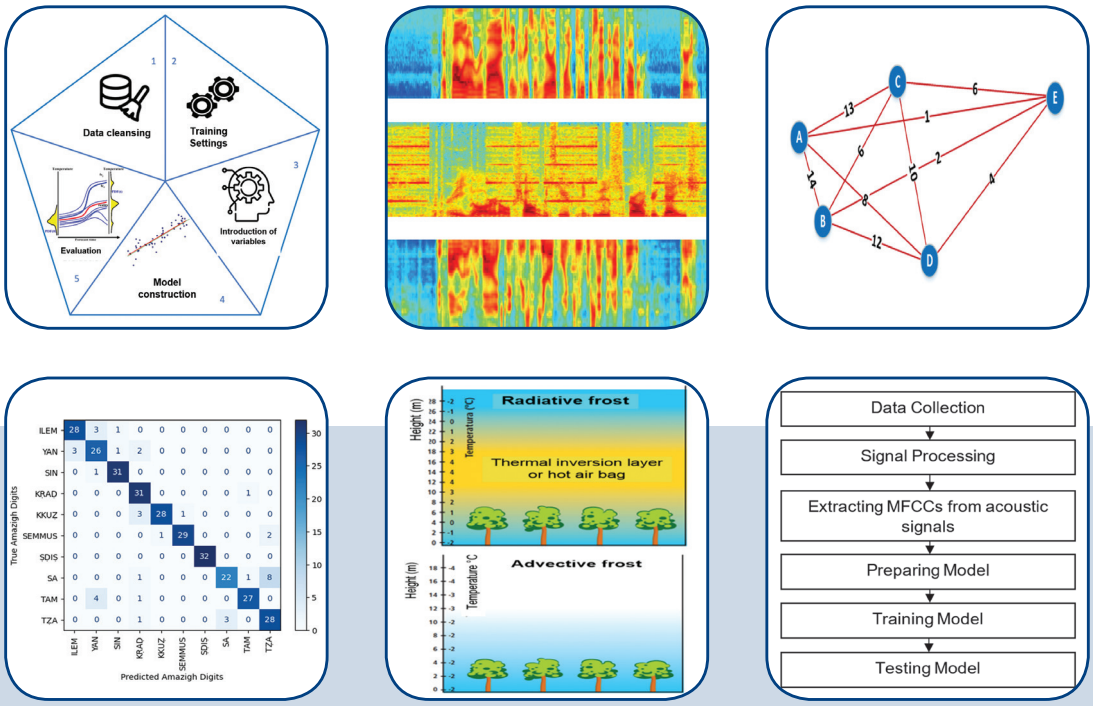
Volume 14, Number 7, 2023 ISSN 1847-6996

FERIT
FACULTY OF ELECTRICAL ENGINEERING, COMPUTER SCIENCE AND INFORMATION TECHNOLOGY **OSIJEK**

IJECES International Journal of Electrical and Computer Engineering Systems

International Journal of Electrical and Computer Engineering Systems



ISSN: 1847-6996

INTERNATIONAL JOURNAL OF ELECTRICAL AND COMPUTER ENGINEERING SYSTEMS

Published by Faculty of Electrical Engineering, Computer Science and Information Technology Osijek, Josip Juraj Strossmayer University of Osijek, Croatia

Osijek, Croatia | Volume 14, Number 7, 2023 | Pages 733 - 819

The International Journal of Electrical and Computer Engineering Systems is published with the financial support of the Ministry of Science and Education of the Republic of Croatia

CONTACT

International Journal of Electrical and Computer Engineering Systems (IJECES)

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek,
Josip Juraj Strossmayer University of Osijek, Croatia
Kneza Trpimira 2b, 31000 Osijek, Croatia
Phone: +38531224600, Fax: +38531224605
e-mail: ijeces@ferit.hr

Subscription Information

The annual subscription rate is 50€ for individuals, 25€ for students and 150€ for libraries.
Giro account: 2390001 - 1100016777,
Croatian Postal Bank

Journal is referred in:

- Scopus
- Web of Science Core Collection (Emerging Sources Citation Index - ESCI)
- Google Scholar
- CiteFactor
- Genamics
- Hrčak
- Ulrichweb
- Reaxys
- Embase
- Engineering Village

Bibliographic Information

Commenced in 2010.
ISSN: 1847-6996
e-ISSN: 1847-7003
Published: quarterly
Circulation: 300

IJECES online

https://ijeces.ferit.hr

TABLE OF CONTENTS

Novel Clustering Techniques in Wireless Sensor Networks – A Survey

Review Paper

T.C. Swetha Priya
Department of Computer Science and Engineering,
JNTUH University College of Engineering, Science and Technology,
Hyderabad, Telangana, India
tcswetha3552@gmail.com

R. Sridevi
Department of Computer Science and Engineering,
JNTUH University College of Engineering, Science and Technology,
Hyderabad, Telangana, India
sridevirangu@jntuh.ac.in

Abstract *– A study of Wireless Sensor Networks has been growing tremendously these days. Wireless Sensor Networks play a major role in various fields ranging from smart homes to health care. WSN's operate independently in remote places. Because of tiny size of the nodes in such type of networks, they have a limited number of resources in terms of energy and power. Basically, sensor networks can be classified into flat and cluster based Wireless Sensor Networks. But, Clustering based Sensor Networks play a major role in reducing the energy consumption in Wireless Sensor Networks. Clustering also focuses on solving the No.s that arise during transmission of data. Clustering will group nodes into clusters and elects Cluster Heads for all clusters in the network. Then the nodes sense data and send that data to cluster head where the aggregation of data will take place. This paper focuses on various novel clustering techniques that improve the network's lifetime.*

Keywords*: Wireless Sensor Network, Clustering, Fuzzy clustering, Optimal Cluster Size, Adaptive Clustering, Quad Clustering, K-Means, Dragonfly Algorithm*

1. INTRODUCTION

Due to recent advancements in technology, Wireless Sensor Networks have a great demand in the market. Their versatility has increased the demand for this technology. A Wireless Sensor network is made up of a huge number of less cost sensor nodes which are of small size and limited power [1]. These nodes have the capability to communicate over shorter distances. They have limited resources in terms of storage, bandwidth, energy and computational abilities. When a node fails or becomes inactive there is no chance for replacement of that node because of their constrained amount of energy. Because the sensors are operated with battery, the lifetime of a network and depletion of energy are the major challenges in this type of networks.

Basically, the sensor networks are classified into flat Wireless Sensor Networks and cluster based Wireless Sensor Networks. The flat Wireless Sensor Networks are traditional networks in which all the nodes will take the role to generate data and route data. But this type of network is not suitable for large-sized networks because the nodes will flood the data into the network. This will generate duplicate data causing a lot of overhead during communication of data. So, the traditional flat WSN works well in small networks. The sensor nodes and base station will be randomly placed in the sensor network area. The main responsibility of these nodes is sensing the data from the environmental conditions of a particular application and forwarding that data to Base Station. This Base Station may be in a place that is distant to the nodes. So, the data that is received from nodes may be repeated or redundant leading to energy depletion. To overcome this, clustering technique is introduced in Wireless Sensor Networks. The Clustering method guarantees increase in the network lifetime. Clustering is one of the best methods for large sensor networks. Clustering is partitioning the network into sub-networks [2]. It helps to manage the network easily and it will be easy to extend the network. Each cluster consists of a Cluster-Head, sensor nodes and a Base Station. The cluster heads are selected through any of the various clustering algorithms based on several factors such as residual node energy, distance etc. The sensor nodes will gather data and send to the cluster head. The cluster heads act as a local aggrega-

tor and sends this data to Base Station. This data can be sent in either a single-hop or multiple-hop manner [3]. The communication in cluster based networks can be of two types: Inter cluster communication and Intra cluster communication. The features that make the cluster based Wireless Sensor Network an effective choice are data fusion, load balancing, energy efficiency, maintenance of cluster, secure transmission of data and better network lifetime.

This paper provides a comprehensive overview of various novel approaches to perform clustering. This paper provides a study of the cluster-based techniques in Wireless Sensor Networks. This section deals with the introduction of Wireless Sensor Networks and the significance of Clustering in Wireless Sensor Networks. The rest of the paper deals with providing a brief overview about Clustering schemes and comparison of clustering methods. The conclusion is given in the last section of this paper.

2. CLUSTERING TECHNIQUES

2.1. LOW ENERGY ADAPTIVE CLUSTERING

In this type of clustering [1, 2], the cluster heads are selected in such a way that each node present in the sensor network gets an opportunity to be a cluster head. The cluster heads consume more energy than that of cluster members. So, we need to distribute the load among all sensor nodes so that a node's energy do not get depleted after some amount of time because of frequent selection of nodes as Cluster Head. This clustering involves 2 phases: 1. Setup and 2. Steady phase. In the first phase, formation of clusters and selection of Cluster heads based on the number of times that a particular node has been selected as cluster head is done. If the probability of a node is less than the selected threshold value then that node is selected as a Cluster Head. In the second phase, the data that is gathered by Cluster heads will be sent to the Base Station. One disadvantage with this method is the node's initial energy will not be considered during the cluster head selection. Also, the nodes which already became Cluster heads are again chosen as Cluster heads causing depletion of energy of such nodes. This type of clustering will not be appropriate for larger networks.

2.2. EVENT TO SINK DIRECTED CLUSTERING

This type of clustering [4] will provide high efficiency by reducing consumption of energy. When an event is discovered by a node, a report is sent to the sink node. This data which is collected by the nodes is sent to cluster head to avoid data redundancy. In this type of clustering, unnecessary cluster formation is avoided by allowing formation of clusters only on the occurrence of an event. The movement of data is minimized within a cluster because the formation of clusters is in the direction where an event occurs. The selection of Cluster Heads will be done from upstream whereas non-cluster sensor nodes will be selected from the nodes that are downstream. The data flow will be unidirectional. The node delay in LEACH is almost double that of Event to sink directed clustering.

2.3. LOAD BALANCED CLUSTERING

In this type of clustering [5], an assistant node is chosen from all the nodes and this node helps cluster head in performing the aggregation and processing of data. This selected node helps in the transmission of data to Base Station. The received data is processed by Cluster head and transmits it to the selected assistant node. This assistant node forwards this data to Base Station. But the drawback with this type of clustering is that the flow of data among sensor nodes is not constant. So, the nodes that are nearer to Base station will receive more data compared to nodes that are far away. So there is a chance for depletion of energy of closest nodes.

2.4. ENERGY EFFICIENT HIERARCHICAL CLUSTERING

In this type of clustering [6, 7], each sensor node has the ability to decide upon whether it can be a Cluster head (CH) or not. If a node chooses to be a cluster head, then it announces its presence to all adjacent nodes. Now, this Cluster head is known as Volunteer Cluster Head. All sensor nodes which are not present within k-hop distance away from Cluster Head will retrieve the data from CH. If this advertisement message is received by any of the nodes other than the Cluster Head then those nodes will become the members of Cluster.

2.5. WEIGHT BASED CLUSTERING

In this type of clustering [8], the parameter for the selection of Cluster Head is "Weight". This can be computed based on several aspects such as the number of times that a particular node has been selected as cluster head, distance between the cluster heads and nodes, remaining node energy. The node weight is computed at each periodic interval of Clustering process. Here the cluster formation will be done in a manner in which less energy is consumed in a network. The node with highest remaining energy is chosen to be a Cluster Head. This means that the nodes with less energy will be excluded from being Cluster Heads. If any node has energy greater than that of remaining energy then that node will become a Cluster Head whereas the other nodes will be cluster members. Here the threshold value is set based on remaining energy. So, if any nodes' energy falls below this threshold value, then that node will be treated as a "dead node". Because of this, each and every node present in the network will send a message about its presence after periodic intervals of Clustering process. This method is best in comparison with other clustering methods because there are a smaller number of dead nodes.

2.6. CLUSTER HEAD ROTATION SCHEME

In this method [9, 10], there is a chance for a sensor node to go to inactive mode for an arbitrary amount of time. Then the node announces itself as a Cluster Head (CH) and transmits this announcement message to all sensor nodes present within the scope of the network. Then the adjacent nodes that are within the reach sends their responses about the selection of cluster head and based on their opinion the Cluster head will decide whether to remain as a Cluster Head or to change its role as a member of cluster. Here cluster head nodes will be rotated by giving an opportunity to the next nodes based on previous information collected at an initial point of time. The information that is collected initially includes a node identifier; its remaining energy and number of times it got a chance to become CH [11]. This information is collected in the form of packets. In this manner, Cluster Heads are rotated in a timely manner.

2.7. SELF-ORGANIZED ENERGY CONSCIOUS CLUSTERING (SECC)

This type of clustering has 2 steps [12, 13, 14]. In the first step, node energy of all nodes and neighbor node distance is computed. Here, the average distance of nodes is compared and the node within same distance gets combined and forms a cluster. Here, the clusters are formed in an energy-conscious manner based on the Threshold value. Whereas the nodes which have less value than that of Threshold goes to inactive state and further they do not involve in any of the sensor related activities.

2.8. ENERGY AWARE FUZZY CLUSTERING ALGORITHM (EAFCA)

In this algorithm [13-20] [21], first the nodes are deployed and then the computation of distance between the sensors is done. In this method, a signal is sent from sink node to all remaining nodes. Now, by considering the strength of the signal that is received, the distance from sink to sensor nodes is computed. Then, some cluster heads are picked from the entire network. Then, the threshold value is calculated and this value has to be transmitted to all nodes present in the network. Now, the sensor nodes will generate an arbitrary number and this is compared with the threshold value that it has received. The node with highest random generated number than threshold will declare itself as a Cluster Head (CH) whereas the remaining nodes will be normal sensor nodes. This method adapts creation of 2-hop clusters. It considers the factors like remaining node energy, degree of a node at 2-hop distance and CH position also employs a fuzzy logic for selection of a node as a cluster head permanently [22]. The node degree plays a major role in the selection of permanent cluster head. It means that a node with higher node degree becomes a permanent Cluster Head. This indicates that a node will be either a Cluster node member or a Cluster Head. So, the Cluster heads will make periodic announcements to the nodes to join. So, the nodes will decide to join the clusters that are nearer to them. If the distance to the two clusters is same, then the nodes from which the announcement to join is received first will be considered and the node will join in that cluster. After formation of Clusters, the information will be gathered by sensor nodes and will be forwarded to the Cluster heads. These cluster heads are responsible for aggregation of data and sending the information to sink.

2.9. ENERGY EFFICIENT RECURSIVE CLUSTERING (EERC)

This type of clustering [7, 23] is an event-oriented clustering approach. The Clusters are formed only upon the event occurrence. Here, the clusters are generated in a recursive manner and the clusters are formed only for reducing energy consumption while transmitting data. This method follows 2 steps in formation of clusters after deployment of nodes. In the first step, the Euclidean distance between two nodes is calculated. Then, based on that value clusters are formed resulting in the first level of clusters. Because this method follows recursive clustering, the first level of clusters is again partitioned to form the second level of clusters after considering the distance between nodes. After the formation of Clusters, the cluster head selection is done based on Round Robin Scheduling. In this Scheduling, the turn-around time is computed and a node with minimum value and highest energy will be selected as a Cluster Head. Now this computation of turn-around time is done after every 2 rounds. Then the node with minimum value is selected as a Cluster Head. Now the data is gathered and is sent to Head node where the aggregation of data will be done and finally data is sent to Base station by multi hop routing. This method reduces the consumption of energy and extra overhead of redundant data by aggregation.

2.10. ADAPTIVE DISTRIBUTED CLUSTERING ALGORITHM (ADCA)

In this type of clustering [24] [25], Similarity value is measured. The sensor nodes that are adjacent to each other may have a chance of producing same data. So, the Similarity will be high for such sensor nodes. Also, there is a chance that the data generated at consecutive time intervals from the same sensor nodes will be similar. The data redundancy and generation of data can also be limited through this approach helping in reducing the energy. This method follows two steps: 1. Formation of Cluster and 2. Adaptive sleep duty cycle. In the first phase, the sink node will analyze the rate at which the information is generated and Similarity of information from sensor nodes. Based on the analyzed information, the nodes will form groups leading to Clusters. The clusters that are formed will not be of same size. Now, based on the remaining node energy and connectivity of nodes the Cluster Heads will be chosen. In the second phase, a

comparison is made between the rate at which the data is generated by member nodes and the minimum level of threshold. Then the nodes with a low threshold will be assigned a duty cycle for a specific amount of time period. This sleep or awake time period has to be informed to all sensor nodes. A fair distributed scheduling is done to limit the consumption of energy. The Cluster heads gather data from the nodes and checks if there is any data that is similar. If any deviation in the data is identified then this has to be reported to sink node. Then the sink will decide whether to change sleep-awake duty cycle or whether to perform re clustering process based on the necessity. In this way some part of the periodic monitoring work will be shared by Cluster heads instead of depleting the energy of sink node alone.

2.11. IMPROVED ENERGY AWARE DISTRIBUTED UNEQUAL CLUSTERING

In this type of clustering, the random deployment of nodes is done and the nodes will have varying energies. [24, 26, 27]. The Base Station is positioned far from the Sensor Network area. Then after deployment of nodes, a signal is broadcasted to all nodes. Based on the strength of signal received the approximate distance from nodes to the Base Station is estimated. This method works in the form of rounds. A signal will be broadcasted to all the nodes by the Base Station after the deployment of nodes. Now based on the strength of the signal received, the nodes will calculate the distance. Each round involves 2 phases: 1. Setup and 2. Steady phase. The first phase comprises of 3 steps: 1. Neighborhood information Collection, 2. Cluster Head Selection and 3. Formation of Cluster head. After the formation of clusters, the second stage starts where the data transmission take place. In the steady state, there will be a number of major slots. After completion of one major slot, the rotation of cluster heads will take place. This version of extended EADUC helps in reduction of Cluster overheads and in balancing consumption of energy.

2.12. HIERARCHICAL CLUSTERING

This type of clustering [24, 28] works in 2 layers. The selection of Cluster heads will be done in the first layer and data routing is done in the second layer. Here the selection of cluster head is of major concern as it aims at reducing the consumption of energy and delay [29]. This method considers 3 parameters: energy remaining in nodes, distance from the Base Station and node degree. The weight of each node will be computed based on these 3 factors. Finally, the Cluster head will be the node which has maximum weight.

2.13. TWO LEVEL HIERARCHY FOR LOW ENERGY ADAPTIVE CLUSTERING HIERARCHY

In Hierarchical clustering [25], unbalanced Clustering is employed. But if there is no balancing in clustering, there is a chance for more consumption of energy during data collection by some of the cluster Heads. So to overcome this the clusters must be balanced by employing various levels. In this method, all the levels are connected that is first level Cluster Heads are connected to second level cluster heads which in turn gets connected to all the member nodes of the Cluster. Here the first level will summarize the data and second level will aggregate data whatever is received from the first level which will reduce the packets sent over the network and also reduces depletion of energy level of nodes.

2.14. MOBILE SINK-BASED APPROACH

Generally, the nodes are assumed to be static and are deployed randomly. But the sink node is always mobile. So, the burden will be on one mobile sink node [30]. To overcome this, in this method two mobile sink nodes will be deployed [31]. After completion of every half round these sink nodes move in anti-clockwise direction. The sensor network area is partitioned into equal areas and each area consists of unequal clusters [32]. At the beginning, the center node is chosen as Cluster Head. After completion of each round, the sensor node which is closest to the center of sensor network area and having highest remaining energy will be chosen as a cluster head. As the sink nodes are mobile, a routing mechanism is also required for transmission of data to the current location of sink. The new path information will be maintained by only a single Cluster head [33].

2.15. CLUSTERING USING FUZZY LOGIC

In Fuzzy Logic [13-19, 34], the 2 variables that are taken as input are: Remaining Energy and Required energy. This type of clustering works in 2 stages. In the first step, based on the strength of signal received every node will compute the energy that is needed to send 'n' number of bits to the Base Station. Here, a Timer is set by all nodes that is conversely proportional to an output variable "chance". If the value of "chance" is higher than it has more possibility for being a Cluster Head. A countdown timer is set by all the nodes. If this value becomes zero, then that node will become a CH. Then the distance is computed and based on the distance the formation of clusters will happen. In the second step, Time Division Multiple Access scheduling is used by Cluster Head and the cluster members will also be informed about this schedule.

2.16. CLUSTERING USING NODE RELEVANCE

This clustering [35] focuses on calculation of adjacency matrix with binary values specifying only the nodes that are adjacent to each other. This method is appropriate for nodes that are stationary because this reduces frequent announcements from nodes about their occurrence to Cluster Heads. Then the selection of Cluster Heads will be done in several levels based on factors such as number of times a node has acted as Cluster Head, association of nodes and node advertise-

ments. This must be repeated before every round. This method selects Cluster Heads for several rounds at a time by using Relevance index (R-Index). By examining the structure of the network, the nodes' R-index will be computed from adjacency matrix. The distance will be calculated between Cluster heads. The minimum distance and the adjacency matrix will be chosen to perform clustering. In this manner, the Cluster heads for multiple rounds will be computed at once and if suddenly a node fails, then adjacency matrix will be recomputed and the same process is repeated.

2.17. DISTRIBUTED FUZZY LOGIC BASED ENERGY AWARE AND COVERAGE PRESERVING UNEQUAL CLUSTERING (DECUC)

In this type of clustering [18, 35-36], a distributed method is used in the formation of clusters. The selection of clusters will be done by sensor nodes but not the sink node. This method employs 3 algorithms. In the first algorithm, the fuzzy logic inputs that are used are distance between nodes to sink, Energy level and Area coverage. Now, the degree of Fuzzy output is determined and the node with high chance or output value will be chosen as center of Cluster. Now, sensor nodes will transmit data to this cluster center where the data aggregation is done and the data that is aggregated will be sent to Sink. In the next round, the cluster centers will be selected again. In the second algorithm, election will not be there. Suppose if any Cluster Center exits the network before this round, then one of the nodes with highest degree from the previous round will be selected as new Cluster center. Here, the nodes may be operating in different algorithms in multiple rounds. The nodes that have not achieved the threshold level of energy in the second algorithm will still be in the same cluster whereas the nodes which have reached threshold will proceed to the second algorithm. Each cluster head must calculate a different threshold value. In some rounds, the value of the threshold should be reduced to half to reduce unwanted elections. The high threshold will be generated by the cluster heads with less value of chance. These nodes will be allowed to enter into the third algorithm. This clustering follows an adaptive threshold to make it significant among other methods. Now, the nodes that are in the third algorithm will have to reselect the cluster head. As the Cluster head selected during the first round will be the best, the next cluster head that is chosen must be in a good condition with respect to the energy and must be at a distance nearer to cluster center. Now the inputs that are given to the third algorithm of fuzzy logic method will be the remaining energy and distance to cluster center. Now after defuzzification, the output of inference engine is taken and sensor node must compare this value with adjacent nodes' value. The one with highest chance is chosen as cluster center. Now after these subsequent selections, the nodes in third algorithm will move to first and the sensors in second algorithm will move to third if the required threshold is reached. Otherwise they remain in same algorithm. Here, the selection of cluster centers will be reduced in different levels and also it controls the information passing in the network thereby reducing the energy.

2.18. OPTIMAL CLUSTER SIZE SELECTION-BASED CLUSTERING PROTOCOL USING FUZZY LOGIC (OCSSP)

In the clustering methods that involve fuzzy logic [37] the determination of output variables for the selection of Cluster Head involve multiple input variables. A different combination of input variables will generate different output variables called as "chance" which are used in the cluster head selection. So, a new clustering algorithm that involves multiple input and multiple output variables will be used to increase the lifetime of a network. To achieve better performance of the network, selection of inference rules will play a major role. After completion of several rounds, the rules will be decided. A two-fold restriction is imposed on the cluster size to reduce the load on the entire network. This clustering computes output variable values based on varying factors of nodes. Here, the cluster head selection will be done in first step and data transmission is done in second step. The cluster members will sense information from the environment and the cluster heads will in turn collect data from these member nodes. The Fuzzy inference system will be used in this type of clustering. This method involves several rounds. Each round has 2 steps: 1. Setup and 2. Steady phase. In the first phase, the Base Station implements fuzzy inference system for the selection of cluster head. Each Cluster head will provide space for a restricted number of members. The fuzzy Inference system is used for efficient supervision of cluster members onto which cluster head is appropriate to join. In a particular round, any node that has highest value of chance is selected as Cluster Head. Then, in the second step through this cluster head the transmission of data will be done to the Base Station.

