issue with many situations and challenges to address. Continuously train the data-set with a more diverse set of machine learning algorithms and deep learning architectures that will benefit this research. The use of a developed and more robust technique will enhance the detection of the forecasting economy. Finally, a large amount of data-set will allow us to predict more accurately by filtering out noise embedded in the time series data.

Furthermore, cross-validation and comparison of training and testing data sets are essential. Since it helps one truly understand the models' meaning and efficiency and thus improve the outcome's reliability. In this prospective research aspirants, the importance of these latest technologies needs to be discussed. Before a country faces an economic recession, it is crucial to identify which sector to emphasize to minimize this unexpected scenario. Analyzing the data and detecting the economic impact to help save a country's significant capital using machine learning.

5- Conclusion

Several studies demonstrate that pandemic outbreaks caused considerable economic changes. In this present day, Covid-19 makes a tremendous economic impact on the global economy in terms of revenue, employment level, workers' earnings, and many more. The situation worsens due to the covid-19 pandemic, so prediction of the economic factor is necessary for taking early steps. So, this paper uses machine-learning algorithms to identify and predict the recession. This paper detected the recession in three different sectors of the USA that occurred due to this pandemic and got higher accuracy. When more data is available, the performance will be much higher. Including other algorithms and more data, we can forecast sector-wise economic recessions. We can achieve a model that can detect and forecast recessions that might occur in the future and help stakeholders take decisions as early as possible. To save a country's economy before it breaks apart, this model could play a vital role.

Acknowledgments

This research is funded by Woosong University Academic Research in 2023.

References

[1] M. K. Goyal and A. K. Gupta, “Integrated risk of pandemic: Covid-19 impacts, resilience and recommendations,” Berlin: Springer, 2020.

[2] D. Hanna and Y. Huang, “The impact of sars on asian economies,” Asian Economic Papers, vol. 3, no. 1, 2004, pp. 102–112.

[3] W.Y. Lin, Y.H. Hu, and C.F. Tsai, "Machine learning in financial crisis prediction: a survey," IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 42, no. 4, 2011, pp. 421– 436.

[4] M. R. Keogh-Brown and R. D. Smith, "The economic impact of sars: how does the reality match the predictions?," Health policy, vol. 88, no. 1, 2008, pp. 110–120.

[5] R. D. Smith, M. R. Keogh-Brown, and T. Barnett, "Estimating the economic impact of pandemic influenza: an application of the computable general equilibrium model to the uk," Social science and medicine, vol. 73, no. 2, 2011, pp. 235–244.

[6] M. Fern´andez-Delgado, E. Cernadas, S. Barro, and D. Amorim, "Do we need hundreds of classifiers to solve real world classification problems?," The journal of machine learning research, vol. 15, no. 1, 2014, pp. 3133–3181.

[7] S. Balcaen and H. Ooghe, "35 years of studies on business failure: an overview of the classic statistical methodologies and their related problems," The British Accounting Review, vol. 38, no. 1, 2006, pp. 63–93.

[8] P. R. Kumar and V. Ravi, "Bankruptcy prediction in banks and firms via statistical and intelligent techniques–a review," European journal of operational research, vol. 180, no. 1, 2007, pp. 1–28.

[9] S. J. Fong, G. Li, N. Dey, R. G. Crespo, and E. Herrera-Viedma, "Composite monte carlo decision making under high uncertainty of novel coronavirus epidemic using hybridized deep learning and fuzzy rule induction," Applied Soft Computing, vol. 93, 2020, pp. 106282.

[10] K. Bluwstein, M. Buckmann, A. Joseph, M. Kang, S. Kapadia, and O. Simsek, "Credit growth, the yield curve and financial crisis prediction: evidence from a machine learning approach," 2020.

[11] R. Nyman, and P. Ormerod, "Predicting economic recessions using machine learning algorithms," arXiv preprint arXiv:1701.01428, 2017.

[12] L. Breiman, "Random forests," Machine learning, vol. 45, no. 1, 2001, pp. 5–32.

[13] U. Gr¨omping, "Variable importance assessment in regression: linear regression versus random forest," The American Statistician, vol. 63, no. 4, 2009, pp. 308–319.

[14] Z. Car, S. B. Šegota, N. Anđelić, I. Lorencin, and V.Mrzljak, "Modeling the spread of COVID-19 infection using a multilayer perceptron", Computational and mathematical methods in medicine, Hindawi, vol. 2020, 2020.

[15] J. A. Ohlson, "Financial ratios and the probabilistic prediction of bankruptcy", Journal of accounting research, vol. 18, no. 1, 1980, pp. 109–131.

[16] R. C. West, "A factor-analytic approach to bank condition", Journal of Banking and Finance, vol. 9, no. 2, 1985, pp. 253–266.

[17] A. F. Atiya, "Bankruptcy prediction for credit risk using neural networks: A survey and new results," IEEE Transactions on neural networks, vol. 12, no. 4, 2001, pp. 929–935.