2.19. QUAD CLUSTERING BASED ON K-MEANS ALGORITHM

This type of clustering [38] adapts K-Means clustering algorithm. The K-Means clustering is one of the unsupervised Clustering techniques. In sensor networks, this technique is employed for the cluster formation in a particular area. The whole network area is divided into 'n' clusters. The clusters have to be formed by iteratively computing the Euclidean distance between the selected data points. Now, the data points will be assigned to nearest cluster center. After computing the mean of the selected data points of corresponding clusters then a new cluster center has to be found. If there is a change in the assignment of clusters, then only this process has to be repeated for all the newly formed clusters. This clustering method involves three steps. In the first step, by using K-Means algorithm the clusters will be formed. The re clustering process must include deter-

mining the cluster's centroid position and inter-node Euclidean distances. Then the nodes' with small Euclidean distance and high residual energy will be chosen as Cluster head. In the second step, the entire network is partitioned into 4 parts. This partition is to improve energy efficiency and network coverage. Now, each partition of the network will be working in isolation. In the third step, each quadrant has a separate cluster Head. Now, again K-Means clustering algorithm must be implemented on each quadrant to further partition this cluster into 4 parts. Hence it is known as Quad Clustering. Before the final cluster head in a quad cluster is selected, the cluster head energy has to be compared with selected threshold. When compared to a full cluster, these quad clusters will utilize half of the total energy by reducing the energy consumption of network.

2.20. DRAGONFLY ALGORITHM FOR HIERARCHICAL CLUSTERING

The dragonfly based clustering [39, 40] comes under biologically-inspired method because it makes use of the way the dragonflies moves. This helps to achieve good optimization in terms of finding the better cluster head. This algorithm calculates the number of nodes that are alive and the lifetime of a network. The nodes are assumed to be present at arbitrary positions and then the clusters are formed using LEACH method. At the beginning, for the selection of cluster head the nodes with highest energy remaining will be considered. Further, the dragon fly algorithm will select cluster head. In this method, first energy and distance variables have to be initialized to locate CH node. Now the fitness function for all the Cluster heads has to be computed and the Cluster head with a least value will be selected as Cluster Head. This data has to be sent to all members of cluster heads. Then the data gathered by member nodes will be aggregated and sent to Base Station. The dragon flies form sub groups and move in multiple directions. They update their location based on factors such as structure, alignment, similarity and disturbances. To indicate the movement of dragon flies, a step vector has to be calculated. With the help of random walk concept, they determine the neighbors and solution to step vector is found. Then by utilizing the nearest distance and level of energies, a fitness value is calculated. The nodes with least fitness value will be chosen as Cluster Head. In this method, the best Cluster heads are the ones that are closer to the adjacent Cluster heads and the Base station because this will save energy.

2.21. ENERGY EFFICIENT SLEEP AWAKE AWARE NODE SCHEDULE CLUSTERING

This type of clustering [41-43] makes use of node scheduling concept. Here, the nodes will be in either sleep or active mode at the time of transmission of data among nodes. This method implements the concept of pairing to improve the lifetime of a network. The main reason for this method is to overcome the redundant data. When pairing of nodes is done then the nodes may sometimes sense similar data and forward that to the Cluster Head. Because of coupling, only one node will be active at a time by switching off another node temporarily during communication. It works on uniform network. This works well when the sensor nodes are static and do not require any transmission of data between them. This reduces the redundant transmission of data and increases the lifetime of sensor network. Here, first the data that is retrieved from various sources have to be segregated and a common pattern must be obtained and the result has to be displayed.

2.22. NODE OVERHAUL SCHEME FOR ENERGY EFFICIENT CLUSTERING

In LEACH uniform size clustering approach, the selection of cluster head will depend on probability. At the beginning of formation of clusters, the nodes will join with the cluster heads that are nearby. Because of applying the probability approach in this method, there is a chance of clusters being in various sizes. All the nodes will be having a cluster head. Based on the second-best choice, re clustering is performed for obtaining uniform clusters by moving some of the nodes from a larger cluster to other clusters [44]. To perform this operation, first the largest cluster among all the clusters must be found and then the distance must be computed between the node members of cluster and remaining cluster heads. From this the second-best choice of cluster head is obtained. The nodes having less distance to remaining cluster heads will be assigned to second best cluster heads whereas nodes closer to boundary of clusters will be assigned to remaining cluster heads. This process has to be repeated until equal sized clusters are formed. Then, a Time Division Multiple Access schedule has to be created by all cluster heads present in the sensor network. Then the data has to be transmitted in the assigned time schedules. During transmission of data, aggregation of data has to be done by Cluster Heads and then it is forwarded to the Base station. After every round, re clustering is performed to turn the roles of cluster heads. This method helps in balancing the network load by creation of uniform clusters.

2.23. DISTANCE AND ENERGY CONSTRAINED K-MEANS CLUSTERING SCHEME (DEKCS)

This type of clustering [45-47] aims at improving the performance of the network and selection of cluster head. This method uses k-Means Clustering to perform clustering. This method chooses the proximity principle that means the nearest node that is closest to the greatest number of nodes in the network. This makes sure that all the nodes in every cluster must be nearer to the cluster head so that less amount of energy will be consumed during transmission of data. The K-Means algorithm [47] divides the unlabeled multi-dimensional data into k number of clusters. The number of clusters has to be determined before itself. This method tries to reduce the distance between the nodes within a cluster and the

cluster center thereby increasing the distance between clusters. Here, the clusters that are formed will be in such way that all nodes are 1-hop away to cluster Head. For the number of clusters to be optimal, an elbow method is used. This method computes intra cluster sum of squares for the multidimensional data. After clustering, cluster head must be selected. During the selection of cluster head, the position of cluster has to be considered because the proximity principle is used in this clustering. The cost function is used to compute the Euclidean distance between a node that is selected and remaining cluster node members. From this, the nearest node that is closest to the greatest number of nodes in the network will be chosen as Cluster Head. The chosen cluster head node remaining energy level must be above the threshold value that is set. This is done to avoid dead nodes and to not make the network disconnected. If any node has energy less than the value set for threshold then it has to be made as an "Ordinary Node". The nodes that are dead will be taken out of the network. This process has to be repeated so as to ensure that the network is always connected. Here, a radio frequency link is used to gather data from cluster heads.

3. COMPARISON OF CLUSTERING TECHNIQUES

The comparison of various clustering techniques is summarized as shown in Table 1.

Table 1. Comparison of Clustering Techniques

Clustering Approach	Strengths	Weaknesses
Low Energy Adaptive Clustering [1]	• Depletion of node Energy • It balances the energy consumption of the entire network • It extends the lifetime of the network	• It is not appropriate for larger networks • The node's initial energy will not be considered during the cluster head selection
Event to sink directed clustering [4]	• It avoids the formation of unnecessary clusters in the network. • Maintenance overhead is reduced	• Clustering depends on selection of sink nodes.
Load Balanced Clustering [5]	• Flow of data among sensor nodes is not constant • Balances consumption of energy	• Energy is balanced only by restricting the size of nodes • Depletion of energy of closest nodes
Energy Efficient Hierarchical Clustering [7]	• Easy implementation • Minimize total energy • Suitable for large network	• Not suitable for less number of nodes
Weight Based Clustering [8]	• Less number of dead nodes • Low energy sensors will never be selected as Cluster Heads.	• Large number of nodes may be elected as cluster heads.
Cluster Head Rotation Scheme [9]	• Balances energy of nodes • Suitable for sensing in environments where we need to sense for longer period of time	• Not suitable for network with less number of nodes.
Self Organized Energy Conscious Clustering [12]	• Conserves energy • Lifetime of network is improved	• Do not support scalability and mobility of nodes
Energy Aware Fuzzy Clustering Algorithm [13-21]	• Aggregation of data • Lifetime of network is improved • Enhances throughput	• Do not support real time data
Energy Efficient Recursive Clustering [7, 23]	• Reduces the consumption of energy and extra overhead of redundant data	• Repeatedly Clustering need to applied causing overhead.
Adaptive distributed Clustering Algorithm [24, 25]	• Reduces depletion of energy by avoiding redundant data	• Do not support mobility.
Improved Energy Aware Distributed Unequal Clustering [24, 26-27]	• Reduces Cluster overheads • Balances consumption of energy	• Clusters formed are of uneven size.
Hierarchical Clustering [29,34]	• Reduces the consumption of energy • Improves lifetime of network.	• Imbalances consumption of energy
Two Level Hierarchy for Low Energy Adaptive Clustering Hierarchy [25]	• Reduces depletion of energy of nodes • Provides proper load distribution among sensors.	• Lowers overall energy consumption.
Mobile Sink - based Approach [30]	• Traffic load is reduced by using Data aggregation • More suitable for large-scale deployed networks • It prolongs the lifetime of network • Solves the energy hole problem	• Energy consumption problem is not resolved completely.
Clustering using Fuzzy logic [13-19]	• Avoids redundant data • Increases lifetime of network.	• It reduces number of messages
Clustering using Node Relevance [35]	• Balances node overhead • Suitable in remote and inaccessible areas.	• Size and timing of clustering is determined partly.
Distributed Fuzzy Logic based Energy aware and Coverage preserving Unequal Clustering [18]	• Provides energy efficiency • Controls the information passing in the network thereby reducing the energy	• Forms unequal sized clusters.
Optimal Cluster Size Selection-based Clustering protocol using Fuzzy Logic [37]	• Increases the lifetime of a network	• Clusters formed are of uneven size

Quad Clustering based on K-Means Algorithm [38]	• Reduces the consumption of energy	• It do not support multiple clusters.
Dragonfly Algorithm for Hierarchical Clustering [39]	• Saves Energy	• Optimization is not present
Energy Efficient Sleep Awake Aware Node Schedule Clustering [41]	• Improve the lifetime of a network by reducing energy utilization. • Supports more transmissions than usual.	• Optimization reduces flow of data.
Node Overhaul Scheme for Energy Efficient Clustering [44]	• Increases the lifetime of a network and also the node death rate.	• Do not work for unequal sized clusters.
Distance and Energy constrained K-Means Clustering Scheme [45-47]	• Increases the energy of network.	• Do not concentrate on coverage of nodes and consistency.

4. CONCLUSIONS

In this paper, the novel clustering techniques have been studied. This study reveals that Clustering algorithms follow various parameters such as distance, mobility, node position, etc. in the formation of clusters. Also, it has been found that each Clustering algorithm has its own advantages and disadvantages. So, these techniques have to be applied based on the requirement. This paper will provide a beneficiary source to all the researchers working in the domain of sensor networks.

5. ACKNOWLEDGEMENT

I would like to thank my supervisor Dr. R. Sridevi, Professor, Department of CSE, JNTU Hyderabad for providing her valuable timely suggestions in the preparation of this article and in guiding me in all aspects to carry out my research.

6. REFERENCES

[1] L. Yadav, Ch. Sunitha, "Low Energy Adaptive Clustering Hierarchy in Wireless Sensor Network (LEACH)", International Journal of Computer Science and Information Technologies, Vol. 5, No. 3, 2014, pp. 4661-4664.

[2] I. Daanoune, B. Abdennaceur, A. Ballouk. "A comprehensive survey on LEACH-based clustering routing protocols in Wireless Sensor Networks", Ad Hoc Networks, Vol. 114, 2021.

[3] M. Adnan, L. Yang, T. Ahmad, Y. Tao, "An Unequally Clustered Multi-hop Routing Protocol Based on Fuzzy Logic for Wireless Sensor Networks", IEEE Access, Vol. 9, 2021, pp. 38531-38545.

[4] A. Bereketli, O. B. Akan, "Event-to-Sink Directed Clustering in Wireless Sensor Networks", Proceedings of the IEEE Wireless Communications and Networking Conference, 2009, pp. 1-6,

[5] B. Baranidharan et al. "DUCF: Distributed load balancing Unequal Clustering in wireless sensor networks using Fuzzy approach", Applied Soft Computing, Vol. 40, 2016, pp. 495-506.

[6] B. Jan, H. Farman, H. Javed, B. Montrucchio, M. Khan, S. Ali, "Energy Efficient Hierarchical Clustering Approaches in Wireless Sensor Networks: A Survey", Wireless Communications and Mobile Computing, Vol. 2017, 6457942, 2017, 14 pages.

[7] N. Merabtine, D. Djenouri, D.-E. Zegour, "Towards Energy Efficient Clustering in Wireless Sensor Networks: A Comprehensive Review", IEEE Access, Vol. 9, 2021, pp. 92688-92705.

[8] R. Tandon, B. Dey, S. Nandi, "Weight based clustering in wireless sensor networks", Proceedings of the National Conference on Communications, 2013, pp. 1-5.

[9] S. Pradhan, K. Sharma, "Cluster Head Rotation in Wireless Sensor Network: A Simplified Approach", International Journal of Sensor and Its Applications for Control Systems, Vol. 4. 2016, pp. 1-10.

[10] M. C. M. Thein, T. Thein, "An Energy Efficient Cluster-Head Selection for Wireless Sensor Networks", Proceedings of the International Conference on Intelligent Systems, Modelling and Simulation, 2010, pp. 287-291.

[11] T. Jagannadha Swamy, G. Ramamurthy, P. Nayak, "Optimal, Secure Cluster Head Placement Through Source Coding Techniques in Wireless Sensor Networks", IEEE Communications Letters, Vol. 24, No. 2, 2020, pp. 443-446.

[12] M. Bala Krishna, M. N. Doja, "Self-organized energy conscious clustering protocol for wireless sensor networks", Proceedings of the 14th International Conference on Advanced Communication Technology, 2012, pp. 521-526.

[13] H. Bagci, A. Yazici, "An energy aware fuzzy approach to unequal clustering in wireless sensor networks", Applied Soft Computing, Vol. 13, No. 4, 2013, pp. 1741-1749.

[14] Z. Ullah et al. "Energy-Efficient Harvested-Aware Clustering and Cooperative Routing Protocol for WBAN (E-HARP)", IEEE Access, 2019.

[15] R. Logambigai, A. Kannan, "Fuzzy logic based unequal clustering for wireless sensor networks", Wireless Networks, Vol. 22, 2016, pp. 945-957.

[16] S. Rana, A. N. Bahar, N. Islam, J. Islam, "Fuzzy Based Energy Efficient Multiple Cluster Head Selection Routing Protocol for Wireless Sensor Networks", Computer Network and Information Security, Vol. 4, 2015, pp. 54-61.

[17] S, Mao, C, Zhao, Z, Zhou, Y, Ye, "An improved fuzzy unequal clustering algorithm for wireless sensor network", Proceedings of the 6th International ICST Conference on Communications and Networking in China, 2011, pp. 245-250.

[18] N. Mazumdar, H. Om, "Distributed fuzzy logic based energy-aware and coverage preserving unequal clustering algorithm for wireless sensor networks", International Journal of Communication Systems, 2017.

[19] M. Mirzaie, S. M. Mazinani, "AFLCH: Self-adaptive unequal fuzzy based clustering of heterogeneous sensors in wireless sensor networks", Proceedings of the 28th Iranian Conference on Electrical Engineering, 2020, pp. 1-5.

[20] J. Hu, Y. Pan, T. Li, Y. Yang, "TW-Co-MFC: Two-Level Weighted Collaborative Multi-view Fuzzy Clustering Based on Maximum Entropy", Proceedings of the Seventh International Conference on Advanced Cloud and Big Data, 2019, pp. 303-308.

[21] M. Singh, Gaurav, S. Soni, V. Kumar, "Clustering using fuzzy logic in wireless sensor networks", Proceedings of the 3rd International Conference on Computing for Sustainable Global Development, 2016, pp. 1669-1674.

[22] Z. Zeng-Wei, W. Zhao-Hui, L. Huai-Zhong, "An event-driven clustering routing algorithm for wireless sensor networks", Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2004, pp. 1802-1806.

[23] I. Akila, S. V. Manisekaran, R. Venkatesan, "Modern Clustering Techniques in Wireless Sensor Networks", InTech Open2017.

[24] N. Kalla, P. Parwekar, "A Study of Clustering Techniques for Wireless Sensor Networks", Satapathy, Smart Computing and Informatics. Smart Innovation, Systems and Technologies, Vol 77. Springer, Singapore, 2017.

[25] V. Loscri, G. Morabito, S. Marano, "A two-levels hierarchy for low-energy adaptive clustering hierarchy (TL-LEACH)", Proceedings of the IEEE 62nd Vehicular Technology Conference, 2005, pp. 1809-1813.

[26] B. Gong, L. Li, S. Wang, X. Zhou, "Multihop Routing Protocol with Unequal Clustering for Wireless Sensor Networks", Proceedings of the ISECS International Colloquium on Computing, Communication, Control, and Management, 2008, pp. 552-556.

[27] U. Hari, B. Ramachandran, C. Johnson, "An Unequally Clustered Multihop Routing protocol for Wireless Sensor Networks", Proceedings of the International Conference on Advances in Computing, Communications and Informatics, 2013, pp. 1007-1011.

[28] S. Bandyopadhyay, E. J. Coyle, "An energy efficient hierarchical clustering algorithm for wireless sensor networks", Proceedings of the Twenty-second Annual Joint Conference of the IEEE Computer and Communications Societies, 2003, pp. 1713-1723.

[29] S. G. Neogi, A. A. Bhaskar, P. Chakrabarti. "Energy Efficient Hierarchy-based Clustering Routing Protocol for Wireless Sensor Networks", International Journal of Computer Applications, Vol. 95, No. 13, 2014, pp. 1-8.

[30] J. Wang, Y. Yin, J.-U. Kim, S. Lee, C.-F. Lai, "A Mobile-Sink Based Energy-Efficient Clustering Algorithm for Wireless Sensor Networks", Proceedings of the IEEE 12th International Conference on Computer and Information Technology, 2012, pp. 678-683.

[31] T. K. Jain, D. S. Saini, S. V. Bhooshan, "Increasing Lifetime of a Wireless Sensor Network Using Multiple Sinks", Proceedings of the 11th International Conference on Information Technology: New Generations, 2014, pp. 616-619.

[32] S. Arjunan, S. Pothula, "A survey on unequal clustering protocols in Wireless Sensor Networks", Journal of King Saud University - Computer and Information Sciences, Vol. 31, No. 3, 2019, pp 304-317.

[33] B. Tang, J. Wang, X. Geng, Y. Zheng, J. Kim, "A Novel Data Retrieving Mechanism in Wireless Sensor Networks with Path-Limited Mobile Sink", International Journal of Grid and Distributed Computing, Vol. 5, No. 3, 2012.

[34] J.-S. Lee, C.-L. Teng, "An Enhanced Hierarchical Clustering Approach for Mobile Sensor Networks Using Fuzzy Inference Systems", IEEE Internet of Things Journal, Vol. 4, No. 4, 2017, pp. 1095-1103.

[35] M. Mirzaie, S. M. Mazinani, "MCFL: an energy efficient multi-clustering algorithm using fuzzy logic in wireless sensor network", Wireless Networks, Vol. 24, 2018, pp. 2251-2266.

[36] S. Gajjar, A. Talati, M. Sarkar, K. Dasgupta, "FUCP: Fuzzy based unequal clustering protocol for wireless sensor networks", Proceedings of the 39th National Systems Conference, 2015, pp. 1-6.

[37] P. K. Mishra, S. K. Verma, "OCSSP: Optimal Cluster Size Selection-based Clustering Protocol using Fuzzy Logic for Wireless Sensor Network", Proceedings of the IEEE International Conference on Advent Trends in Multidisciplinary Research and Innovation, 2020, pp. 1-4.

[38] B. Kumar, U. K. Tiwari, S. Kumar, "Energy Efficient Quad Clustering based on K-means Algorithm for Wireless Sensor Network", Proceedings of the Sixth International Conference on Parallel, Distributed and Grid Computing, 2020, pp. 73-77.

[39] B. Pitchaimanickam, "Dragonfly Algorithm for Hierarchical Clustering in Wireless Sensor Networks", Proceedings of the 5th International Conference on Intelligent Computing and Control Systems, 2021, pp. 192-197.

[40] 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, 2016, pp. 1053-1073.

[41] T. Shah, N. Javaid, T. N. Qureshi, "Energy Efficient Sleep Awake Aware (EESAA) intelligent Sensor Network routing protocol", Proceedings of the 15th International Multitopic Conference, 2012, pp. 317-322

[42] V. Kumar, V. K. Sinha, "Energy Efficient Node Schedule Clustering with Optimal Cluster in Wireless Sensor Network", Proceedings of the 2nd International Conference on Advances in Computing, Communication Control and Networking, 2020, pp. 363-367.

[43] M. Ashourian, M. Gheisar, A. H. Talkhoncheh, "An Improved Node Scheduling Scheme for Resilient Packet Ring Network", Majlesi Journal of Electrical Engineering, Vol. 9, No. 2, 2015, pp. 43-50.

[44] J. Singh, S. S. Yadav, V. Kanungo, Yogita, V. Pal, "A Node Overhaul Scheme for Energy Efficient Clustering in Wireless Sensor Networks", IEEE Sensors Letters, Vol. 5, No. 4, 2021, pp. 1-4.

[45] K. G. Omeke et al. "DEKCS: A Dynamic Clustering Protocol to Prolong Underwater Sensor Networks", IEEE Sensors Journal, Vol. 21, 2021. pp. 9457-9464.

[46] G. Y. Park, H. Kim, H. W. Jeong, H. Y. Youn, "A Novel Cluster Head Selection Method based on K-Means Algorithm for Energy Efficient Wireless Sensor Network", Proceedings of the 27th International Conference on Advanced Information Networking and Applications Workshops, 2013, pp. 910-915.

[47] M. Bidaki, R. Ghaemi, K. Tabbakh, "Towards Energy Efficient k-MEANS Based Clustering Scheme for Wireless Sensor Networks", International Journal of Grid and Distributed Computing, 2016, pp. 265-276.

[48] R. Logambigai, A. Kannan, "Fuzzy logic based unequal clustering for wireless sensor networks", Wireless Networks, Vol. 22, 2016, pp. 945-957.

Sum Rate Maximization and Consistency in D2D Communication Based on ACO and Game Theory

Original Scientific Paper

Amel Austine
Research Scholar, Department of Computer Science and Engineering,
Noorul Islam Centre for Higher Education, Tamil Nadu, India
amelaustine.noorul.in@gmail.com

Suji Pramila R
Associate Professor, Department of Computer Science and Engineering,
Mar Baselios Institute of Technology and Science, Kerala, India
sujipramila.in@gmail.com

Abstract *– Cellular network is the most popular network setup among today's wireless communication systems. The primary resource in a cellular system is the spectrum for communication, and owing to the rising number of cellular users, the spectrum that is currently accessible from different service providers is depleting quickly. The resource or channel allocation is the most hindering task in cellular networks. Many efforts have been taken by many researchers to allocate the resources properly in order to increase the channel utilization and it is found that one effective method for reusing the channels inside a cell is device to device (D2D) communication. D2D communication was first developed in order to achieve the fundamental goals of fast data rates, widespread coverage with little latency, energy efficiency, and low per-information transmission costs. The dynamic behaviour of this network set-up again increases the risk of different types of interferences, which is another issue faced by the researchers. In this paper an effort is taken to understand and solve various aspects of channel allocation and Cellular networks have incorporated interference management in D2D communication especially. The two major issues of allocation of resource and management of interference in D2D communication is addressed here. This paper considers the meta heuristic algorithm namely Ant Colony Optimization (ACO) for resource allocation issue and interference management. The sum rate maximization is achieved through Game theory along with the concept of resource exchange in turn to increase the consistency of D2D communication setup. The results demonstrate that our algorithm can significantly increase the sum rate of D2D pairs when compared to other algorithms suggested by related works.*

Keywords*: Cellular Network; D2D Communication; Interference Management; ACO, Game Theory research exchange*

1. INTRODUCTION

In the current era, the cellular system is the point of interest for mobile wireless communication. It is the underlying technology for various requirements like mobile telephones, wireless Internet, various wireless webs, other applications, and much more in mobile wireless communication. It is a wireless technology in which several small geographical areas are equipped with low-power radio antennas known as base stations (BS) [1]. This geographical area under a base station is known as a cell. There will be a central unit to interconnect these antennas, which is known as a Mobile Switching Centre (MSC).

In any wireless system, radio bandwidth is distributed as channels. From this set, only a limited number of channels (radio spectrum) are provided to a service provider. A single channel or block of channels can be allotted to carry out communication between two parties. These channels are usually allotted to a certain cell. The same channel can be reused in different cells that are away from the current cell, ensuring the distance from the current cell is greater than a predefined reusable distance. Channels can be allocated to a cell based on three standard schemes [2]. Every cell in the case of Fixed Channel Allocation (FCA) is given a set number of channels, while in the case of Dynamic Channel Allocation (DCA), channels are assigned to a new cell whenever one is needed from a central repository of channels, and in the case of Hybrid Channel Allocation (HCA), a combination of both FCA and DCA is used. While choosing an allocation technique, certain points must be kept in mind. In a cellular system, it is expected to support diverse services that guarantee various levels of quality. Advanced bandwidth reservation must be performed in order to facilitate and support an uninterrupted communication link through seamless handoff. Finally, the provision of bandwidth

reconfiguration depends on the various dynamic cellular environments.

The major challenge related to resource management in the cellular system is the allocation of channels in various cells with little or no interference. Wireless systems are affected with two types of interferences.

ACI and co-channel interference are examples of adjacent channel interference (CCI). Signal-based introduction of ACI, which are adjacent in frequency, when allocated to cell Equipment limitations are the major source of ACI, such as receiver bandwidth, frequency instability, and imperfect filtering. CCI is introduced when the same signals are assigned to different cells close to each other. Figure 1 illustrates the fundamental resource management elements of D2D communication.

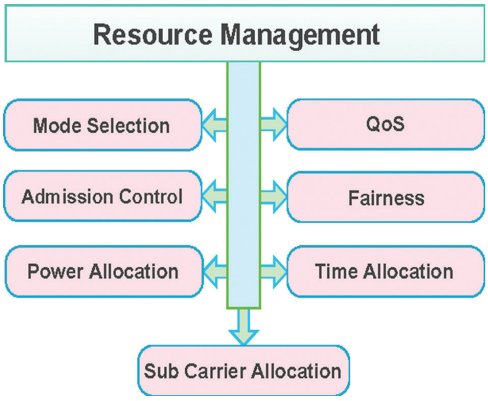


Fig. 1. Resource Management elements in D2D Communication

The service demand for proximity-based services has received increased attention from the Third-Generation Partnership Project (3GPP) (ProSe) [3] for LTE-Advanced. Proximity-based services (ProSe), which require components to be physically close to each other in order to recognise each other and directly communicate, are broadly described in 3GPP Rel. 12 of the LTE-Advanced standard. ProSe aims to advance public security [4]. They started the standardisation procedure for the ProSe radio access network recently. Figure 1 displays a very basic representation of D2D communication relying on the architecture of ProSe.

With 3GPP Release 12, D2D communication was first incorporated into LTE-A. The FCC assigned 700 MHz of the reserved spectrum to this release, which the US's next-generation public safety network has adopted [5].

To use the high bit rate services of today's cellular network, this framework potentially decreases the data rate as well as increases the delay. D2D Communication [6] is the best candidate for it. D2D communication enables mobile devices in close proximity to communicate directly without the involvement of a Base Station (eNB in LTE). It can reuse the channels, already allocated to cellular users, that is, better reuse of the available spectrum. The cell boundaries and areas with weaker signals are served better by this mechanism.

The major benefits identified for device-to-device communication are; one hop communication, spectrum reusability, Higher and better network performance, reduced power consumption, congestion control, fairness, and QoS guarantees [7]. Users have complete control over D2D communication [8].

The first step in establishing D2D links is, identifying and pairing user devices that are at a distance enough to have direct communication. This is achieved during peer device discovery stage. The user's devices will identify their neighbours for direct communication link establishment.

A mathematical framework of scientific tools called "game theory" examines how an individual or group of players will behave and move. In other words, it serves as a decision-making tool for rational players [9]. Because game-theoretic solutions are manifestly self-directed and reliable, they are required for reducing the interference that D2D and cellular broadcasts could produce with one another in cellular networks.

Numerous combinatorial optimisation problems can be resolved using ant colony optimisation (ACO) algorithms. Most of the previous works used heuristic or partial optimal based algorithms. Through simulation experiments, it is possible to see how well these methods perform, but it is still difficult to put them into practise in real-world industrial settings. There are primarily two parts:

Due to the unique user distribution and geographic environment, it is challenging to create a perfect mathematical model for practical implementation scenarios. The computational complexities of these approaches are quite high, especially in D2D communication systems. Implementing these algorithms with interference optimization thus becomes impractical.

The most popular random search techniques to solve intractable problems are metaheuristic algorithms. In practise, metaheuristic algorithms are compliant with optimization and do not necessitate a perfect mathematical model. Modern methods for resolving issues with resource allocation and interference management and power distribution in a D2D system are discussed.

In this paper, examine the communication network under the influence of multiple interfering channels. To increase the overall sum-rate in the D2D system mainly focuses on the metaheuristic optimization for the resource allocation issues. The simulation results show which algorithm performs better than the other metaheuristic algorithms.

The contribution of our research work is mentioned:

- Demonstrate an optimal resource allocation issue under interference management and propose meta heuristic approaches.
- Through numerical experiments on simulated communication networks and wireless channels,

validate the efficiency and adaptability of the proposed metaheuristic methods and search for the potentially best algorithms for this issue.

The following is the paper's structure. In Section 2, the most recent D2D communication research literature rview is described. Section 3, states the problem formulation and findings of D2D and section 4 explains the proposed methodology. Section 5 showcases the findings and outcome are evaluated and finally in Section 6, the whole term paper is concluded.

2. RELATED WORKS

While establishing and maintaining D2D mechanism in standard wireless cellular networks the following challenges need to be addressed effectively. Peer Discovery, Mode Selection and Resource Allocation with Interference Management.

Han et al. [10] put forth a brand-new D2D communication strategy based on socially conscious peer discovery. Here the characteristics, of social network are exploited in an ad hoc peer discovery mechanism. The devices send a well-known synchronization or reference signal sequence for peer finding. To improve D2D communication, Yanru Zhang et al. [11] suggested a novel strategy that takes into account socially conscious factors. The physical wireless network layer and the social network layer, two distinct layers, are used to accomplish it. There is potential for stable D2D connections, by defining the OffSN using the connection degree of users, basis on their daily routes. The mechanism considers data security as well.

Quinghe Du et al. [12] proposed a D2D routing design for an underlying cellular network with interference controlled to support multi-hop D2D transmission. This is an innovative idea, as most of the works were single hop in nature. The main objective of routing is to diminish the end-to-end delay by reducing the hop count and in turn reduces the power consumption for D2D connections.

Xiaoqiang Ma et al. [13] proposed application specific D2D communication network management. They considered two types of applications like streaming and file sharing. Their main objective was to increase the rate of aggregated data. It is noted that careful consideration must be given to the power distribution for both cellular and D2D lines. Yanxiang Jiang et al. [14] proposed an efficient scheme for iterative resource allocation with competent power control. Resources blocks are used by CUs and D2D pairs. The channel gain of all the subcarriers incorporated in the same RB will be the identical. At this point the power control problem is found to be more practical. A non-orthogonal dynamic spectrum sharing mechanism was proposed by Tuong Duc Hoang et al. [15] for D2D communication in a cellular network below. A graph-based approach is used for the formulation of a subband assignment problem. They considered the scenario where single/multiple subbands are allocated for each active D2D link, and at most one D2D link can be allocated by this subband.

Jun Hunang et al. [16] addressed a new scenario. They specifically looked into the problems with resource distribution for a D2D link placed in the overlap or boundary regions of two cells close by. Chih-Yu Wang et al. [17] proposed a trader assisted resource ex-change mechanism for an efficient resource allocation. In this approach a sequence of enthusiastic resource block groups is predefined for D2D communication. Tinghan Yang et al. [18] employed the D2D communication concept in an advanced scenario of full-duplex cellular network. Here both cellular uplink and downlink are used by the Ues to be in touch with each other.

Albowarab et al. [19] proposed a novel approach named as ROOMMATEs for peer discovery in an indoor environment. Compared to the outdoor scenario, the peer discovery in an indoor environment is challenging. It uses WiFi/other wireless signals to locate the UEs. The location tags are used for peer discovery and pairing if two UEs are in the same place. Energy consumption is the major goal of this proposal. Kar et al. [20] developed a mechanism to discover a potential partner. They utilised sounding reference signal (SRS) to listen the uplink channels, which is present in the OFDM symbol's last part of each scheduling sub-frame. Even if it is a simple method as the SRS is readily available in the cellular network, the major difficulty is in the design of the SRS receiver, because of the uplink overhearing behaviour of devices.

AmalAustine et al. [21] proposed a distributed, user-controlled approach. Here is a possible D2D UE chooses a mode on its own, taking into account performance and cost. Managing the reuse mode is difficult in this environment. To increase the fairness of the system, Miaomiao Liu et al. [22] suggested a novel resource allocation technique. Fairness and throughput are compromised in order to guarantee that each user has equal access to system resources. Based on the results of the recommended allocation technique, they created the notion of virtual D2D linkages and framed the channel allocation problem as a 3-D assignment problem. By applying the Kuhn-Munkres method to a particular 2-D assignment problem, they were able to find a solution.

3. PROBLEM FORMULATION AND FINDINGS

Various literatures addressing the major challenges in D2D are addressed in this work. The new technology will become a major enhancement in LTE towards the 5G release. All the work was simulated using various tools, and it was found that it can contribute a great deal to the evolution of wireless communication. The major focus is on improving the spectrum efficiency of cellular network. Apart from that it is found that this technology can improve the throughput, sum rate and many other aspects because it is properly managing the interferences created when reusing the channels in the same cell.

A detailed comparison is made based on a wide variety of aspects of D2D communication in cellular system. These aspects are clearly represented in Table I. Three potential scenarios were taken into account, with Scenario-I outlining a user-oriented strategy in which each D2D link is given a sub band, with a maximum of one D2D link being given this sub band. In Scenario-II, a resource-oriented method is described, where each active D2D connection is given access to several subbands, with each subband being assigned to a maximum of one D2D link.

The spectral efficiency is very much affected by the mode selection, especially the D2D modes. Even if the overlay mode is a good mitigation for managing interference in cellular and D2D systems, it reduces spectral efficiency. Researchers suggested a number of ways to implement D2D using underlay mode to manage the interference in an optimised manner. From the literature it is seen that the interference management in underlay networks were managed by proper power control. Usually, different link power is used for cellular and D2D link. The cellular link will have a higher power allocation compared to the D2D link in order to minimise the interference caused by the D2D link to a cellular link, when both are using the same channel [23].

Research spanned from the basic single cell scenario to multi cell scenario for D2D communication. It is very useful to provide good service to the user equipment's lying at the cell boundary, where usually the nearby cells overlap and signals from both the cells will be available. The major obstacle in this is the proper handoff management. We cannot expect both the devices in D2D communication will move to the same cell. The selection of channels in the new cell also needs to repeat the resource allocation steps in-order to resume D2D communication.

In most of the work, the allocation of resources is centralised. It is more effective than ad hoc manner as the eNB holds more information about the network status. If any issues regarding device distance or interference occur, the service will toggle back to the infrastructure approach. In a dynamic environment like a cellular system this switching is not feasible. The D2D links will be no longer reliable if it does not consider the dynamic aspects of cellular system. But some recent literatures address that dynamic re-allocation can improve the sustainability of D2D communication and thus the spectral efficiency. Exchanging resources can significantly improve the life of D2D links. Another aspect is the cooperative communication between devices. This will clearly ensure the reliability of the link between devices. As the distance between the devices increases, an intermediate node will act as a relay to the destination device. The drawback is the increased number of hops in communication.

Table 1. Summary of State-of-the-Art Research works

Ref. No	Scenario	Approach	Objective	Mpdal	Multi-D2D& Cellular Links	Theoretical Performance Analysis	Optimal Solution	Single Cell	Allocation	Dynamic Reallocation	Mode
[4]	ad hoc	Optimization	Peer Discovery	PA	No	Yes	Yes	Yes	Network	Static	1H, BS
[5]	ad hoc	Optimization	Neighbor Discovery	NA	No	Yes	Yes	Yes	Ad Hoc	Static	1H, BS
[6]	ad hoc	Optimization	Neighbor Discovery	PA	No	No	Yes	Yes	Ad Hoc	Static	1H, BS
[7]	II	Optimization	Joint mode election& spectrum partitioning	SA	Yes	Yes	Yes	Yes	Ad Hoc & Network	Dynamic	1H, BS
[8]	II	Optimization	Spectrum Sharing	SA, PA	Yes	Yes	Yes	Yes	Network	Static	1H, BS
[9]	II	Optimization	Improve packet transmission, reduce network load.	Yes	Yes	Yes	Yes	Yes	Network	Static	1H, BS
[10]	II	Optimization	System throughput and fairness	PA	Yes	Yes	Yes	Yes	Network	Dynamic	IH, BS
[11]	II	Optimization	Mode Selection, Packet Scheduling	RBA	Yes	No	Yes	Yes	Network	Static	1H, BS
[12]	II	Optimization	Aggregated Data Rate, Weighted Cell Utility	PA	Yes	Yes	Yes	Yes	Network	Static	1H, BS
[13]	I	Optimization	Spectrum and Energy Efficiency	RBA, PA	Yes	Yes	Yes	Yes	Network	Static	1H, BS
[14]	I, II	Graph based and Optimization	Weighted Sum Rate	SA, PA	Yes	Yes	Yes	Yes	Network	Static	1H, BS
[15]	I, II	Game theoretic Optimization	BS Sum rate, sum rate gain	SA, PA	Yes	Yes	Yes	No	Network	Static	1H, BS

[16]	I, II, III	Optimization, Graph Coloring	Network Performance and spectrum utilization	RBA, PA	Yes	Yes	Yes	Yes	Network	Static	1H, 2H, BS
[17]	I, II, III	Optimization, Graph Based	Spectral Efficiency	RBA	Yes	Yes	No	Yes	Network	Dynamic	1H, BS
[18]	I	Multi hop route optimization	hop count minimization	SA	Yes	Yes	No	Yes	Ad Hoc	Static	1H, 2H, BS

There are some recent techniques that have been proposed to minimise the hop count and thus the delay of direct communication between devices. Increasing the number of links utilising same resource will obviously increase the spectral efficiency. Most of the recent works address this aspect in a decent manner. According to the study performed on various literature few common aspects are considered and are represented in Table 1.

4. PROPOSED METHODOLOGY

The proposed work was on the basis of our previous work, which combined Ant Colony Algorithm for better resource allocation and the resource exchange mechanism in order to improve the consistency of the D2D link in the cellular system. Apart from reducing the interference in the organization due to the reuse of frequency, another add on mechanism is proposed in order to maximize the sum rate in the system using a Game theory mechanism, as shown in the Fig. 2.

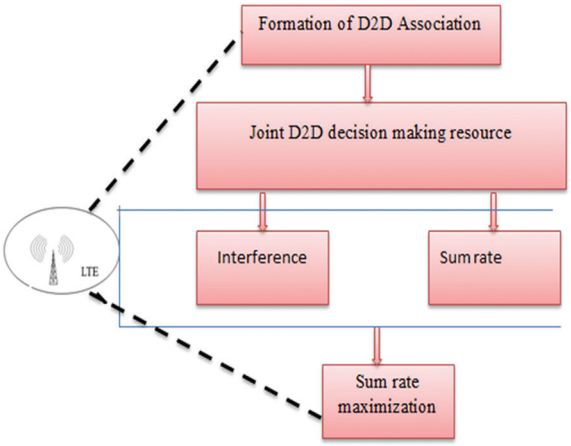


Fig. 2. Framework for the System

The interference caused by the increasing number of D2D pairs in a cell and the number of RBGs assigned to a cell to initiate the resource exchange are the two criteria that are taken into consideration. The suggested D2D resource exchange framework introduces the Trading-based Resource Exchange (TREX) method to address the resource exchange issue.

4.1 RESOURCE EXCHANGE

Resource exchange can make a link between two directly connected devices more reliable. The eNodeB must allocate resources after assessing the cell's present condition. The mechanism for resource selection is not yet known. The goal of this work is to sustain D2D linkages with the least amount of interference and the longest possible lifetime while considering the likelihood of introducing resource exchange to a cellular system.

When an additional D2D pair joins and seeks an RBG, or a D2D pair's RBG contribution expires, and one or more D2D pairs renew their CQIs, the eNodeB may initiate a resource exchange. The revised D2D pair set or CQIs may contain the favourable exchange sequence whenever the aforementioned circumstances arise. If a sequential exchange is discovered, by issuing the most recent RBG grants to the pair heads, the eNodeB can force all pairs in the series to trade resources. The alerted D2D pairs will accept and use the newly authorised RBG in the subsequent transmission. After the exchange process is complete, both D2D pairs will use the recently swapped RBGs for their respective D2D communications. Additionally, an exchange between an eNodeB and a D2D pair is feasible. The eNodeB may give this D2D pair an unallocated RBG in exchange for the RBG it currently owns. In this case, just one D2D pair is informed by the resource sending the signal.

When eNodeB is down or the traffic is slowdown, the D2D pair can also dynamically initiate a resource swap in delay-sensitive applications. All D2D pairs can communicate with one another thanks to the maintenance of this shared RBG. D2D pairings should pay attention to the shared RBG in order to hear requests from other D2D pairs. Every 3n+1 sub frame, all D2D couples may use a slotted-ALOHA technique to access the shared RBG. A D2D pair head can transmit its own and necessary RBGs to other nearby D2D pairs in the same service area by using the shared RBG. If the other D2D pair head who is in possession of the requested RBG is in favour of the exchange as well, they may answer the demand in the subsequent subframe of the collective RBG. In the third subframe, after receiving an acknowledgement from the new D2D pair head, both pairs switch over to the newly traded RBGs and can begin using them for D2D broadcasts. Both pairings should inform the eNodeB about the exchange following it so that it can keep track of the allocation of RBG positions.

The aforementioned mode lessens the eNodeB's load by shifting various duties from the eNodeB to the D2D pairs. Additionally, it decreases the time required for eNodeB-triggered exchanges and CQI reporting

signals. Additionally, relatively stable channels and surroundings are preferred. The eNodeB can further reduce the overhead in the case of channels that are comparatively stable, or overloaded networks by mandating that all D2D resource exchanges operate in D2D-triggered mode.

A centralised resource exchange system called T-REX run on the eNodeB. According to the gathered CQIs, In the D2D service zone, it established the resource exchange sequences for each D2D pair. The CQIs from each D2D pair within a service region are first collected using the T-REX approach. The preference of each D2D pair on RBGs is then developed in accordance with the available CQIs. The T-REX technique then builds an exchange graph using D2D pair preferences. By looking for cycles in the trade graph, the exchange order is in turn determined by the mechanism. Then, it is requested that all D2D pairs and traders involved carry out resource exchanges in accordance with the chosen exchange order. A D2D pair encounters differing levels of interference in different RBGs. The more interference-free an RBG is, the more useful the D2D pair.

4.2 D2D INTERFERENCES GRAPH REPRESENTATION

The eNB initially creates a fully linked weighted graph as an inner model of the mutual interference conditions among contemporaneous D2D links (as if all were allotted the same resources). The data that the UEs submitted to the eNB was used to construct this graph. Particularly, each UE reports details regarding the received power resulting from whichever possible corresponding D2D transmission. The UEs may be able to obtain this interference information by using methods like the D2D peer discovery approach, which was not examined in this study. Figure 3 illustrates how the eNB graphs the acquired interference levels.

The edges are weighted, and depending on the actual channel circumstances, these weights correspond to the degree of mutual interference among a few potentially corresponding D2D transmissions. We introduce the Interference Level Indicator (ILI) concept to quantify this. ILI accepted values on an interval scale with customizable minimum and maximum values. The scale's min/max values reflect the estimated interference's min/max in the present topology, while all values in between are determined equally.

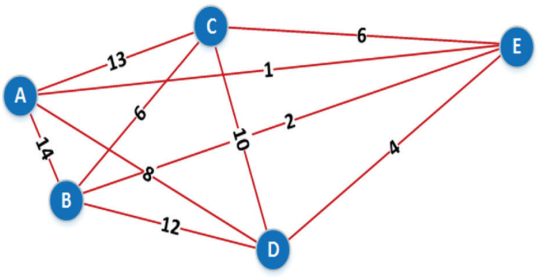


Fig. 3. D2D pairs Interference graph

A scale can thus depict 100 different levels of interference, A [0-1] scale, however, merely shows the presence or absence of interference. The mapping of interference values to ILIs is done by the eNB. Since all vertexes can interact with one another, the resulting graph is fully connected when zero values are removed from the equation, either significantly or barely. Consequently, in this form, the likelihood of safe spectrum sharing increases with decreasing node weights due to reduced mutual interference.

In the network graph model, nodes A through E stand in for five D2D requests, whereas nodes 1–15 represent ILIs. These concurrent D2D queries are close together in this scenario since the ILI linking vertexes A and B is 14, since the ILI connecting A and E is 1, indicating that they are far apart and most likely not interfering with one another at all.

5. RESULT AND DISCUSSION

In NS3, the planned method is modelled. The outcomes of the simulation are contrasted with three other comparable methods that used resource exchange techniques. They were built on greedy cycle-complete preferences and the Trader-Assisted Resource Exchange (TREX) technique with two strategy proof preferences (RAN and DRAN) (CYC). First, we find the number of D2D pairs that affect the interference of the T-REX mechanism. We maintain 15 RBGs and increase the number of D2D pairs in the simulation from 6 to 15.

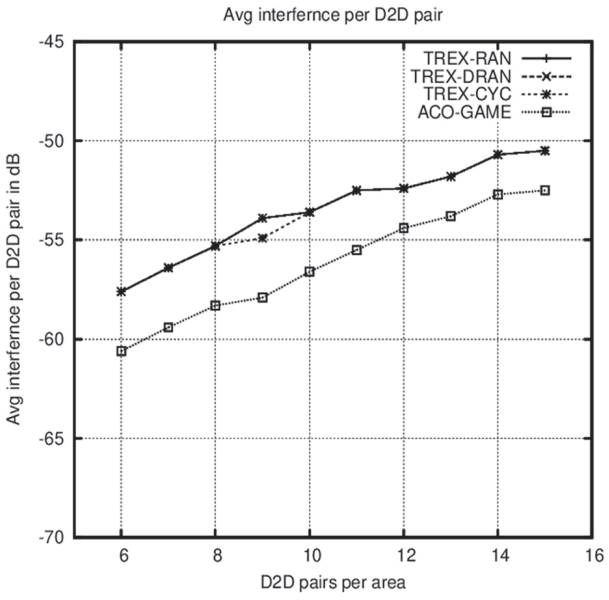


Fig. 4. Interference Vs D2D pairs per area

Fig. 4 shows that when the number of D2D pairs grows, the interference level rises for all T-REX routes. Because there are fewer RBGs available for exchange, there are fewer opportunities for improvement through the exchange. The BS's reduced ability to produce RBGs is what led to this outcome.