[18] J. H. Min and Y.-C. Lee, "Bankruptcy prediction using support vector machine with optimal choice of kernel function parameters," Expert systems with applications, vol. 28, no. 4, 2005, pp. 603–614.

[19] K.-S. Shin, T. S. Lee, and H.-j. Kim, "An application of support vector machines in bankruptcy prediction model," Expert systems with applications, vol. 28, no. 1, 2005, pp. 127–135.

[20] S. Sarkar, and R. S. Sriram, "Bayesian models for early warning of bank failures", Management Science, vol. 47, no. 11, 2001, pp. 1457–1475.

[21] E. Fedorova, E. Gilenko, and S. Dovzhenko, "Bankruptcy prediction for russian companies: Application of combined classifiers", Expert Systems with Applications, vol. 40, no. 18, 2013, pp. 7285–7293.

[22] J. Abell´an and C. J. Mantas, "Improving experimental studies about ensembles of classifiers for bankruptcy prediction and credit scoring," Expert Systems with Applications, vol. 41, no. 8, 2014, pp. 3825–3830.

[23] R. Chetty, J. N. Friedman, N. Hendren, M. Stepner, and T. O. I. Team, "The economic impacts of covid-19: Evidence from a new public database built using private sector data," National Bureau of Economic Research, Working Paper 27431, June 2020.

[24] R. M. Simon, J. Subramanian, M.-C. Li, and S. Menezes, "Using crossvalidation to evaluate predictive accuracy of survival risk classifiers based on high-dimensional data," Briefings in bioinformatics, vol. 12, no. 3, 2011, pp. 203–214.

[25] Haykin S., "Neural networks and learning machines," 3/E: Pearson Education India, 2010.

[26] I. Syarif, A. Prugel-Bennett, and G. Wills, "Svm parameter optimization using grid search and genetic algorithm to improve classification performance," Telkomnika, vol. 14, no. 4, 2016, pp. 1502.

[27] L. Fraiwan, K. Lweesy, N. Khasawneh, H. Wenz, and H. Dickhaus, "Automated sleep stage identification system based on time–frequency analysis of a single EEG channel and random forest classifier," Computer methods and programs in biomedicine, vol. 108, no. 1, 2012, pp. 10-19.

[28] V. Y. Kulkarni and P. K. Sinha, "Pruning of random forest classifiers: A survey and future directions," in 2012 International Conference on Data Science and Engineering, IEEE, 2012, pp. 64–68.

[29] Y. E. Cakra and B. D. Trisedya, "Stock price prediction using linear regression based on sentiment analysis," in 2015 international conference on advanced computer science and information systems, IEEE, 2015, pp. 147–154.

[30] B. Wang, L. Gao, and Z. Juan, "Travel mode detection using gps data and socioeconomic attributes based on a random forest classifier," IEEE Transactions on Intelligent Transportation Systems, vol. 19, no. 5, 2017, pp. 1547–1558.

[31] F. A. Gers, J. Schmidhuber, and F. Cummins, "Neural Nets WIRN Vietri-99," Continual prediction using LSTM with forget gates: Springer, 1999.

[32] K. Suri and R. Gupta, "Transfer learning for semg-based hand gesture classification using deep learning in a master-slave architecture," in 3rd International Conference on Contemporary Computing and Informatics (IC3I), IEEE, 2018, pp. 178–183.

[33] S. Polyzos, A. Samitas, and A. E. Spyridou, "Tourism demand and the covid-19 pandemic: An lstm approach," Tourism Recreation Research, vol. 46, no. 2, 2021, pp. 175–187.

[34] L. Breiman, "Bagging predictors," Machine learning, vol. 24, no. 2, 1996, pp. 123–140.

[35] J. Kevric and A. Subasi, "Comparison of signal decomposition methods in classification of EEG signals for

motor-imagery BCI system," Biomedical Signal Processing and Control, vol. 31, 2017, pp. 398–406.

[36] Md. Monirul Islam, Mohammod Abul Kashem, Jia Uddin, "Fish Survival Prediction in an Aquatic Environment Using Random Forest Model," IAES International Journal of Artificial Intelligence (IJ-AI), Indonesia, 2021, vol. 10, no. 3, pp. 614-622.

[37] J. Ferdoush, B. N. Mahmud, A. Chakrabarty, J. Uddin, "A Short-Term Hybrid Forecasting Model for Time Series Electrical-Load Data using Random Forest and Bidirectional Long Short-Term Memory" International Journal of Electrical and Computer Engineering, Indonesia, 2020, vol. 11, no. 1, pp. 763-771.

[38] R. Islam, J. Uddin, J.M. Kim, "An Acoustic Emission Sensor based Fault Diagnosis of Induction Motors using Gabor filter and Multiclass SVM", Journal of Ad-hoc and Sensors Wireless Networks, Old city publisher, 2016, vol. 34, no. 1, pp. 273-287.