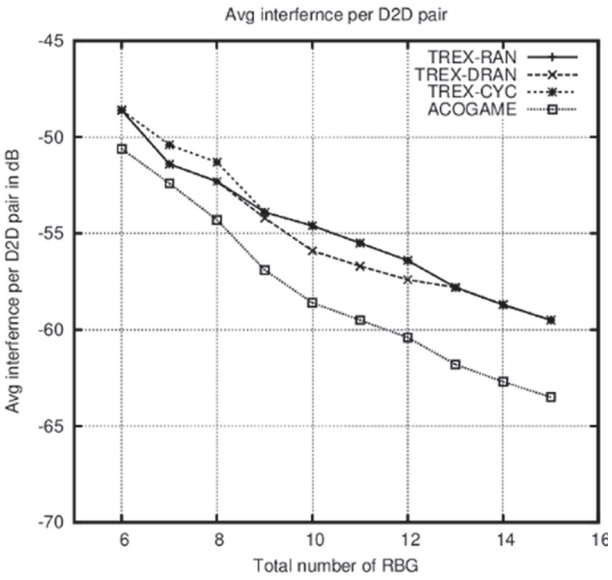


Fig. 5. Interference Vs Total number of RBG

Next, we simulate using six D2D pairings while increasing the amount of RBGs accessible from six to fifteen. The outcomes are displayed in Figure. 5. We discover that as the number of RBGs shown increases, interference in all schemes decreases and the interference mitigation provided by the T-REX mechanism in the Random scheme increases.

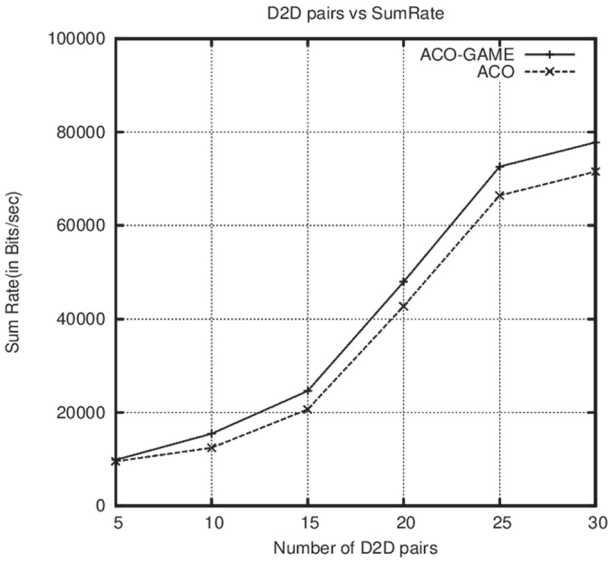


Fig. 6. Number of D2D pairs Vs Sum Rate

The system sum rate vs the quantity of D2D pairs is depicted in Figure 6. It is evident that as there are more D2D pairs, the sum rate of both schemes increases monotonically. However, because there are a finite number of subcarriers and a finite amount of bandwidth, increasing the D2D user base could result in more co-channel interference between the D2D link and the cellular link, which the system might not be able to adequately mitigate. The best performance was obtained with the proposed approach we used with ACO-Game Theory.

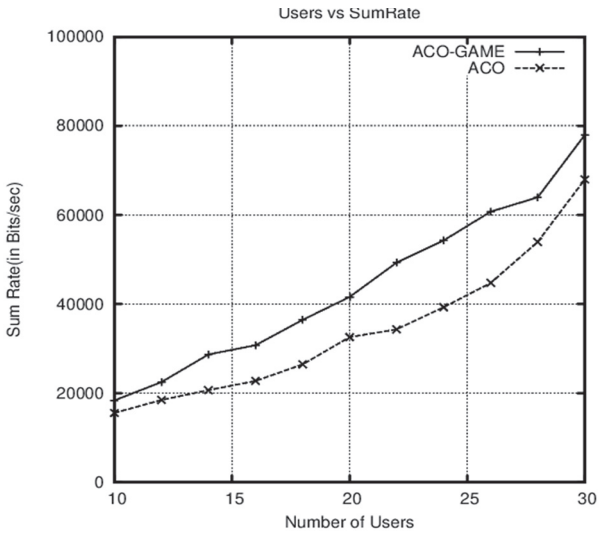


Fig. 7. Number of users Vs Sum rate

Fig. 7 displays various sum rates attained with various D2D user counts. As more D2D users join the network, it is evident that the network's sum rate rises as well. As a result, Ant Colony Optimization can maintain more D2D linkages with little disruption.

6. CONCLUSION

A thorough analysis of current research on device-to-device communication in cellular networks is carried out. According to the main difficulties in D2D communication, such as peer discovery, mode selection, resource allocation with interference, and power management, we classify the currently done work. In this paper, the sum rate maximisation of D2D pairs using ACO-Game theory, when compared to ACO optimisation algorithms, achieved better results in terms of number of users and D2D area. The strengths and weaknesses of this literature are assessed, and a few recommendations for improvements are suggested. The application of Game theory along with Ant Colony Optimisation and resource exchange frameworks can produce considerable optimisation in D2D communication in a cellular network. In this case, an attempt is made to use a resource-sharing mechanism to keep the D2D link alive for as long as is practical. When cellular or D2D link interference threatens another D2D link, swapping takes place. By extending a D2D link's lifespan, this tactic can improve the network's stability.

7. REFERENCES

[1] J. Schiller, "Mobile Communications", Second Edition, Pearson Education Limited, 2003, pp. 61-64

[2] S. Glisic, J. P. Makela, "Advanced wireless networks: 4G technologies", Proceedings of the IEEE Ninth International Symposium on Spread Spectrum Techniques and Applications, Manaus, Brazil, 28-31 August 2006, pp. 442-446.

[3] "Device to Device Communication in LTE", Rohde & Schwartz Whitepaper, Rohde & Schwarz USA, Inc., Columbia, MD, USA, 2015.

[4] N. T. Nguyen, K. W. Choi, L. Song, Z. Han, "ROOMMATEs: An unsupervised indoor peer discovery approach for LTE D2D communications", IEEE Transactions on Vehicular Technology, Vol. 67, No. 6, 2018, pp. 5069-5083.

[5] H. Tang, Z. Ding, B. C. Levy, "Enabling D2D communications through neighbor discovery in LTE cellular networks", IEEE Transactions on Signal Processing, Vol. 62, No. 19, 2014, pp. 5157-5170.

[6] B. Zhang, Y. Li, D. Jin, P. Hui, Z. Han, "Social-aware peer discovery for D2D communications underlaying cellular networks", IEEE Transactions on Wireless Communications, Vol. 14, No. 5, 2014, pp. 2426-2439.

[7] K. Zhu, E. Hossain, "Joint mode selection and spectrum partitioning for device-to-device communication: A dynamic Stackelberg game", IEEE Transactions on Wireless Communications, Vol. 14, No. 3, 2014, pp. 1406-1420.

[8] X. Lin, J. G. Andrews, A. Ghosh, "Spectrum sharing for device-to-device communication in cellular networks", IEEE Transactions on Wireless Communications, Vol. 13, No. 12, 2014, pp. 6727-6740.

[9] G. Nardini, G. Stea, A. Virdis, D. Sabella, M. Caretti, "Resource allocation for network-controlled device-to-device communications in LTE-Advanced", Wireless Networks, Vol. 23, 2017, pp. 787-804.

[10] Z. Han, "Game theory in wireless and communication networks: theory, models, and applications", Cambridge University Press, 2012.

[11] Y. Zhang, E. Pan, L. Song, W. Saad, Z. Dawy, Z. Han, "Social network aware device-to-device communication in wireless networks", IEEE Transactions on Wireless Communications, Vol. 14, No. 1, 2014, pp. 177-190.

[12] Q. Du, H. Song, Q. Xu, P. Ren, L. Sun, "Interference-controlled D2D routing aided by knowledge extraction at cellular infrastructure towards ubiquitous CPS", Personal and Ubiquitous Computing, Vol. 19, 2015, pp. 1033-1043.

[13] X. Ma, J. Liu, H. Jiang, "Resource allocation for heterogeneous applications with device-to-device communication underlaying cellular networks", IEEE Journal on Selected Areas in Communications, Vol. 34, No. 1, 2015, pp. 15-26.

[14] Y. Jiang, Q. Liu, F. Zheng, X. Gao, X. You, "Energy-efficient joint resource allocation and power control for D2D communications", IEEE Transactions on Vehicular Technology, Vol. 65, No. 8, 2015, pp. 6119-6127.

[15] T. D. Hoang, L. B. Le, T. Le-Ngoc, "Resource allocation for D2D communication underlaid cellular networks using graph-based approach", IEEE Transactions on Wireless Communications, Vol. 15, No. 10, 2016, pp. 7099-7113.

[16] J. Huang, Y. Yin, Y. Zhao, Q. Duan, W. Wang, S. Yu, "A game-theoretic resource allocation approach for intercell device-to-device communications in cellular networks", IEEE Transactions on Emerging Topics in Computing, Vol. 4, No. 4, 2014, pp. 475-486.

[17] C. Y. Wang, G. Y. Lin, C. C. Chou, C. W.Yeh, H. Y. Wei, "Device-to-device communication in LTE-advanced system: A strategy-proof resource exchange framework", IEEE Transactions on Vehicular Technology, Vol. 65, No. 12, 2016, pp. 10022-10036.

[18] T. Yang, R. Zhang, X. Cheng, L. Yang, "Graph coloring based resource sharing (GCRS) scheme for D2D communications underlaying full-duplex cellular networks", IEEE Transactions on Vehicular Technology, Vol. 66, No. 8, 2017, pp. 7506-7517.

[19] M. H. Al-Bowarab, N. A. Zakaria, Z. Z. Abidin, Z. K. Maseer, "Review on device-to-device communication in cellular based network systems", International Journal of Engineering & Technology, Vol. 7, No. 3.20, 2018, p. 435.

[20] U. N. Kar, D. K. Sanyal, "An overview of device-to-device communication in cellular networks", ICT Express, Vol. 4, No. 4, 2018, pp. 203-208.

[21] A. Austine, R. S. Pramila, "Interference Management by Resource Exchange for D2D Communication in Cellular Network", International Journal of Recent Technology and Engineering, Vol. 8, No. 5S, 2020.

[22] M. Liu, L. Zhang, "Resource allocation for D2D underlay communications with proportional fairness using iterative-based approach", IEEE Access, Vol. 8, 2020, pp. 143787-143801.

[23] L. Song, D. Niyato, Z. Han, E. Hossain, "Game-theoretic resource allocation methods for device-to-device communication", IEEE Wireless Communications, Vol. 21, No. 3, 2014, pp. 136-144.

A Lightweight Island Model for the Genetic Algorithm over GPGPU

Original Scientific Paper

Mohammad Alraslan
Idlib University, Faculty of Informatics Engineering, Department of Software Engineering
Idlib, Syria
Mohammad-Alraslan@idlib-university.com

Ahmad Hilal AlKurdi
Idlib University, Faculty of Informatics Engineering, Department of Software Engineering
Idlib, Syria
Ahmad-Hilal-AlKurdi@idlib-university.com

Abstract *– This paper presents a parallel approach of the genetic algorithm (GA) over the Graphical Processing Unit (GPU) to solve the Traveling Salesman Problem (TSP). Since the earlier studies did not focus on implementing the island model in a persistent way, this paper introduces an approach, named Lightweight Island Model (LIM), that aims to implement the concept of persistent threads in the island model of the genetic algorithm. For that, we present the implementation details to convert the traditional island model, which is separated into multiple kernels, into a computing paradigm based on a persistent kernel. Many synchronization techniques, including cooperative groups and implicit synchronization, are discussed to reduce the CPU-GPU interaction that existed in the traditional island model. A new parallelization strategy is presented for distributing the work among live threads during the selection and crossover steps. The GPU configurations that lead to the best possible performance are also determined. The introduced approach will be compared, in terms of speedup and solution quality, with the traditional island model (TIM) as well as with related works that concentrated on suggesting a lighter version of the master-slave model, including switching among kernels (SAK) and scheduled light kernel (SLK) approaches. The results show that the new approach can increase the speed-up to 27x over serial CPU, 4.5x over the traditional island model, and up to 1.5–2x over SAK and SLK approaches.*

Keywords*: GPGPU, Genetic algorithm, TSP, Island Model, Speed up*

1. INTRODUCTION

Nowadays, GPUs (Graphics Processing Units) are playing a significant role in general-purpose computing. Applications for engineering and research are accelerated by GPUs' capabilities in various scientific domains. Nvidia introduced CUDA (Computer-Unified Device Architecture) in 2007 as a general-purpose parallel computing API [1, 2]. Programmers can efficiently solve computational issues by utilizing the GPU's parallel architecture using CUDA. These days, laptops are equipped with powerful GPUs that have thousands of cores [3]. These reasons motivated the researchers to develop GPU-based parallel applications.

The traveling salesman problem (TSP) is a complex combinatorial optimization problem that has been used to solve many problems, like UAV path planning [4, 5]. A popular evolutionary algorithm for solving the TSP problem is the genetic algorithm (GA) [6]. The genetic algorithm is an iterative process that requires a lot of computation. To speed up the GA process, many studies have developed parallel approaches over the GPU. They relied on the basic models of the parallel genetic algorithm, including the master-slave model, the island model, and the cellular model. They developed them to suit the studied problem and decrease the execution time [7-9].

In the traditional master-slave model, each step of the genetic algorithm was mapped to a separate kernel. Performing the evolving process, in a single iteration, involves calling these kernels. There is much CPU-GPU communication overhead to call the kernels from the host in every iteration. The master-slave model based on persistent threads was suggested to collapse the multi-kernel into one kernel and keep the threads alive throughout the execution of all iterations in the genetic algorithm [10]. This approach reduces CPU-GPU communication because there is only one kernel call, and iterations are made within the kernel.

Calling a single kernel means that we will pass the number of participating threads (Grid and Block sizes) only once in order to execute the genetic algorithm. For this reason, a method was adopted to distribute the

work between these active and live threads to achieve the best possible performance.

In the traditional island model (TIM), the large population is divided into small subpopulations (islands). Each iteration is achieved with only one kernel call. This method also introduces a CPU-GPU communication overhead.

Converting this model to a method based on persistent threads requires discussing two issues: the synchronization mechanisms and the work distribution between live threads, whose number will be fixed throughout the kernel's execution.

In this paper, we focus particularly on how to convert the island model of the genetic algorithm into a new approach named the Lightweight Island Model (LIM) that follows the concept of persistent threads. For that, we highlight the earlier studies that worked on converting the traditional master-slave model into a lighter version by the number of kernel invocations. We will introduce two approaches and discuss the effects of the synchronization techniques used. The work distribution and the warp-based implementation will be presented at every step of the genetic algorithm. The introduced approach will be compared with the traditional island model and with similar and previous works.

The remainder of this paper is organized as follows: Section 2 introduces the previous works. Background information about the genetic algorithm for the TSP, GPU computing, and implementation details of the genetic algorithm over the GPU are highlighted in Section 3. Section 4 presents the lightweight island approaches, while Section 5 focuses on simulation and experimental results. Finally, we conclude the paper in Section 6.

2. PREVIOUS WORKS

The genetic algorithm is one of the most popular algorithms used to solve many complex problems, such as TSP. Many researchers address the acceleration of the genetic algorithm on GPUs to obtain results in a better time. Generally, researchers followed the three models of the parallel genetic algorithm: the master-slave model, the island model, and the cellular model.

Authors in [11, 12] introduced an approach based on the master-slave model. Each step of the genetic algorithm is associated with a separate kernel. This means a multi-kernel invocation and CPU-GPU data exchange at each iteration, which negatively affects performance.

An approach based on master-slave was also introduced in [13] to solve the traveling salesman problem. This approach depends on switching among kernels (SAK). It used three kernels to implement the steps in GA. The first kernel is responsible for fitting the population. The second kernel performs crossover, mutation, and fitness calculations. The third one executes the selection operator. The number of threads in the CUDA configuration was set equal to the number of individuals (grid-size*block-size=population-size). The time needed for kernel invocations is minimized because there is no data exchange between the host and the device. This method reduced the number of kernel invocations per iteration, but there are still a significant number of implicit synchronizations that can affect the execution time.

Many studies presented the island model of the GA over the GPU. A fully Distributed Island Model approach was introduced in [14]. In this approach, they ensured implicit global synchronization between the CPU and the GPU. This synchronization was performed by associating one kernel execution with one iteration of the evolutionary process. When the execution of one iteration is finished, the hand returns to the CPU. This synchronization mechanism decreases performance due to the large number of implicit synchronization points and the overhead of the kernels call. Each island was associated with a single block, and it contained 128 individuals (island-size). One individual was represented by one thread, which means that block-size=island-size= 128. The results are introduced for various numbers of islands (grid-size).

Authors in [15-18] also presented how to use GPUs to parallelize the island model of the genetic algorithm (IMGA). They focused on proposing parallel strategies for the genetic operators that are appropriate to the studied issue, but they did not address details about the mechanism for achieving global synchronization.

An approach named Scheduled Light Kernel (SLK) was presented in [10] for implementing GA on the GPU. This approach was inspired by the concept of persistent threads introduced in [19]. The introduced approach concentrated on persistently applying only the master-slave model to keep the threads alive inside a single kernel invocation. They collapse multiple kernel invocations into a single persistent kernel call. The execution method is determined by a work scheduling matrix. The GPU launching configuration (grid-size, block-size) of the persistent kernel is defined by finding the maximum number of blocks that is needed among the separate invocations. The researchers relied in their experiments on using the same number of available SMs (streaming multiprocessors) as a grid-size. As for the number of threads per block (block-size), they calculated the number of maximum threads as T = population-size (NPOP) / grid-size (B).

Since the aforementioned research did not focus, in its content, on developing a version of the persistent island model, we will provide the necessary details about that. We will make use of the research that applied the master-slave model using persistent threads to achieve the mechanism in the island model.

Considering that global synchronization greatly affects performance, we will discuss the mechanisms of synchronization and provide sufficient details about them. We will develop a warp-based parallelization strategy to achieve the best possible performance. We expect that the proposed approach will result in better performance.

3. BACKGROUND

3.1. THE GENETIC ALGORITHM FOR THE TSP PROBLEM

A genetic algorithm (GA) is an evolutionary problem-solving method based on Darwin's theory. This Algorithm evolves a group of solutions (population) by repeating some steps in sequence. GA starts by generating an initial population which is a group of solutions. After that, GA starts evolving these solutions in an iterative process. At each iteration, a set of steps are performed, including Crossover, Mutation, Evaluation, and Selection [20]. This process is repeated until it reaches a pre-specified condition. Fig. 1 shows the process of the GA.

The Traveling Salesman Problem (TSP) is one of the most studied combinatorial optimization problems. It is used in many real-world applications, such as UAV Path planning. The TSP should find the shortest route that visits a group of cities exactly once and returns to the initial city. The genetic algorithm is widely used to solve TSP and other NP-hard problems [6].

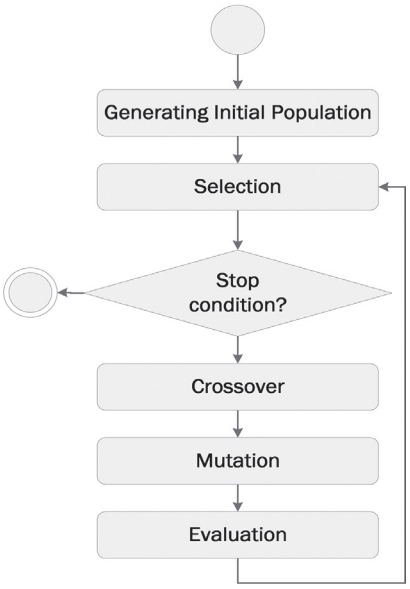


Fig. 1. Steps of the genetic algorithm

When using the genetic algorithm for solving TSP, each solution is represented by a route. A list of cities' indexes in a variation order makes up the route. The distances between cities are stored in an array. The fitness of each route is calculated by the distances array. There are various crossover techniques for the TSP, including PMX, CX, OX, etc.

3.2. GPU COMPUTING

In recent years, general-purpose GPUs (GPGPUs) have evolved into highly parallel, multithreaded, many-core processors with very high memory bandwidth [21]. Many laptops are now available with modern and powerful GPUs that contain thousands of cores, such as the NVIDIA GeForce RTX 30 Series [3].

Compute Unified Device Architecture (CUDA) is a general-purpose parallel computing platform and programming model introduced by NVIDIA. CUDA made developing parallel GPU applications much easier [21]. A typical CUDA program executes on both the GPU (Device) and the CPU (Host). The code executed on the GPU is grouped into a special function determined by the "global" qualifier and launched by the kernel invocation. The kernel parameters determine the grid and block dimensions. Threads within the block are partitioned into warps. A warp is made up of 32 parallel threads that are all executed based on the single instruction multiple thread (SIMT) paradigm.

The CUDA kernel should be invoked by the host code to carry out parallel computations [22]. The kernel invocation is asynchronous with respect to the host. After the kernel call, the host code must use the "cudaDeviceSynchronize()" API function to make the invocation synchronous. This indicates that the host will wait until all GPU threads have finished running. This API function ensures implicit global synchronization within the grid [1].

When the computational algorithm requires some synchronization points, the developer should break it up into several steps. Each step is mapped to one device kernel. The invocation of each kernel should be followed by the cudaDeviceSynchronize() function. This mechanism will perform synchronization between the algorithm steps. The CUDA API also contains the "__syncthreads()" function to perform inter-block local synchronization among threads in the same block.

In CUDA 9, NVIDIA introduced the cooperative groups extension. This extension allows the programmer to synchronize all threads in the same group [2, 23]. The group could be the grid. In this way, the programmer would be able to synchronize all threads in all blocks.

One of the requirements to use cooperative groups is to launch the cuda kernel through the "cudaLaunchCooperativeKernel" API function. The drawback of cooperative group synchronization is the limited number of launched blocks per multiprocessor. It cannot exceed the maximum number returned by the "cudaOccupancyMaxActiveBlocksPerMultiprocessor" function. This function can provide an occupancy prediction based on the block size and shared memory usage of a kernel [21].

GPUs, with a compute capability of 3.0 or higher, provided a mechanism to allow threads to directly read another thread's register in the same warp. The shuffle instruction enables threads in a warp to interact with one another without using shared or global memory. It offers applications a quick way to exchange data among threads in a warp [1].

3.3. GENETIC ALGORITHM OVER GPU

It is not easy to implement a GA on GPGPU, and numerous implementations and models have been suggested and explored in earlier literature. When you apply

the genetic algorithm in a parallel way, more than one thread will participate in completing the steps of this algorithm. Attention should be paid to performing each step completely before moving on to the next step. This means inserting synchronization points between the steps [24]. There are three models of the parallel genetic algorithm (PGA) on GPGPU: the master-slave model, the island model, and the cellular model [7–9].

In the traditional master-slave model, there is a single population. The evaluation step is achieved on the GPU by using a single kernel. The other GA steps are achieved sequentially on the CPU [8]. This model requires one kernel call for each iteration as well as two operations to copy data from host to device and vice versa. In the master-slave model, there is the possibility of applying all GA steps in parallel on the CPU by employing multiple CPU threads [25]. It can also be done on a GPU by mapping each step to a separate kernel.

This means that each iteration involves multi-kernel invocations. This model is good for performing implicit global synchronization by using the cudaDeviceSynchronize() function. However, it is inefficient since it needs frequent CPU-GPU communication to launch kernels at each iteration [7]. The execution time is negatively affected by this communication.

In the traditional island model, the population is divided into subpopulations called islands [8]. Each subpopulation is kept in the shared memory and evolves separately in one block. This model includes an additional step called migration, which is carried out between neighboring islands every predetermined number of iterations.

This model doesn't require global synchronization following each genetic step because each island evolves independently. Inter-block synchronization is implemented after each step [17]. The migration step is an exception and involves global synchronization between all GPU blocks to migrate some individuals from an island to a neighboring island through the global memory. Therefore, each iteration that doesn't perform migration can be achieved with only a single kernel call.

4. PROPOSED APPROACHES

This section describes the procedures required to implement the lightweight island model (LIM). Mixing between the island model and persistent threads is implemented in two approaches. The first is the lightweight island model based on cooperative groups (LIM-CG), and the second is the lightweight island model based on implicit synchronization (LIM-IS). The scheme in both approaches will be discussed, in addition to the work distribution mechanism at the steps of the genetic algorithm. Since both approaches are based on the island model, the work distribution mechanism will be identical in both.

4.1. APPROACH SCHEMES

In the first approach (LIM-CG), grid synchronization is applied using cooperative groups. All steps of the genetic algorithm, including migration, are mapped to only one kernel (GAKernel1), as shown in Fig. 2.

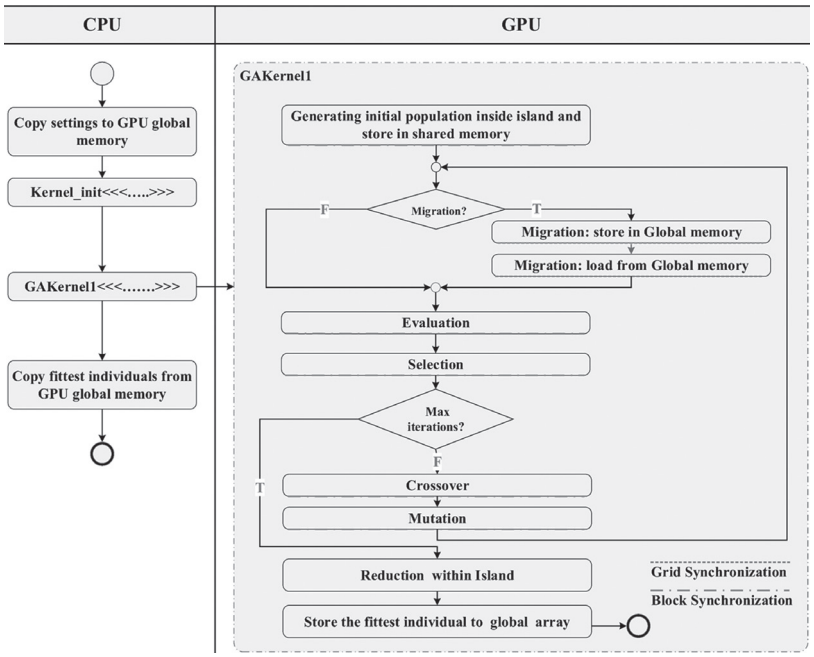


Fig. 2. Scheme of the lightweight island model based on cooperative groups (LIM-CG).

This kernel stays alive until it reaches the maximum number of iterations. This kernel code starts by generating the initial population. Each thread, in each block, is responsible for producing one individual through shuffle operations. The created individual is then stored in the island subpopulation array located in the shared memory. The "extern" identifier can be used to allocate this array. After that, all threads in the block enter the

evolving process (while loop). Migration is the first step that is encountered. It is performed frequently after a predetermined number of iterations (migration frequency). A global array is allocated in global memory to help migrate individuals between islands.

The second approach (LIM-IS) depends on performing an implicit global synchronization by returning control to the host after a predefined number of iterations. This number depends on the migration frequency. In this way, the number of global synchronization points is reduced. The migration kernel is invoked only when there is a migration. There are three kernels (InitialPopKernel, GAKernel2, and MigrationKernel3), as shown in Fig. 3.

The first one is responsible for generating the initial population. Each thread, in every block, is responsible for producing one individual, calculating the fitness, and then storing it in the global array. The second kernel repeats the steps of the genetic algorithms (while loop) until there is a need for migration (migration frequency). At that point, the control is returned to the host to launch the migration kernel just once. The loop counter at the host side is incremented by the value of migration frequency.

Synchronization inside every island is ensured by inter-block local synchronization. Global synchronization is implicitly guaranteed by returning the control to the host.

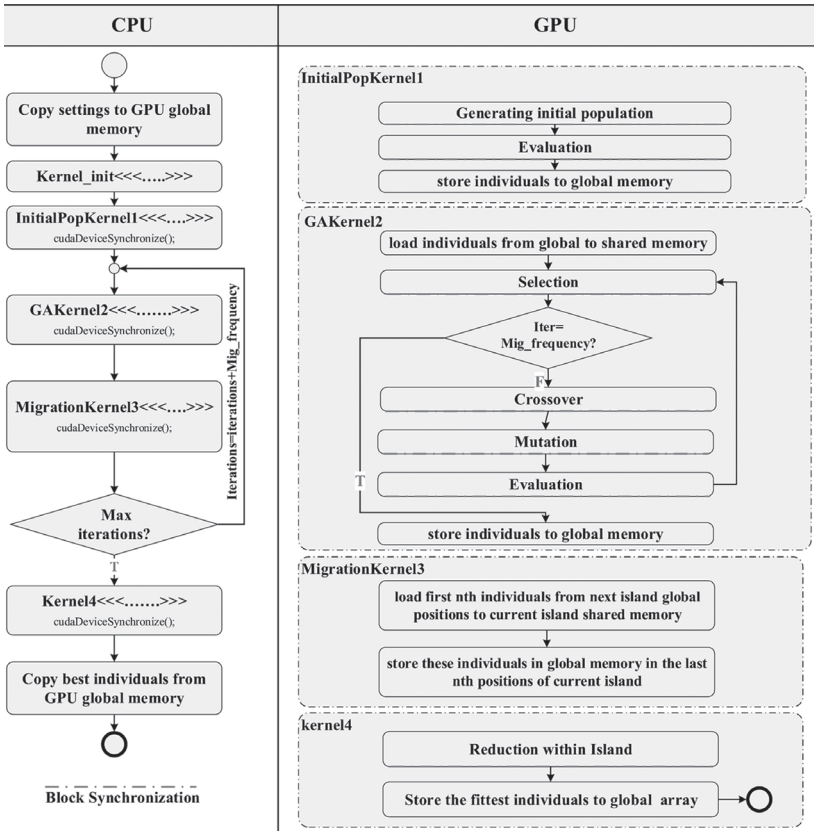


Fig. 3. Scheme of the lightweight island model based on implicit synchronization (LIM-IS).

Table 1. The number of synchronization points and the number of kernel invocations required in LIM-IS, LIM-CG, SAK, and SLK approaches.

Approach	Global synchronization points	Block synchronization points	Kernel invocations
SAK	#iterations * #steps		#iterations * #steps
SLK	#iterations * #steps		1
TIM	#iterations	#iterations * #steps	#iterations
LIM-CG	#iterations / #migrations	#iterations * #steps	1
LSM-IS	#iterations / #migrations	#iterations * #steps	#iterations / #migrations + #migrations

Migrating individuals between islands (blocks) is achieved through global memory. This approach does not rely on cooperative groups therefore, the number of blocks that can be executed on the device is unrestricted. It can be controlled either directly from the host or by a nested kernel. The host will be responsible for calling a single parent kernel, with one block and one thread, which in turn will invoke the other kernels.

Table 1 summarizes the synchronization points and the number of kernel invocations required. It compares the proposed approaches (LIM-IS, LIM-CG) with the switching among kernels approach (SAK) presented

in [13], the scheduled light kernel approach (SLK) presented in [10], and the traditional island model (TIM) presented in [14]. This is done without regard to the kernel that initializes the random vector.

In the SLK approach, they launched a single kernel with multiple blocks. All the blocks must be synchronized before moving toward the next scheduled computation (the next step). This style will increase the global synchronization points inside the kernel. In the SAK approach, there are three kernel invocations per iteration. This means three global synchronization points that are achieved by implicit synchronization.

In our approach, there is no need for global synchronization after each step. Inter-block synchronization, within the island, is performed after each step. It is guaranteed by the _syncthreads() function. Global synchronization is involved only when there is a migration between islands. Global synchronization is achieved through cooperative groups or implicit synchronization.

Several kernel invocations occur in the SAK model, and the number of threads participating in each step can differ. Although this paradigm reduces the number of kernel-running blocks, it increases the CPU-GPU communication overhead.

4.2. WORK DISTRIBUTION

When launching the persistent kernel in the SLK model, the maximum number of threads required for the overall steps is employed. A work scheduling matrix is used to distribute work among threads. It has several rows corresponding to the algorithm steps and several columns that define the working state of the block at this step. Some blocks can be assigned a no-operation (NOP) if they are not needed during a specific computation step.

In our proposed approaches, we'll try to effectively distribute the work among the threads to ensure that the most active threads are participating in the current step.

Work distribution is guaranteed by some variables. The value of these variables is determined by the probability of the genetic operators, the length of the route, and the configuration of the islands.

As illustrated in Pseudo-code 1, these variables are grouped into a structure called settings that is initially copied to the device. At each step, some equations are calculated using these variables to define the number of threads participating and how the work will be distributed among them.

Crossover and selection operations require the most computation time in comparison to the other steps. The one-point crossover is used in the crossover operator, while the tournament selection method is used in the selection step since it's preferable for parallel implementation.

In the selection step, the island population is divided into several groups depending on the number of selected individuals, needed and the number of warps inside the island.

Pseudo-code 1. The structure of the settings.

```
struct settings
{
    float CROSSOVER_RATE;
    float MUTATION_RATE;
    float SELECTION_RATE;
    float MIGRATION_RATE;
    int MAX_ITERATIONS;
    int ISLAND_SIZE;
    int ROUTE_SIZE;
    int NUM_ISLANDS;
    int MIGRATION_FREQ;
};
```

As seen in Fig. 4, each warp will manipulate one group to find the fittest individuals inside it through an unrolled reduction operation.

The reduction stops when it reaches the fittest individuals involved. Then, based on the warp and thread indices, the fittest individuals will be stored in the proper positions in the island population.

To facilitate the task, an alternative matrix is used, in which we store the fitness values of the individuals and their indices. Pseudo-code 2 describes the selection step.

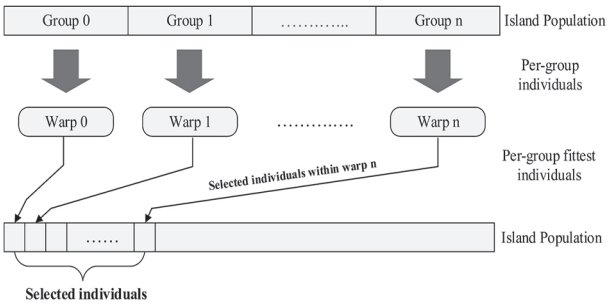


Fig. 4. Work distribution inside the island in the selection step.

The one-point crossover process takes place in two phases. In the first phase, the points located before the crossing point are transferred from the first parent to the child. In the second phase, all the points of the second parent are tested to see if they are not duplicated in the child. Then the non-duplicated points are added to the child after the crossing point.

The second stage requires many comparisons to be completed, depending on the length of the route. For this reason, we will allocate a warp to accomplish the crossover process between two parents to generate a child, as shown in Fig. 5.

At the start, Thread0 within the warp generates the index of the crossover point, the position for parent1, and the position for parent2.

Pseudo-code 2. The selection step.

```
#SelectedIndiveduals = SELECTION_RATE * ISLAND_SIZE #warpsInIslad= ISLAND_SIZE /32
WarpIndex= trunc(thid / 32)
#SelectionOperationsInsideWarp=ceil  (#SelectedIndiveduals / #warpsInIsland)
Index= WarpIndex * #SelectionOperationsInsideWarp
Unrolled Reduction Inside the group
Store fittest individuals to shared memory starting from index
```

Pseudo-code 3. The crossover step.

```
#CrossedIndiveduals= CROSSOVER_RATE * ISLAND_SIZE
#warpsInIslad= ISLAND_SIZE /32
WarpIndex= trunc(thid / 32)
#CrossoverOperationsInsideWarp=ceil (#CrossedIndiveduals / #warpsInIsland)
ChildIndex= #SelectedIndiveduals + WarpIndex * #CrossoverOperationsInsideWarp
for(j=0;j<#CrossoverOperationsInsideWarp;j++)
    CrossPosition=generatRandom (1,ROUT_SIZE-1)
    parent1Index= generatRandom (0,#selectedIndividuals)
    parent2Index= generatRandom (0,#selectedIndividuals)
    CrossPosition = __shfl_sync (0xFFFFFFFF,CrossPosition,0,32)
    parent1Index = __shfl_sync (0xFFFFFFFF,parent1Index, 0, 32)
    parent2Index = __shfl_sync (0xFFFFFFFF,parent2Index, 0, 32)
    read parent1, parent2
    stride=32
    threadIndex=threadIdx.x%32
    for(i= threadIndex; i< CrossPosition ; i=i+stride)
       if(i< CrossPosition)
              IslandPop[ChildIndex].rout[i]=parent1.rout[i]
    GroupSize = ceil(ROUT_SIZE/32)
    for(i= threadIndex * GroupSize;i<threadIndex * GroupSize+ GroupSize;i++)
       Test duplicate points in parent2[i] with parent1[1, CrossPosition]
    Store non-duplicated points to shared memory
    Thread0 will update the child located at the ChildIndex position
    childIndex= childIndex+1
```

Then it broadcasts these values to all of the threads in the same warp via the shuffle operation.

The threads within the warp will participate in transferring the points from the first parent to the child. This will decrease the divergence among threads in the same warp. After that, the points of the second parent will be divided and distributed among the threads to be tested for doubling in the child.

The value of the non-duplicated point, or -1, will be recorded to indicate the presence of doubling.

This process is done by using the data of the first father [0, crossing position] that is stored with each thread to avoid saving multiple copies of the child.

Finally, the non-duplicated points will be stored in shared memory. Thread0 inside the warp will update the points of the child in the population after the crossing position. Pseudo-code 3 describes the details of the crossover step.

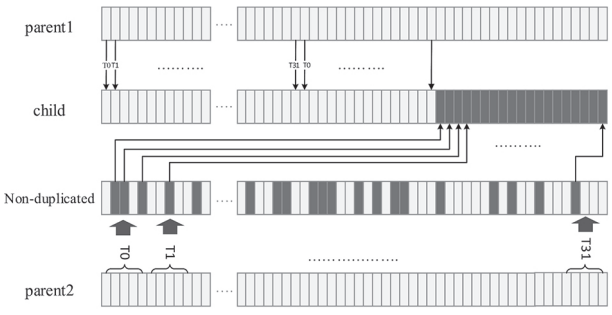


Fig. 5. Work distribution in the crossover process

5. EXPERIMENTAL RESULTS

This section presents the results of the two presented approaches (LIM-IS, LIM-CG) compared with the serial CPU, the switching among kernels approach (SAK) presented in [13], the scheduled light kernel approach (SLK) presented in [10], and the traditional island model (TIM) presented in [14].

The experiments were implemented on a laptop with an Intel Corei5-10 2.5GHz CPU and Nvidia RTX 3050Ti GPU. This GPU has 20 SMs, which means a total of 2560 CUDA cores. The compilation was performed using Microsoft Visual Studio 2019 with CUDA 11.6 SDK. The experiments are carried out with various numbers of population-size.

Since we applied warp-based parallelism, as explained in Section 4, the block-size may vary from the island-size. Where a single individual is mapped to several threads. In order to determine the GPU configurations (grid-size, block-size) that lead to the best possible performance, the first experiment will test the performance of our island-based approaches (LIM-IS, LIM-CG) against the serial CPU implementation. The number of individuals per island (island-size) is set to 128. Each block manipulates a single island. The grid-size determines the number of running blocks (islands). The population-size results from island-size*128. The same settings, listed in Table 2, were adopted during the implementation.

Table 2. Parameters of the TSP and the genetic algorithm

Parameter	Value
Problem	Att48
Maximum iterations	1000
Selection	Tournament selection (30%)
Crossover	One-point crossover (65%)
Mutation	Swap mutation (5%)
Migration topology	Unidirectional ring
Migration frequency	Every 50 iterations

The results of the first experiment are displayed in Table 3. Results show that choosing a block-size=1024 will provide the best performance compared to the serial implementation. Taking into account that the maximum number of threads that can run on the GPU being used is 1024 threads per block.

This experiment also proves that the GPU implementation gives a significant improvement in execution time compared to the serial CPU implementation. Acceleration of up to 27x has been achieved using the GPU, knowing that the population-size and algorithm parameters are the same in both cases.

In the second experiment, we will compare our approaches to those of SAK, SLK, and TIM for different population sizes. The GPU configuration (grid-size, block-size), as described in Section 2, was different for each approach. This relates to the population-size that was being used in the experiment.

To demonstrate the experimental process, Table 4 displays the necessary GPU settings to test each of the aforementioned approaches on the same population size. Each thread in the TIM, SLK, and SAK approaches, was mapped to a single individual. The number of GPU running threads must be equal to the population-size. In our approaches, and based on the results of the first experiment, the block-size is set to 1024.

The experiments were carried out with various numbers of population-size. The GA-TSP parameters, listed in Table 2, were adopted during the implementation of serial CPU and parallel GPU approaches.

Table 3. The execution time and speedup of the LIM-CG and LIM-IS approaches for different numbers of block-size and population-size

Block-size	Grid-size	Island-size	Population-size	Serial time (ms)	LIM-CG		LIM-IS	
					Time (ms)	Speedup	Time (ms)	Speedup
128	8	128	1024	1434.62	1304.20	1.1	2049.45	0.7
	16	128	2048	2857.78	2198.29	1.3	2597.98	1.1
	32	128	4096	5661.31	2358.88	2.4	2461.44	2.3
	64	128	8192	11278.44	3759.48	3	3638.21	3.1
	128	128	16384	22685.26	5532.99	4.1	4931.58	4.6
	256	128	32768	45543.33			8433.95	5.4
256	8	128	1024	1434.62	1024.73	1.4	1103.55	1.3
	16	128	2048	2857.78	893.06	3.2	952.59	3
	32	128	4096	5661.31	1286.66	4.4	1347.93	4.2
	64	128	8192	11278.44	1819.10	6.2	2128.01	5.3
	128	128	16384	22685.26	3287.72	6.9	3065.58	7.4
	256	128	32768	45543.33			5117.23	8.9
512	8	128	1024	1434.62	843.89	1.7	896.63	1.6
	16	128	2048	2857.78	476.30	6	529.22	5.4
	32	128	4096	5661.31	602.27	9.4	622.12	9.1
	64	128	8192	11278.44	989.34	11.4	947.77	11.9
	128	128	16384	22685.26	1731.70	13.1	1786.24	12.7
	256	128	32768	45543.33			2828.78	16.1
1024	8	128	1024	1434.62	531.34	2.7	683.15	2.1
	16	128	2048	2857.78	280.17	10.2	357.22	8
	32	128	4096	5661.31	339.00	16.7	365.25	15.5
	64	128	8192	11278.44	433.79	26	458.47	24.6
	128	128	16384	22685.26	807.30	28.1	840.19	27
	256	128	32768	45543.33			1615.01	28.2

Table 5 and Fig. 6 display the execution times for each approach. Fig. 7 compares the speedup of the suggested approaches with serial execution and earlier-mentioned studies.

The results shown in Fig. 6 and Fig. 7 demonstrate that the two new approaches can increase the speed-up to 4.5x over the TIM, and up to 1.5x– 2x over SLK and SAK approaches.

Table 4. The GPU configuration.

Approach	GPU configurations
LIM-CG LIM-IS	Island-size: 128 Grid-size: Population-size/128, Block-size: 1024
TIM	Island-size: 128 Grid-size: Population-size/128, Block-size: 128
SLK	Grid-size: 40 (twice the number of available SMs) Block-size: population-size / grid-size.
SAK	Grid-size: population-size / block-size. Block-size: 1024 (The maximum number allowed)

Table 5. The execution time (ms) of the proposed approaches LIM-IS and LIM-CG compared with SAK, SLK, and TIM approaches.

Population-size	Serial	TIM	SAK	SLK	LIM-CG	LIM-IS
1024	1434.6	1434.6	1024.7	478.2	531.3	683.1
2048	2857.7	1512.9	697.1	453.6	280.1	357.2
4096	5661.3	2358.8	775.5	496.6	339.0	365.2
8192	11278.4	2819.6	848.0	751.9	433.7	458.4
16384	22685.2	3979.8	1597.5	1281.6	807.3	840.1
32768	45543.3	7116.1	3098.1	2384.4		1615.0

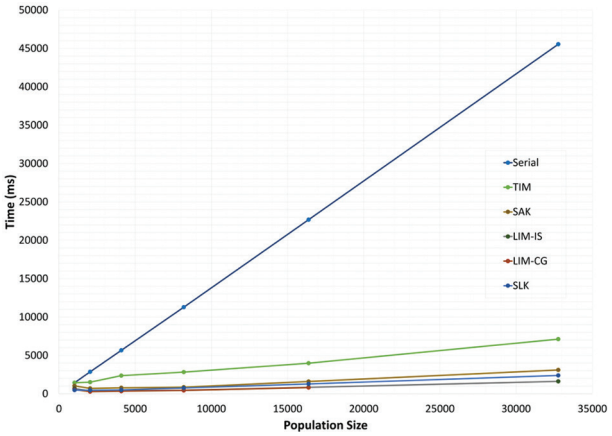


Fig. 6. The execution time of the proposed approaches LIM-IS and LIM-CG compared with SAK, SLK, and TIM approaches.

It can be seen that the first approach, which uses co-operative groups, is limited to a maximum number of blocks within the device.

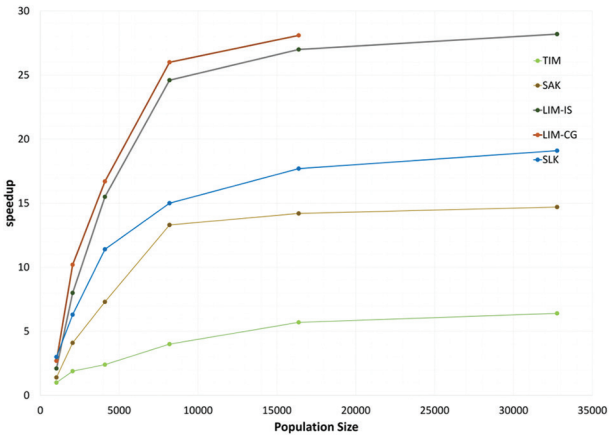


Fig. 7. The speedup achieved by the proposed approaches LIM-IS, and LIM-CG compared with SAK, SLK, and the TIM approaches

The last experiment will focus on evaluating the quality of the obtained solutions. The parameters and settings applied in this experiment are those listed in Table 2 and Table 4. The relative error between the fittest solution's cost and the optimal solution's cost, represented by equation (1), was calculated. The execution was repeated ten times, and the average value was recorded.

$$Relative\ error = \left| \frac{Cost_{Fittest} - Cost_{Optimal}}{Cost_{Optimal}} \right| \quad (1)$$

Fig. 8 displays the relative error resulting from serial CPU and GPU-based parallel implementations. It has been observed that the quality of the solution improves when the size of the population increases. The island-based approaches (TIM, LIM-CG, LIM-IS) have an advantage in obtaining the highest quality because of the migration step. Migrating some individuals between islands gives them the possibility to explore different regions of the search space and discover better-quality solutions.

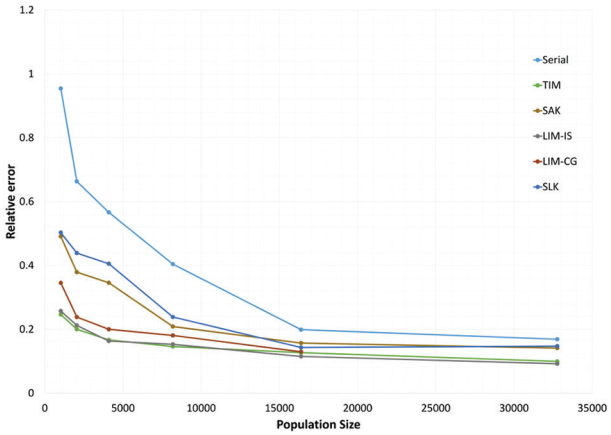


Fig. 8. The relative error of the solutions that resulted from serial, LIM-IS, LIM-CG, SAK, SLK, and TIM approaches.

Noting that the experiments were carried out for 1000 iterations for all aforementioned approaches.

6. CONCLUSION AND FUTURE WORK

In this paper, we highlight the details of implementing a lightweight island model (LIM) on a general-purpose graphic processor unit (GPGPU) for the TSP problem. Two approaches were suggested (LIM-IS, LIM-CG) that follow the concept of persistent threads. The necessary details to convert the traditional island model (TIM) into a lightweight island version were discussed. We also reviewed the previously suggested researches that worked on transforming the traditional master-slave model into new approaches, such as the switching among kernels approach (SAK) and the scheduled light kernel approach (SLK). A new mechanism was presented for distributing work between live threads inside the islands. The GPU configurations were tested and detailed to get the best possible performance. The introduced approaches were compared, in terms of execution time and solution quality, with serial implementation and with previous works including the traditional island model (TIM), switching among kernels approach (SAK), and scheduled light kernel approach (SLK).

Our suggested approaches produced much better results compared with these previous works. The speedup achieved is up to 27x compared with the serial CPU and up to 4.5x compared to the TIM approach. The speedup improvement was up to 1.5– 2x over SLK and SAK approaches.

In terms of solution quality, the proposed approaches produced better solutions than SAK and SLK, whereas the results were nearly identical between the proposed approaches and TIM.

For future work, these approaches can be tested on larger TSP problems to measure the effectiveness of the synchronization methods and the parallelization strategy.

7. REFERENCES

[1] J. Cheng, M. Grossman, T. McKercher, "Professional CUDA C Programming", Wiley, 2014.

[2] G. Barlas, "Multicore and GPU Programming: An Integrated Approach", 2nd Edition, Elsevier, 2022.

[3] NVIDIA, "GeForce RTX 30-Series Laptops | NVIDIA", https://www.nvidia.com/en-me/geforce/gaming-laptops/. (accessed: 2022)

[4] B. Basbous, "2D UAV Path Planning with Radar Threatening Areas using Simulated Annealing Algorithm for Event Detection", Proceedings of the International Conference on Artificial Intelligence and Data Processing, Malatya, Turkey, 28-30 September 2018, pp. 1-7.

[5] M. Cakir, "2D path planning of UAVs with genetic algorithm in a constrained environment", Proceedings of the 6th International Conference on Modeling, Simulation, and Applied Optimization, Istanbul, Turkey, 27-29 May 2015, pp. 1-5.

[6] P. Larrañaga, C. M. H. Kuijpers, R. H. Murga, I. Inza, S. Dizdarevic, "Genetic Algorithms for the Travelling Salesman Problem: A Review of Representations and Operators", Artificial Intelligence Review, Vol. 13, No. 2, 1999, pp. 129-170.

[7] J. R. Cheng, M. Gen, "Accelerating genetic algorithms with GPU computing: A selective overview", Computers & Industrial Engineering, Vol. 128, 2019, pp. 514-525.

[8] J. R. Cheng, M. Gen, "Parallel Genetic Algorithms with GPU Computing", Industry 4.0 - Impact on Intelligent Logistics and Manufacturing, Vol. 32, 2020, pp. 137-144.

[9] T. Harada, E. Alba, "Parallel Genetic Algorithms: A Useful Survey", ACM Computing Surveys, Vol. 53, No. 4, 2021, pp. 1-39.

[10] N. Capodieci, P. Burgio, "Efficient Implementation of Genetic Algorithms on GP-GPU with Scheduled Persistent CUDA Threads", Proceedings of the Seventh International Symposium on Parallel Architectures, Algorithms and Programming, Nanjing, China, 12-14 December 2015, pp. 6-12.

[11] R. Saxena, M. Jain, S. Bhadri, S. Khemka, "Parallelizing GA based heuristic approach for TSP over CUDA and OPENMP", Proceedings of the International Conference on Advances in Computing, Communications and Informatics, Udupi, India, 13-16 September 2017, pp. 1934-1940.

[12] M. Oiso, Y. Matsumura, T. Yasuda, K. Ohkura, "Implementation Method of Genetic Algorithms to the CUDA Environment using Data Parallelization", Tehnički Vjesnik, Vol. 18, No. 4, 2011, pp. 511-517.

[13] M. Abbasi et al. "An efficient parallel genetic algorithm solution for vehicle routing problem in cloud implementation of the intelligent transportation systems", Journal of Cloud Computing, Vol. 9, No. 1, 2020.

[14] T. Van Luong, N. Melab, E.-G. Talbi, "GPU-based island model for evolutionary algorithms", Proceedings of the 12th annual conference on Genetic and evolutionary computation, 2010, pp. 1089-1096.

[15] X. Sun, L.-F. Lai, P. Chou, L.-R. Chen, and C.-C. Wu, "On GPU Implementation of the Island Model Genetic Algorithm for Solving the Unequal Area Facility Layout Problem", Applied Sciences, Vol. 8, No. 9, 2018, p. 1604.

[16] B. Wang et al. "Multipopulation Genetic Algorithm Based on GPU for Solving TSP Problem", Mathematical Problems in Engineering, Vol. 2020, 2020.

[17] D. M. Janssen, A. W. C. Liew, "Acceleration of genetic algorithm on GPU CUDA platform", Proceedings of the 20th International Conference on Parallel and Distributed Computing, Applications and Technologies, Gold Coast, QLD, Australia, 5-7 December 2019, pp. 208-213.

[18] P. Pospichal, J. Schwarz, J. Jaros, "Parallel genetic algorithm solving 0/1 knapsack problem running on the GPU", Mendel, 2010, pp. 64-70.

[19] K. Gupta, J. A. Stuart, J. D. Owens, "A study of Persistent Threads style GPU programming for GPGPU workloads", Proceedings of Innovative Parallell Computing, San Jose, CA, USA, 13-14 May 2012, pp. 1-14.

[20] D. E. Goldberg, J. H. Holland, "Genetic Algorithms in Search, Optimization, and Machine Learning", Machine Learning, Vol. 3, No. 2, 1979, pp. 95-99.

[21] NVIDIA, "Cuda C Programming Guide", 2017, pp. 1-261.

[22] T. Soyata, "GPU Parallel Program Development Using CUDA", CRC Press, Chapman and Hall/CRC, 2018.

[23] NVIDIA, "Programming Guide :: CUDA Toolkit Documentation", https://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html#cooperative-groups (accessed: 2022)

[24] S. Chen, S. Davis, H. Jiang, A. Novobilski, "CUDA-Based Genetic Algorithm on Traveling Salesman Problem", Studies in Computational Intelligence, Vol. 364, 2011, pp. 241-252.

[25] M. AlRaslan, A. H. AlKurdi, "UAV Path Planning using Genetic Algorithm with Parallel Implementation", International Journal of Computer Science and Information Technologies, Vol. 10, No. 02, 2022, pp. 1-15.

Multi Indicator based Hierarchical Strategies for Technical Analysis of Crypto market Paradigm

Original Scientific Paper

[1] **V S S K R Naganjaneyulu G**
Department of ECE,
National Institute of Technology Karnataka,
Surathkal, Mangalore 575025 India
snd.786@gmail.com

[2] **Prashanth G**
Department of ECE, RGUKT Basar,
Nirmal, Telangana 504107 India
g.prashu18@gmail.com

[3] **Revanth M**
Department of ECE, RGUKT Basar,
Nirmal, Telangana 504107 India
revanthmuniganti328@gmail.com

[4] **Dr. A V Narasimhadhan**
Department of ECE,
National Institute of Technology Karnataka,
Surathkal, Mangalore 575025 India
dhan257@gmail.com

Abstract *– The usage of technical analysis in the crypto market is very popular among algorithmic traders. This involves the application of strategies based on technical indicators, which shoot BUY and SELL signals to help the investors to take trading decisions. However, instead of depending on the popular myths of the market, a proper empirical analysis can be helpful in lucrative endeavors in trading cryptocurrencies. In this work, four technical indicators namely Exponential Moving Averages (EMA), Bollinger Bands (BB), Relative Strength Index (RSI), and Parabolic Stop And Reverse (PSAR) are used individually to devise strategies that are implemented, and their performance is validated using the price data of Bitcoin from yahoo finance for 2018-22, individually for each year and all the five years consolidated to compute the performance metrics including Profit percentage, Net profitability percentage, and Number of total transactions. The results show that the performance of strategies based on trend indicators is better than that of momentum indicators where the EMA strategy provided the best result with a profit percentage of 394.13%. Further, the performance of these strategies is analyzed in three different market scenarios namely Uptrend/Bullish trend, Downtrend/Bearish trend, and Fluctuating/oscillating markets to analyze the applicability of each of these smart strategies in the three scenarios. Based on the insights obtained from the analysis, Hybrid strategies using multiple indicators with a hierarchical approach are developed whose performance is further improved by imposing constraints in a Downtrend market scenario. The novelty of these algorithms is that they identify the scenario in the market using multiple indicators in a hierarchal approach, and utilize appropriate indicators as per the market scenario. Four strategies namely, Multi indicator based Hierarchical Strategy (MIHS) with EMA9, Multi indicator based Hierarchical Strategy (MIHS) with EMA7, Multi-Indicator based Hierarchical Constrained Strategy (MIHCS) with EMA9, and Multi-Indicator based Hierarchical Constrained Strategy (MIHCS) with EMA7 are developed which give profit percentage of 154.45%, 437.48%, 256.31%, and 701.77% respectively when applied on the Bitcoin price data during 2018-22.*

Keywords*: Cryptocurrency, Technical Indicators, Exponential Moving Average, Relative Strength Index, Bollinger Bands, Parabolic Stop and Reverse*

1. INTRODUCTION

Crypto-currency, or crypto is a digital currency designed to work as a medium of exchange through a computer network that is not reliant on any central authority, such as a government or bank, to uphold or maintain. It is a decentralized system for verifying that the parties to a transaction have the money they claim to have, eliminating the need for traditional intermediaries, such as banks, when funds are being transferred between two entities. Cryptocurrencies are a subset of virtual currencies that use cryptography for security. Bitcoin [1] is the first and most popular standard coin followed by Ethereum and Binance. There are thousands of such cryptocurrencies that are introduced over the past decade. The high-profit margins in the crypto market attracted several investors' attention and cryptocurrency became an essential part of their portfolio. However, in the past two years, the crypto market suffered huge losses because of legal issues.

Trading is the practice of exchanging various commodities like metals, spices, stocks, and cryptocurrencies. In the last decade, many people started investing in various market paradigms like cryptocurrencies, stocks, commodities, bonds, and forex exchange. Because of a

lack of knowledge and expertise in these markets, several end up with losses. Experts built technical indicators and strategies based on indicators for technical analysis in these markets to provide suggestions for traders to obtain profits. Technical analysis involves the usage of historical data on the financial commodity to predict the movement of the price using technical indicators. Strategies formed using the technical indicators help the investors in taking trading decisions, by shooting *BUY*, *SELL*, and HOLD signals in appropriate conditions. Automation of trading using these strategies gives rise to algorithmic trading which saves a lot of time and energy for traders who can avoid monitoring the prices of financial commodities continuously. Technical analysis when used wisely depending on the market scenario can deliver high profits in the Crypto market. There are a lot of resources in the form of websites, mobile applications, and YouTube videos available online, suggesting that traders use these strategies to become millionaires in no time. However, seldom do they show a proper validation of those strategies using the real data of stocks and crypto or other commodities. The trading can be made more profitable by a proper analysis of these strategies on real data. In this work, an empirical analysis of four such strategies is performed using the Bitcoin price data from Yahoo Finance.

2. RELATED WORK

The majority of the works on the Cryptocurrency market focused on price prediction of cryptocurrency based on Artificial intelligence and Machine learning methods rather than technical analysis using technical indicators. In [13], it is claimed that to obtain abnormal profits, technical analysis is more relevant than the machine learning approach of price prediction. The price prediction approach doesn't provide suggestions to *BUY* or *SELL* the financial commodity and leaves this task to the investor whereas the strategies that are devised for technical analysis provide these trade signals to help investors to make trade decisions.

EMA being the most popular indicator among investors received good attention from the researchers to develop strategies for algorithmic trading. Simple moving average computes the average with uniform weights to all the data points whereas Exponential moving average prioritizes the recent data points by multiplying them with higher weights when compared to the remaining data points. The profitability of the moving averages is examined in [3] wherein the performance analysis of Simple Moving Average (SMA), Weighted Moving Average (WMA), and Exponential Moving Average (EMA) in the Forex market is studied using EUR/USD, AUD/USD and GBP/USD exchange data to conclude that EMA is the best method followed by WMA and SMA. De Souza et al. [4], validated the profitability of the EMA strategy by developing an automated trading system using technical analysis based on the EMA strategy in stocks of BRICS countries' stock markets and has shown that the returns are higher than the investment in Russia and India. In [5], the EMA strategy is used to determine the trading points for 50 index stocks of Thailand and has shown that there are returns of 9% in a year. In [6], F Papailias et al. proposed a modified EMA strategy which is a combination of cross-over 'buy' signals and a dynamic threshold value that acts as a dynamic trailing stop. The technical analysis performed using DJIA, SP 500, EUR/USD exchange, and ETFs showed that the modified method provides higher cumulative returns when compared to the standard algorithm. Tanaka Y et al. in [7] proposed the combinational use of technical indicators for technical analysis in the stock market. In [8], the authors have proved that the profitability of the specific trading rules using seven trend indicators in the Bitcoin market is higher when compared to Buy and Hold strategy. J.C. Phooi et al [9], proposed a dynamically adjustable Moving average indicator and established its superiority in delivering profits by evaluatory studies on Asian Tiger's future markets. Chu et al [10] proposed a signal-based momentum strategy that has two variations namely, a time series method and a cross-sectional method to employ in the 7 largest cryptocurrencies. It has been observed that signal-based strategy performs exceptionally well when compared with return-based strategies. However, there is no particular single parameter to gain good returns for signal-based strategies.

The relative strength index (RSI) is popular among traders because of its relevance to *fluctuating* markets. RSI indicates overbought and oversold situations in the market employing two thresholds. The key to the profitability is proper selection of these thresholds. In [11], the authors performed empirical studies on Bitcoin price data with simple average and RSI to conclude the supremacy of simple average over RSI in terms of profitability. The authors in [12] examined the RSI strategy with 30-70 thresholds on 10 various cryptocurrencies like Bitcoin, Ripple, Etherium, and Bitcoin cash to prove that the popular strategy doesn't perform well and further proposed a modification of Cardwell's strategy with sub-optimal usage of RSI results in above average profitability. A trading model is developed in [13] using modified RSI which has several parameters including the trend detection period, RSI buy-sell trigger levels, and periods and these are optimized using genetic algorithms. Further, the trading performance of the model is compared against Buy-and-Hold and standard RSI indicator usage where the profits from the Trend-Normalized RSI indicator are not very volatile and achievable in the stock market. Anderson et al., [14], examined the importance of the selection of control parameters for RSI and busted the myth of popular strategy. An empirical comparison of strategies based on Bollinger bands, and exponential moving averages is presented in [15]. In [16], the use of technical analysis based on parabolic stop and reverse strategy in the Forex market is demonstrated.

In the literature, there is little focus on identifying the trend in the market using multiple indicators and using appropriate strategies for each scenario separately to provide loss protection and higher profits. The current work is an attempt to bridge such a gap.

3. TECHNICAL INDICATORS

A technical indicator is a tool used for the prediction of trends in the movement of the price of commodities. They are designed using the volume of buying, selling, and trading share of the commodity in the market, opening, closing, and range of price in the given time frame. Technical Indicators are categorized into four groups namely Trend indicators, Volatility Indicators Momentum Indicators, and Volume Indicators. Trend indicators tell which direction the market is moving in if there is a trend at all. They're sometimes called oscillators because they tend to move between high and low values like a wave. Momentum indicators are technical analysis tools used to determine the strength or weakness of a stock's price. Momentum measures the rate of the rise or fall of stock prices. The volatility indicators are technical tools that measure how far security stretches away from its mean price, higher and lower and they compute the dispersion of returns over time in a visual format that technicians use to estimate whether this mathematical input is increasing or decreasing. Volume means the number of shares traded at one time. Volume indicators are mathematical formulas that are visually represented in the most commonly used charting platforms. A list of popularly used indicators along with the definitions is given in Table [1].

Table 1. Definitions of popular technical indicators

S. No	Name of the Indicator	Definition	Type of indicator
1	Simple Moving Averages	SMA = $(\sum_{i=1}^{N} Price_i)/N$, N= Number of data samples	Trend indicator
2	Exponential Moving Averages [17]	EMA = Price(today) * k+EMA(yesterday)*(1-k), k is a smoothing factor	Trend indicator
3	Relative Strength Index (RSI) [18]	RSI = 100 - (100/(1+RS)) , RS = ($Average\ Gain$)/($Average\ Loss$) $Average\ Loss$ = ($sum\ of\ Loss\ column$)/($RSI\ Period$) , $Average\ Gain$ = ($sum\ of\ Gain\ column$)/($RSI\ Period$)	Momentum indicator
4	Parabolic Stop And Reverse [18]	$Uptrend$: $PSAR = PriorPSAR+PriorAF(PriorEP \cap PriorPSAR)$, $Downtrend$: $PSAR= PriorPSAR \cap PriorAF(PriorPSAR \cap PriorEP)$, $Extreme\ Point(EP)$: Highest high for an $Uptrend$ and lowest low for $Downtrend$, update each time a new EP is reached, $Acceleration\ Factor(AF)$: Default of 0.02, increasing by 0.02, each time a new EP is reached, with a maximum of 0.20	Trend indicator
5	Bollinger Bands [19]	$Middle\ Band = (Price1+Price2+Price3+ ...+PriceN)/N$, $Upper\ Band = Middle\ Band + k * \sigma$, $Lower\ Band = Middle\ Band - k * \sigma$	Volatility indicator

4. SMART STRATEGIES USING INDICATORS

Strategy is a newly formed logic using the combination of existing technical indicators to get maximum benefits. A common framework of strategies has been implemented in Python using 4 Technical Indicators namely Exponential Moving Average (EMA), Relative Strength Index (RSI), Parabolic Stop And Reverse (PSAR), and Bollinger Bands (BB).

4. 1. IMPLEMENTATION OF SMART STRATEGY

The following are the prerequisites for strategy implementation

A) Technical Analysis Library (TA-Lib) is a built-in Python library used for feature engineering which contains all the technical indicators and candlestick pattern recognition tools.

B) Yahoo Finance (yfinance Library) is an open-source tool that uses publicly available APIs and is intended for research and educational purposes. The data set used for the implementation of strategies on various coins is web scraped/downloaded from Yahoo Finance. The columns in the data set are date, time, open, low, high, close, adjacent close, and volume.

C) NumPy is an open-source Python package that provides the flexibility of using arrays for mathematical computations.

D) Pandas is an open-source Python package used for data analysis.

E) Plotly is an open-source plotting library that supports over 40 unique chart types covering a wide range of statistical, financial, geographic, scientific, and 3-dimensional use cases.

Implementation of the strategy contains the following steps:

- Installation of Packages and importing required libraries
- Reading data
- Strategy implementation
- Performance analysis

4.2. DEFINITIONS OF STRATEGIES WITH INDIVIDUAL TECHNICAL INDICATORS

- **Strategy 1**

Strategy 1 is based on EMA which is one of the most common technical indicators by investors which is defined below.

$$\text{EMA} = \begin{cases} BUY\,, & if\ SL\ (EMA9)\ >\ RL\ (EMA20) \\ SELL\,, & if\ SL\ (EMA9)\ <\ RL\ (EMA20) \end{cases}$$

Where, SL is the Signal line, and RL is the Reference line.

- **Strategy 2**

Strategy 2 is devised using Bollinger bands which have three lines namely, Lower band, Middle band, and Upper band that are calculated as per the definition given in Table [1] and are usually shown in the graph along the trace of candle sticks. Bollinger bands are volatility indicators and give overbought and oversold signals if the current price goes above the Upper band and the current price goes below the Lower band respectively. This indicator has two controlling parameters namely moving average period and standard deviation. In this work, a standard set of parameter values, 21 for moving average period and 2 for standard deviation are considered.

Strategy 2 is defined as follows,

$$\text{BB} = \begin{cases} BUY, & if\ CP\ \leq\ LB \\ SELL, & if\ CP\ \geq\ UB \\ HOLD, & otherwise \end{cases}$$

Where CP is the Closing price, and MB is the Middle band.

- **Strategy 3**

Strategy 3 is composed using the RSI indicator whose value will oscillate between 0 and 100 and plotted on the separate chart which is an indirect trace of Candle Sticks. The strategy uses two threshold levels, a standard value of a lower threshold 30 known to be an oversold level, and a standard value of an upper threshold, 70 known to be an overbought level. These threshold values may be varied as per the trade-off between risk and profitability.

Strategy 3 is defined as follows,

$$\text{RSI} = \begin{cases} BUY, & if\ RSI\ line\ \leq\ 40 \\ SELL, & if\ RSI\ line\ >\ 60 \\ HOLD, & otherwise \end{cases}$$

RSI is a momentum indicator, which is usually shown in a separate graph beneath the price date as an oscillating line between 0 and 100. In general, investors tend to buy the financial commodity at a lower threshold and sell if RSI is greater than a higher threshold. The successful use of RSI in the market heavily depends on the threshold values. In the current work, a popular set of thresholds, 40 and 60 are selected.

- **Strategy 4**

Strategy 4 is coined based on the Parabolic Stop And Reverse (PSAR) indicator that gives a signal line plotted as a dotted sequence along the trace of Candle Sticks. Strategy 4, shoots *BUY* if PSAR is less than the candlestick closing price and in an upward direction (Higher High (HH) is greater than Lower High (LH)) and it shoots *SELL* if PSAR is greater than the candlestick closing price and in the downward direction (Lower Low (LL) is smaller than Higher Low (HL)). The strategy is defined as follows,

$$\text{PSAR} = \begin{cases} BUY, & if\ (PSAR\ <\ CP)\ and\ (HH\ >\ LH) \\ SELL, & if\ (PSAR\ >\ CP)\ and\ (LL\ <\ HL) \\ HOLD, & otherwise \end{cases}$$

Where LL is Lower low, HL is Higher low, HH is Higher high, and LH is Lower high.

4.3. ANALYSIS OF STRATEGIES USING INDIVIDUAL INDICATORS:

Fig. [1], indicates that the market is in *Downtrend* most of the time despite a few fluctuations in between 2018, whereas in, 2019 for which the Bitcoin price data is shown in Fig. [2], till April the market has shown inertia prices, and from April to August it is bullish after which it is shown bearish trend till the end of the year. Fig. [3] shows that in 2020, the market is in *Uptrend* and reached the peak at the end of the year with few fluctuations here and there. The price data of Bitcoin data is shown in Fig. [4], from which the rapid moments in markets show signs of a *Fluctuating* market can be observed. In 2022, despite few signs of *Fluctuating* market in the first quarter, the market is Bearish which is depicted in Fig. [5]. Hence performing technical analysis on price data in this time frame is good enough as it is comprising the majority of test cases.

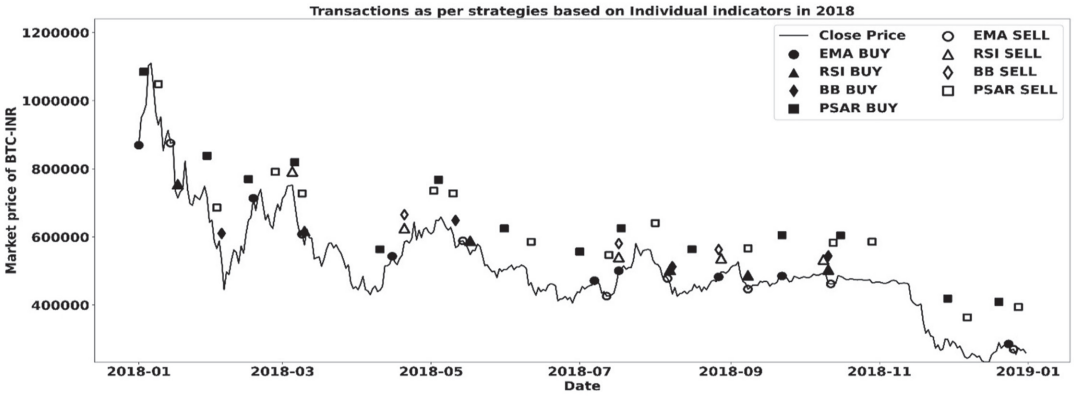


Fig. 1. Analysis of strategies based on individual indicators in 2018

Fig. 2. Analysis of strategies based on individual indicators in 2019

Fig. 3. Analysis of strategies based on individual indicators in 2020

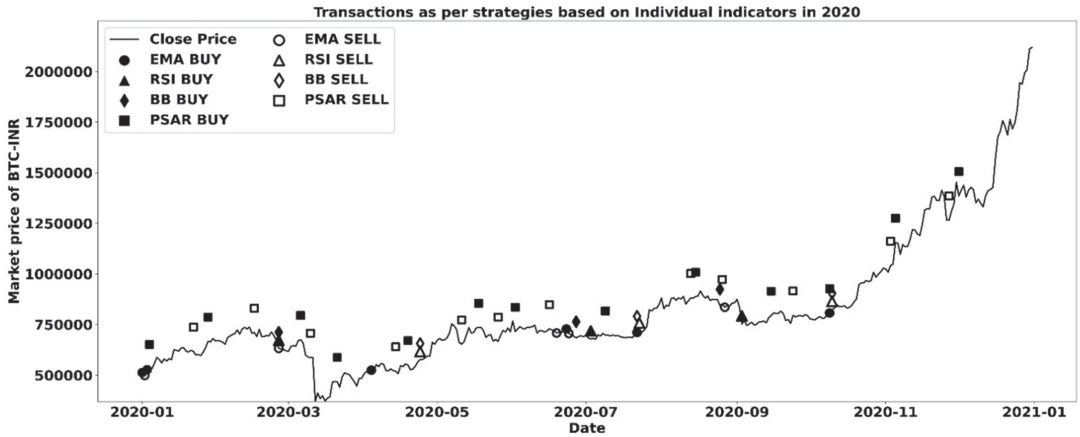


Fig. 4. Analysis of strategies based on individual indicators in 2021

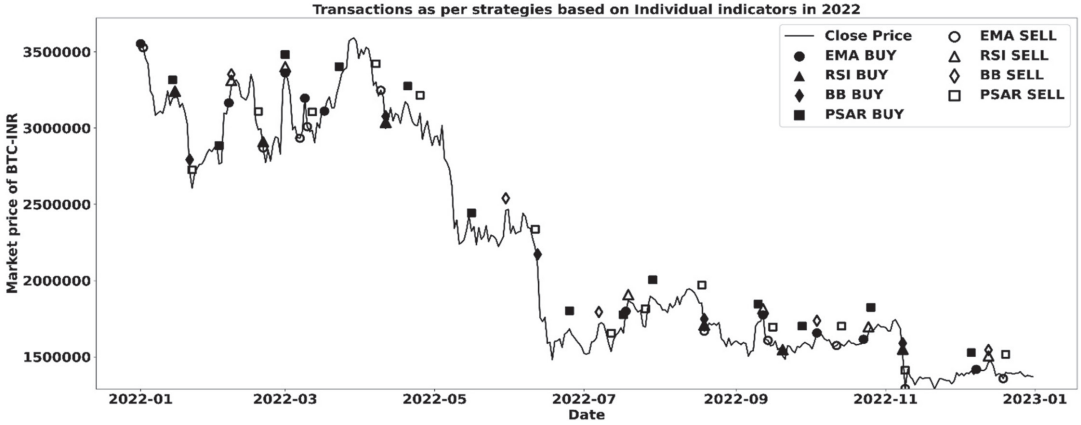


Fig. 5. Analysis of strategies based on individual indicators in 2022

4.3.1. Analysis of EMA strategy

Fig. [6], illustrates the application of EMA with 9 and 20 periods on *Bitcoin* data of 2018-22 consolidated. In the calculation of EMA, the most recent data is given higher weights to minimize the lagging limitation of moving average indicators which is unavoidable in moving average indicators.

From Fig. [6], it can be observed that the EMA strategy is triggering a *BUY/SELL* signal if there is an abrupt change in the price of the coin because of which the bullish and bearish scenarios of the market are captured to deliver good profits with minimal losses.

The transactions that happened between Feb 2109 and mid-July 2019, which can be observed in Fig. [2], are a sign that there is a possibility of obtaining higher profits than that actual, if the strategy generated *SELL* signal earlier. The lagging effect in EMA is the prime reason for these marginal gains. However, as EMA can pick up from the abrupt changes, it can generate the *SELL* signal in case of a bearish scenario which is evident from Fig. [4]. It can also be observed from Fig. [1,5], in the first quarter, where the market is *Fluctuating*, EMA is unable to deliver and capture the fast-moving market and ended up in losses. Hence EMA is suitable in case if market exhibits clear bearish and bullish trends but cannot deliver in case of *Fluctuating* market.

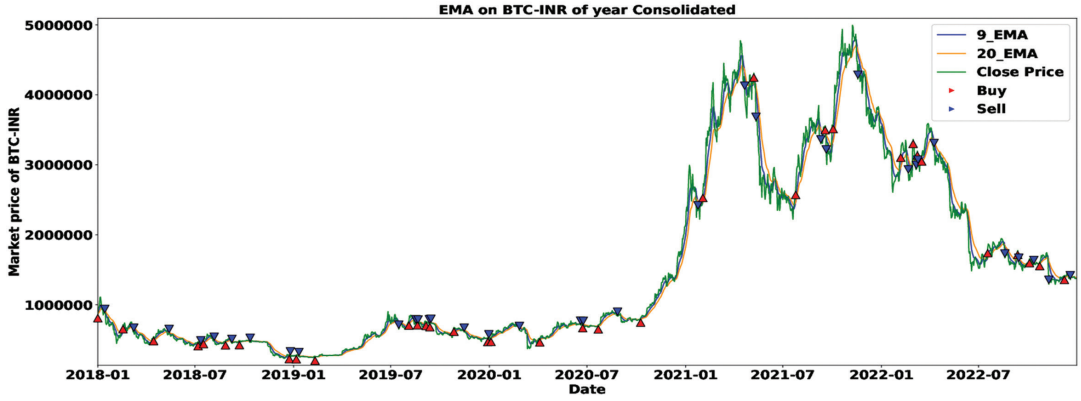


Fig. 6. Transactions as per EMA strategy over five years consolidated on *Bitcoin* price data

4.3.2. Analysis of BB strategy

Fig. [7], shows the application of BB on the bitcoin data of 2018-22 consolidated. Fig. [5], demonstrates that BB performs fairly well in the case of *Fluctuating* market but is not the best performer as it is not capturing all the ups and downs. From Fig. [5], in the *Downtrend*, BB is giving *BUY*, and *SELL* signals because of the *Fluctuating* market, resulting in losses. Fig. [3,4], convey that BB is unable to generate a *BUY* signal during the *Uptrend*. These pitfalls might be due to the sensitive threshold values, change of whose values may result in improvement, however at the cost of a trade-off.

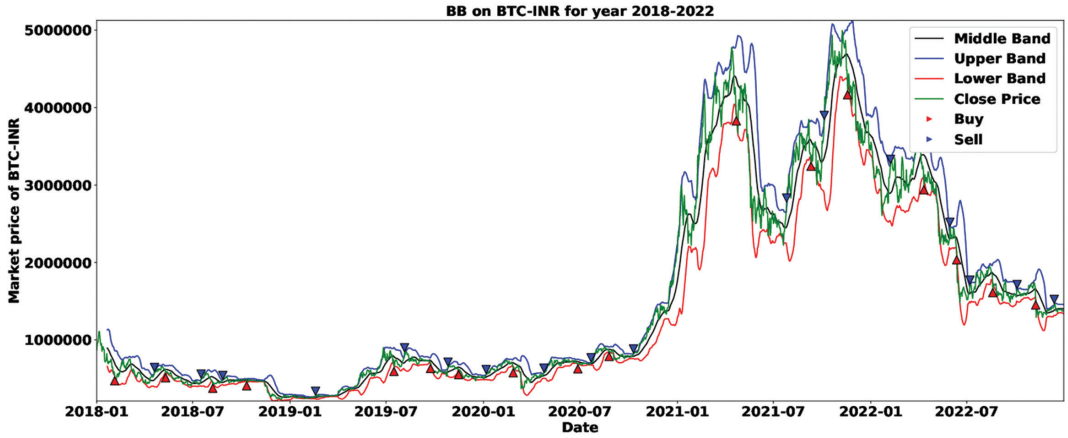


Fig. 7. Transactions as per BB strategy over five years consolidated on Bitcoin price data

4.3.3. Analysis of RSI strategy

The Fig. [8], depicts the application of RSI with 40 and 60 periods on Bitcoin data of 2018-22. In scenarios where there is either an *Uptrend/Downtrend*, the RSI indicator is unable to capture the trends and deliver profits. Fig. [5], ascertains that, in the first quarter of 2022, the market has shown rapid movement which is captured by RSI. This confirms the ability of RSI to deliver profits in such *Fluctuating* scenarios. It can also

be observed that the selection of the thresholds is very important for delivering profits. If the margin between the thresholds is too high, the immediate *SELL* signal after the *BUY* signal may be missed, and in such scenarios, it may result in huge losses. If the margin, between the thresholds, is too low, it may result in too many transactions with very little/no profits. Making these thresholds adaptive may be considered for better results.

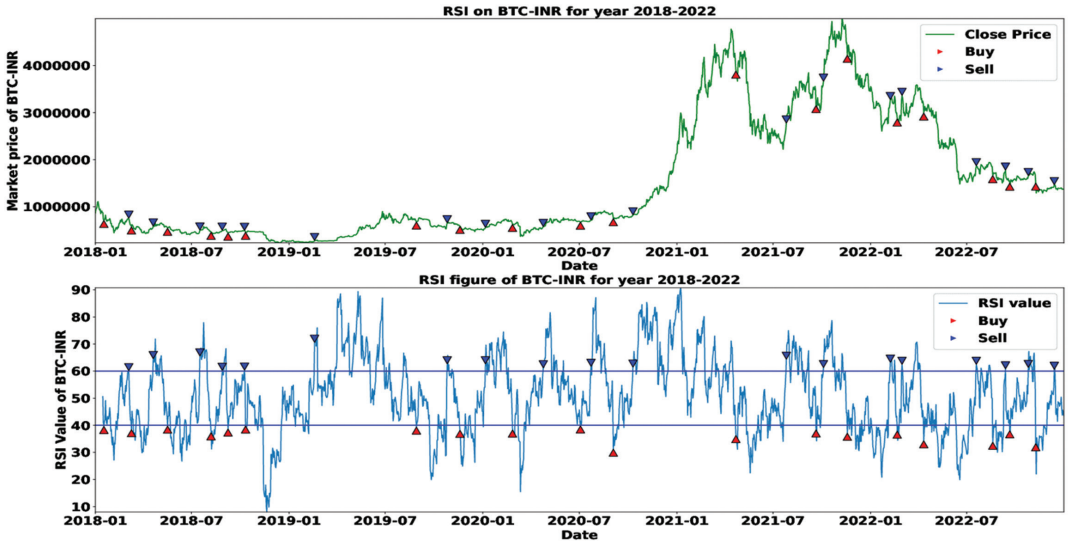


Fig. 8. Transactions as per RSI strategy over five years consolidated on Bitcoin price data

4.3.4. Analysis of PSAR strategy

PSAR is a trend indicator denoted with dots, wherein the dots below the CP show *Uptrend* and, the dots above CP indicates *Downtrend*. Fig. [9], depicts the application of PSAR on the Bitcoin data of 2018-22 consolidated. From gives profits in *Uptrend* and that, strategy using PSAR gives profits in *Uptrend* and losses in *Downtrend* invariantly. From Fig. [4,5], in *Fluctuating* scenario PSAR yielded losses which signify that PSAR is not suitable for *Fluctuating* market. From Fig. [9], it can be observed that PSAR is not giving *SELL* signal that gives maximum profit, however, PSAR can avoid huge losses in *Downtrend* because of quick *SELL*. Hence, this strategy can be treated as a safe strategy with marginal profits.

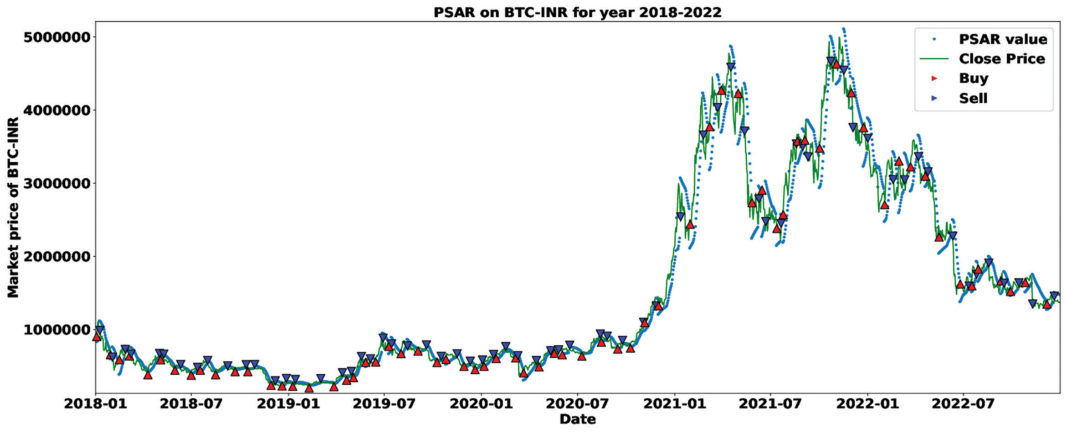


Fig. 9. Transactions as per PSAR strategy over five years consolidated on Bitcoin price data

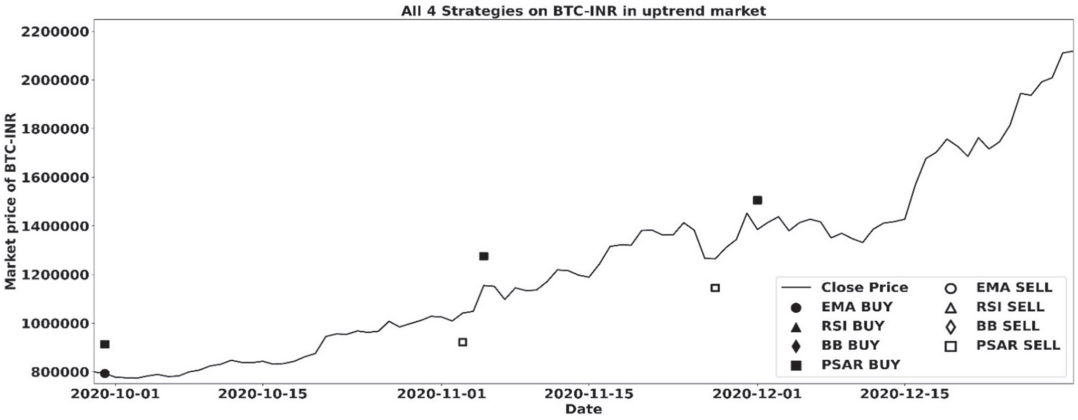
4.4 ANALYSIS OF STRATEGIES BASED ON INDIVIDUAL INDICATORS IN DIFFERENT MARKET SCENARIOS

The *Uptrend* scenario from Oct 2020 to Dec 2020 is considered here for the analysis of the behavior of the strategies based on individual indicators. Fig. [10. a], convey that only EMA and PSAR are performing transactions during this period. Further, it can be seen that PSAR reacts to the fluctuations within the trend, which may be treated as a minimal loss pattern with marginal profits. In *Uptrend* market, the traders who are willing to take risks may consider EMA whereas PSAR provides loss protection. BB, and RSI being momentum indicators, cannot capture this *Uptrend/Bullish* market signs. From April to July 2021, *Bitcoin* showed a *downtrend/bearish* sign which is shown in Fig. [10. b], RSI and BB are unable to identify the *Downtrend* based on the strategies used and hence ended up in loss-

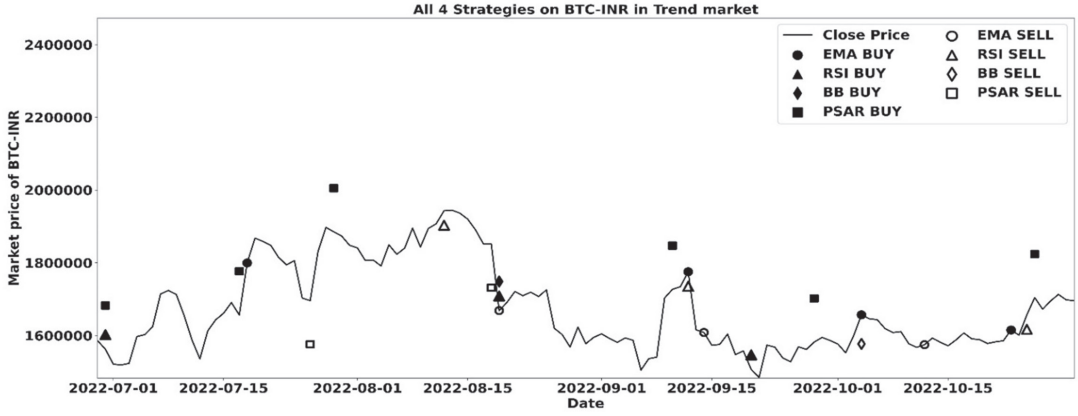
es. PSAR is quicker than EMA in generating *SELL*, giving loss protection. However, in search of market turnover, it is generating false signals during *Downtrend* causing multiple transactions and ending up in further losses.

Fluctuating market in the first quarter of 2022 is considered for comparison of the smart strategies in individual indicators as shown in Fig. [10. c]. It exhibits that all the indicators react in *Fluctuating* market.

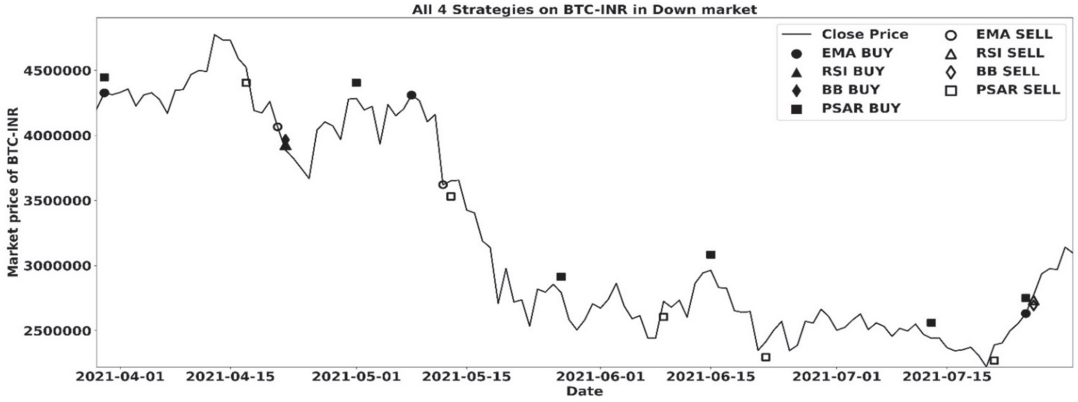
It is because some of the changes in this time frame are abrupt and some of them are gradual changes. With clinical observation, and based on the quantitative analysis it can be observed that RSI provides the best profit margins.



(a) Smart strategies based on individual indicators in *Uptrend/Bullish* trend



(b) Smart strategies based on individual indicators in *Fluctuating* markets



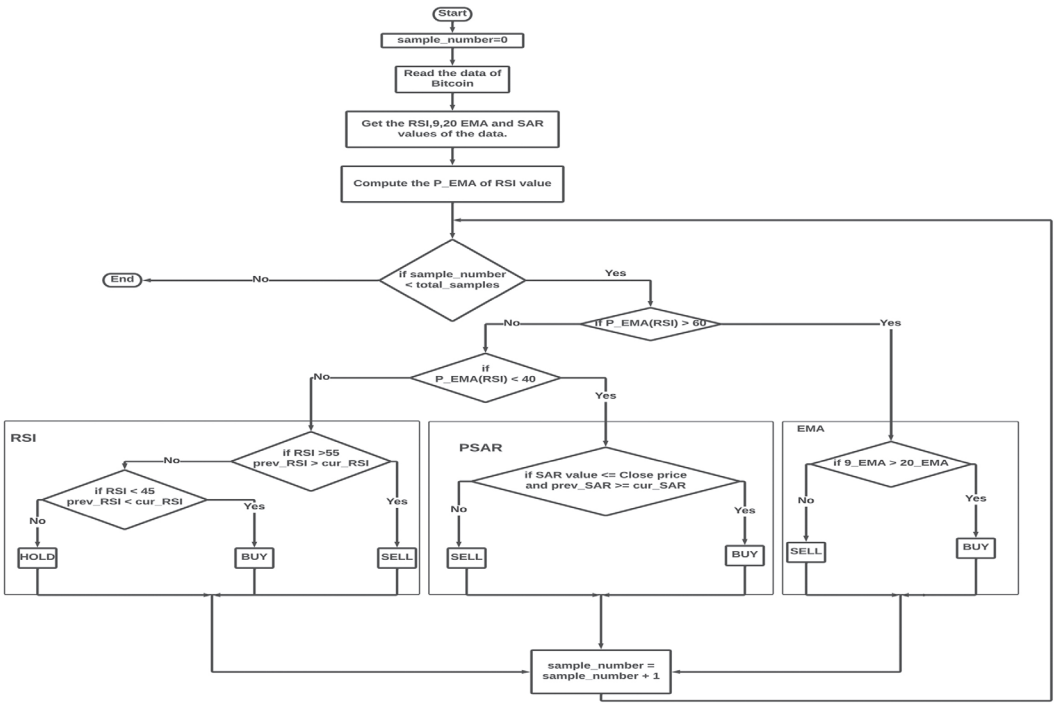
(c) Smart strategies based on individual indicators in *Downtrend/Bearish* trend

Fig. 10. Analysis of strategies in different market scenarios

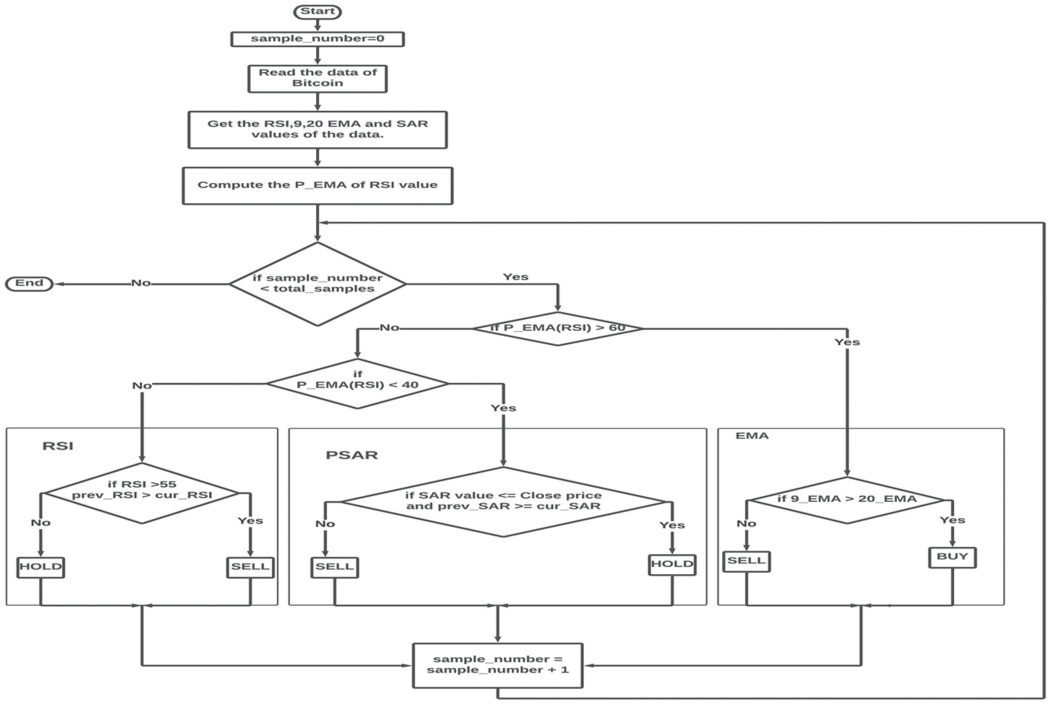
5. STRATEGIES BASED ON MULTIPLE TECHNICAL INDICATORS

Based on the insights obtained in Section [4], imply that strategies devised based on multiple indicators which are used in appropriate scenarios might produce better profits when compared to strategies based on single technical indicators. EMA for *BUY* signal in *Uptrend*, PSAR for *SELL* signal in *Downtrend*, and RSI in *Fluctuating* scenario for both *SELL* and *BUY* are found to be suitable for obtaining higher profits. In the current work, a Multi-Indicator based Hierarchical Strategy

(MIHS) is proposed, wherein the identification of the market is performed before strategically applying signals shot by the respective indicators in corresponding scenarios. The market scenario is identified hierarchically by applying EMA on RSI and further applying dual threshold for identification of *Uptrend* and *Downtrend*. A strategy is designed based on these inputs whose flowchart is shown in Fig. [11. a], this algorithm has two variants where EMA9 and EMA7 are applied on RSI for the identification of *Uptrend* and *Downtrend* scenarios in the market. If the EMA of RSI is greater than 60, the market is considered to be in an *Uptrend* and if it is less than 40, the market is assumed to be in *Downtrend*. In the *Fluctuating* scenario, RSI is used for trading with thresholds 45 for shooting the *BUY* signal when RSI is crossing from below and 55 for shooting the *SELL* signal when RSI is crossing from above. The transactions performed as per MIHS for bitcoin price data with EMA9 and EMA7 applied on RSI to identify the scenario of the market are shown in Figs. [12,13] respectively.



(a) Smart strategy with multiple technical indicator



(b) Flowchart of MIHCS

Fig. 11. Flowcharts of MIHS, and MIHCS strategies

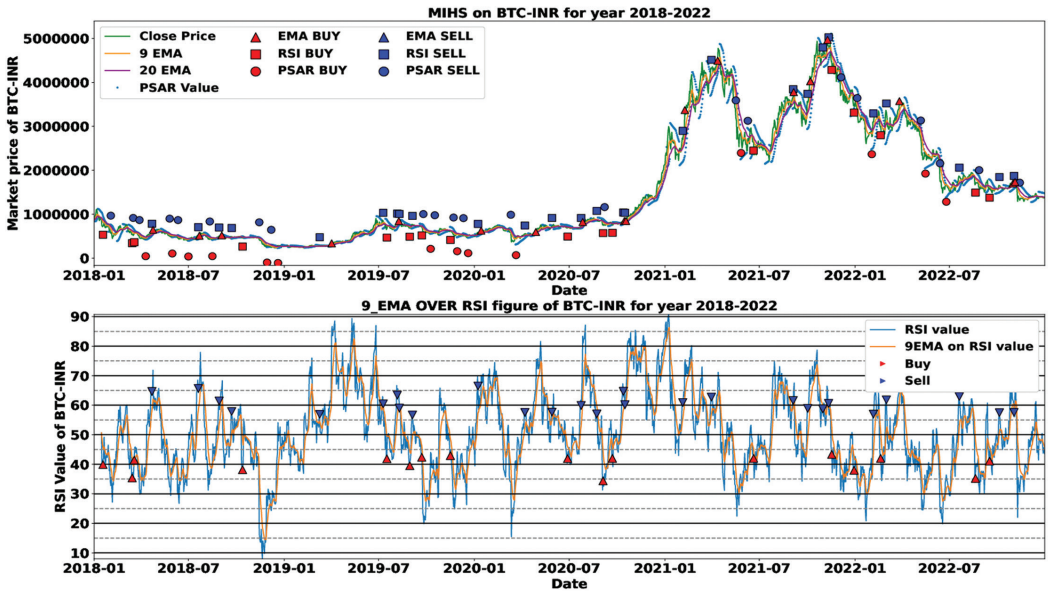


Fig. 12. Transactions performed as per MIHS with EMA9 applied on RSI on Bitcoin during 2018-22

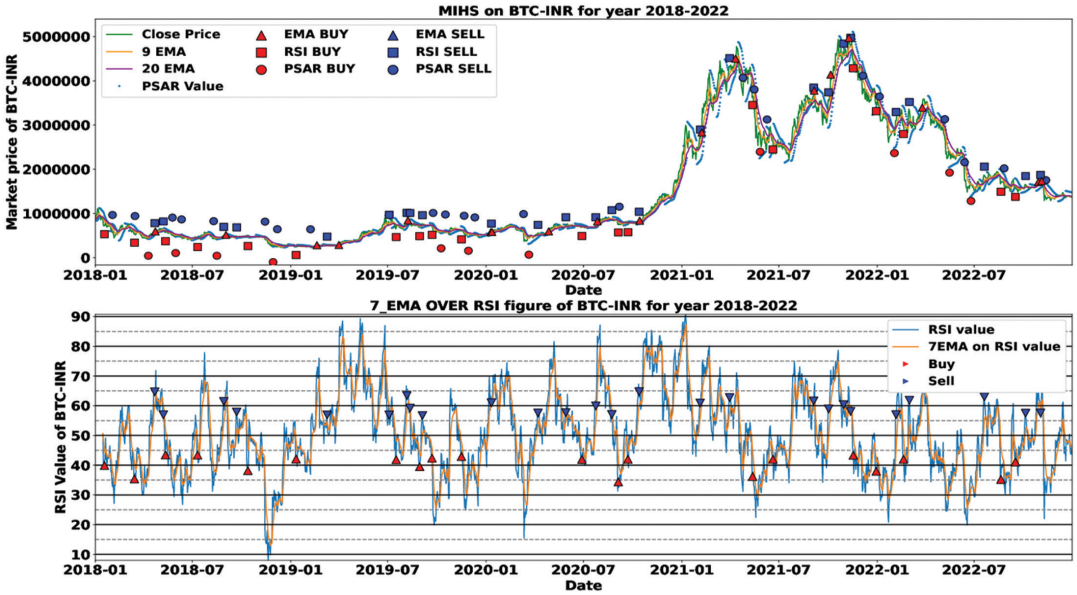


Fig. 13. Transactions performed as per MIHS with EMA7 applied on RSI on Bitcoin during 2018-22

EMA9 when applied to RSI is not quick enough to track the actual RSI and as a result of which there is a delay in identifying the scenario of the market. It affected the *BUY* signal in the *Uptrend* to be delayed resulting in lower profits and a delayed *SELL* signal in *Downtrend* results in higher losses. The lagging effect can be reduced by applying EMA7 on RSI instead of EMA9 which is evident from Figs. [13,12]. MIHS can pick up the fluctuations at a minor level and still exhibits good loss protection as PSAR is quick enough to give the *SELL* signal. In several cases where the market has shown *Fluctuating* scenario, MIHS able to produce profits, however, Figs. [12,13], infer that when the market is moving from the *Uptrend* scenario to the *Downtrend* or *Fluctuating* market scenario, EMA takes time to match the actual RSI because of its lagging nature, during which RSI/PSAR gives the *BUY* signal in *Fluctuating* market caused heavy losses. Removing the *BUY* signal in *Fluctuating* and *Downtrend* scenarios can avoid such losses at the cost of the profits obtained during the *Fluctuating* market scenarios. As Bitcoin is a standard coin fluctuating market is not observed too often. Even if such a scenario exists there may not be good profit as the percentage of change is not too huge. In search of capturing these minor profits, one may end up with losses if there is an abrupt *Downtrend*. With this constraint, a multi-Indicator-based Hierarchical Constrained Strategy (MIHCS) is devised whose flowchart is shown in Fig. [11. b]. The transaction performed as per MIHCS using EMA9 and EMA7 on RSI is shown in Fig. [14,15], from which it can be seen that in a *Downtrend* the Buy signals that are shot by RSI and PSAR are neglected because of which is a good loss protection policy adopted, however, it cost of the profits during the *Fluctuating* markets.

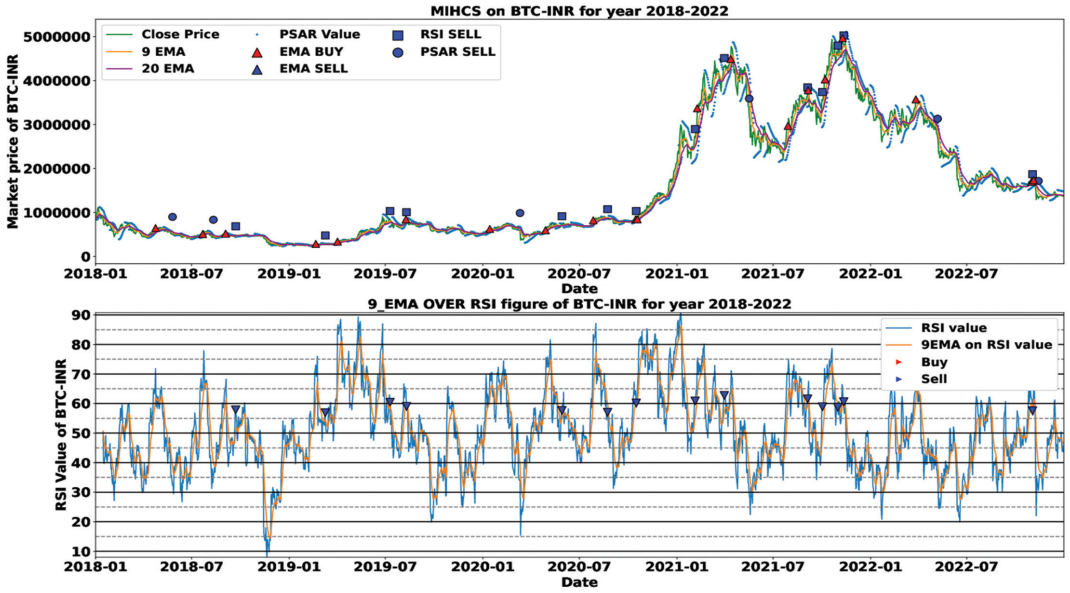


Fig. 14. Transactions performed as per MIHCS with EMA9 applied on RSI on Bitcoin during 2018-22

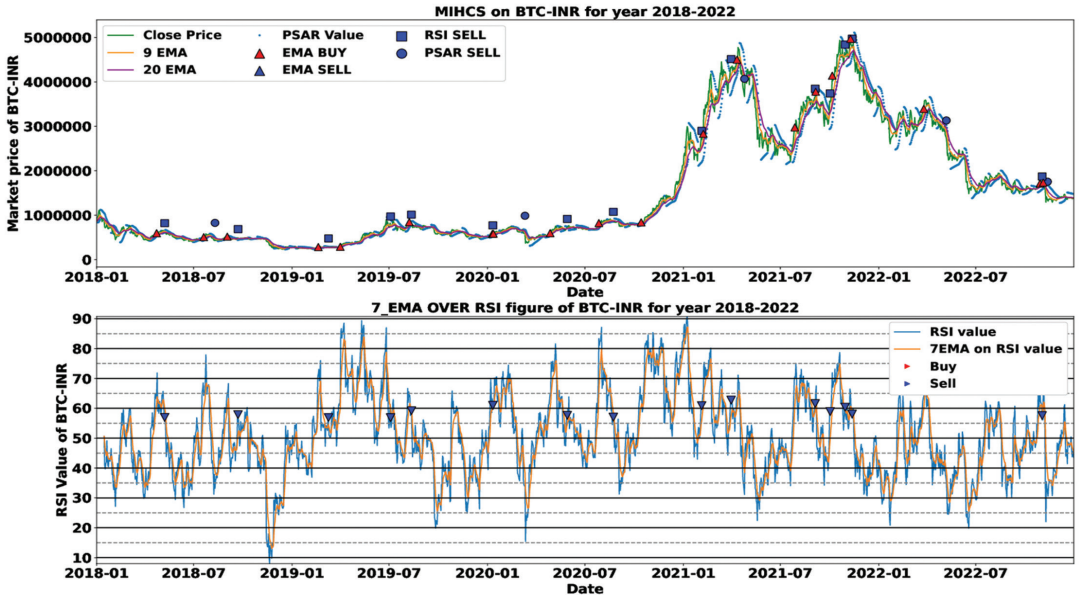


Fig. 15. Transactions performed as per MIHCS with EMA7 applied on RSI on Bitcoin during 2018-22

6. RESULTS AND DISCUSSIONS

The price data of the Bitcoin is obtained from Yahoo Finance at a sampling rate of 1 sample per day. The sampling rate is chosen to be 1 sample per day as the proposed algorithm and the Bitcoin market are more appropriate for long-term investments. An initial investment of 1 *lakh* INR is assumed to be invested, and trading is performed by implementing *BUY* and *SELL* signals as per the strategies on the bitcoin price data. If at the end of the time frame, after the last transaction, a *BUY* signal is shooted but *SELL* is not shooted, the performance metrics are computed assuming the closing price of Bitcoin as the selling price Three performance metrics are Profit Percentage (PP), Net Profitability Percentage (NP), and Number of Total Transactions (NT) which are defined in Equations below, are computed for an empirical comparative study of the strategies based on the followed by *SELL* signal. If *SELL* is not performed after the *BUY* signal, at the end of the year, the profit percentage is computed based on the closing price.

$$\text{PP} = \frac{SellingPrice - BuyingPrice}{BuyingPrice} * 100$$

$$\text{NP} = \frac{Number\ of\ Profitable\ transactions}{Total\ number\ of\ transactions}$$

NT = Total number of *BUY* followed by *SELL* signals

The Profit Percentage, Net profitability percentage, and Number of total transactions of the strategies using individual indicators for the five years (2018-22) and all years consolidated are shown in Figs. [16], and in Table [2].

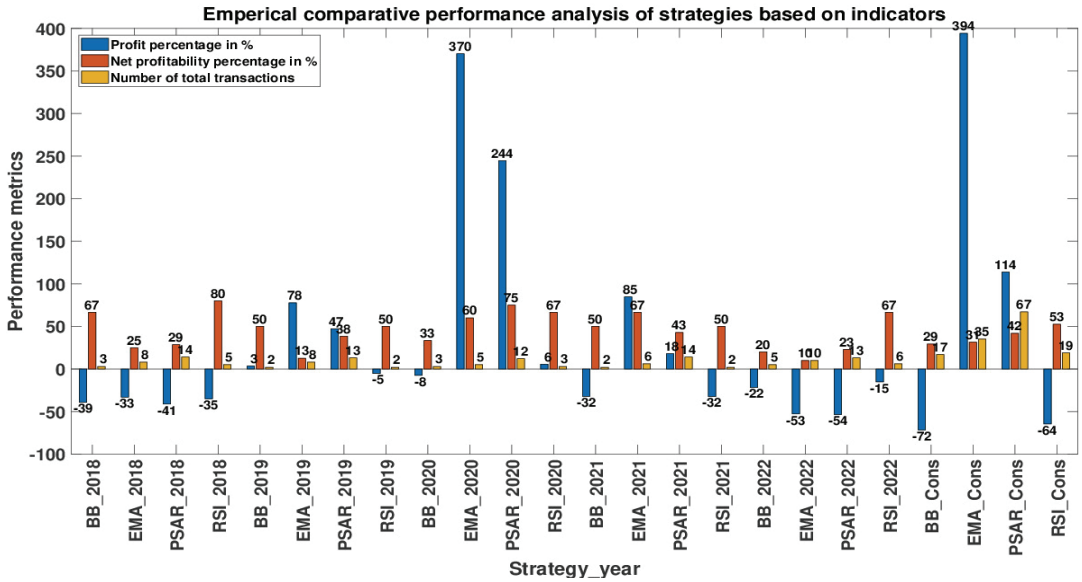


Fig. 16. Performance analysis of smart strategies based on individual indicators over years

Table 2. An empirical comparative study of strategies based on individual indicators

Strategy Name	2018			2019			2020			2021			2022			All years Consolidated		
	PP	NP	NT	PP	NP	NT	PP	NP	NT	PP	NP	NT	PP	NP	NT	PP	NP	NT
BB	-39.2	66.7	3	3.5	50	2	-7.5	33.3	3	-32.1	50	2	-21.8	20	5	-71.7	29.4	17
EMA	-33.1	25.0	8	77.6	12.5	8	370.2	60	5	84.6	66.7	6	-52.8	10	10	394.1	31.4	35
PSAR	-41.2	28.6	14	47.2	38.5	13	244.5	75	12	18.0	42.9	14	-53.6	23.1	13	113.8	41.8	67
RSI	-35.1	80.0	5	-5.4	50	2	5.5	66.7	3	-32.3	50	2	-15.3	66.7	6	-64.5	52.6	19

From Table [2], the EMA strategy is the top performer among the four indicators with a profit percentage of 394.13% followed by PSAR with 113%. This shows the supremacy of trend indicators over the momentum indicators for the bitcoin market as it is a standard coin, and doesn't show *Fluctuating* market signs too often. The number of transactions using PSAR is 67 during 2018-22 whereas for EMA it is 35, however, the profit percentage is higher in EMA which is because of the quick response of PSAR towards the fluctuations when compared to EMA. From, Fig. [10. a], in *Uptrend*, the number of transactions is shot by PSAR strategy, whereas EMA did not respond to those minor changes. In the case of RSI and BB, the number of transactions is small, which is an indication of improper thresholds considered, which are popular selections but are not suitable for higher profits bursting the common myths of the technical indicators. A higher Net profitable percentage for RSI with fewer transactions compared to others is evidence that when proper thresholds or adaptive thresholds are considered, RSI can perform better and the profit percentage obtained in this work, is not indicative of the true potential of RSI. BB consistently resulted in losses/little profits whose results may be improved if the parameters are fine-tuned. The large gap in the profit percentage of EMA and RSI in the consolidated case is not only due to the influence of *fluctuating* market but also because of changes in profit percentage in the initial years that make the difference in the investment for further years. In 2018, and 2022 all the strategies ended up in losses as the market has seen a huge downfall, and technical indicators are susceptible to such changes because of external factors. In 2021, although the market has seen several fluctuations, it is a slow *Uptrend* and rapid *Downtrend* scenario which is properly captured in EMA and PSAR strategies as they are trend indicators and further the momentum indicators failed to produce *BUY* signals in the *Uptrend* scenario. It can also be deduced that the strategies built using single technical indicators are not very efficient, and further, they can be improved by designing such algorithms using multiple indicators from the inferences made by observing the market scenarios studied in this work.

Based on such observations two algorithms namely MIHS and MIHCS are devised with two variants each where EMA9 and EMA7 are applied on RSI for the sake of identification of the market scenario and their performance is compared with EMA strategy over five years individually and consolidated and presented in Table [3].

Table 3. Performance analysis of Multi indicator based strategies for five years individually and consolidated

Name	2018			2019			2020			2021			2022			All years Consolidated		
	PP	NP	NT	PP	NP	NT	PP	NP	NT	PP	NP	NT	PP	NP	NT	PP	NP	NT
EMA	-33.1	25.0	8	77.6	12.5	8	370.2	60.0	5.0	84.6	66.7	6	-52.8	10.0	10.0	394.1	31.4	35.0
MIHS9	-61.1	25.0	12	82.3	30.0	10	281.5	62.5	8.0	14.0	33.3	9	-49.1	40.0	10.0	154.5	40.8	49.0
MIHS7	-35.3	36.4	11	109.7	27.3	11	285.4	71.4	7.0	17.3	30.0	10	-45.0	40.0	10.0	437.5	42.9	49.0
MIHCS9	-38.2	0.0	3	130.2	33.3	3	197.8	50.0	4.0	27.8	42.9	7	-42.7	0.0	3.0	256.3	35.0	20.0
MIHCS7	-17.3	33.3	3	152.9	33.3	3	204.0	66.7	3.0	40.9	42.9	7	-38.7	0.0	3.0	701.8	45.0	20.0

From Table [3], MIHCS with EMA7 applied on RSI gives the best results among the strategies implemented with a profit percentage of 701.77%, followed by MIHS with EMA7 applied on RSI with a profit percentage of 437.48%, whereas EMA strategy gives 394.13% for five years consolidated. This shows that when multiple indicators are wisely used to devise strategies, they can deliver profits better than those individual indicators. The best performance of MIHCS with EMA7 can be attributed to proper identification of the *Uptrend* and avoiding buying in *Downtrend*. From Fig. [15], on several occasions in *Uptrend*, a delay in *BUY* can be observed which can be viewed as a confirmation of *Uptrend*, and a sequence of *BUY* and *SELL* signals in *Uptrend* although limiting the profit percentage, is the basis of loss protection policy adopted. In *Downtrend*, there is a quick *SELL* signal to avoid/limit the probable losses. MIHS and MIHCS with EMA9 are performing poorly when compared to the counterpart MIHS and MIHCS with EMA7 because of the more lagging effect of EMA9 to that of EMA7.

The transactions performed by strategies based on multiple indicators and EMA in 2018, 2019, 2020, 2021, and 2022 are shown in Figs. [17-21] respectively. From these Figs. [17-21], and Table [3], the following inferences can be deduced.

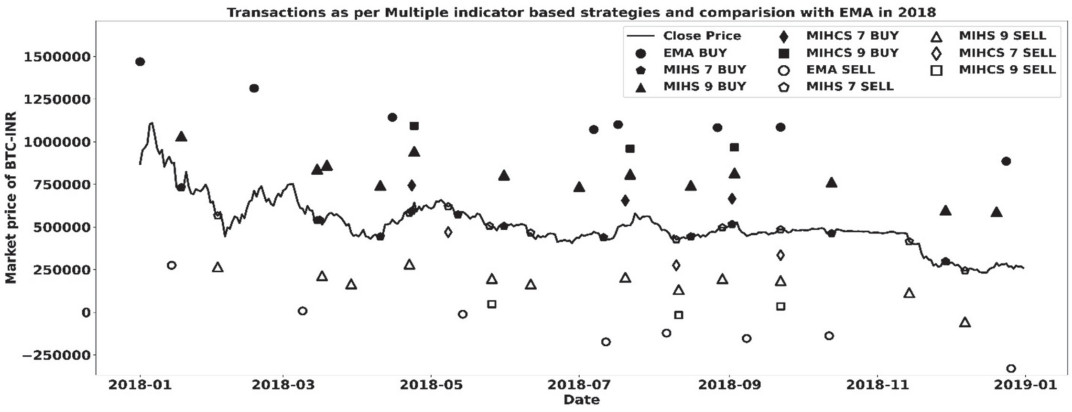


Fig. 17. Analysis of strategies based on multiple indicators and comparison with EMA in 2018

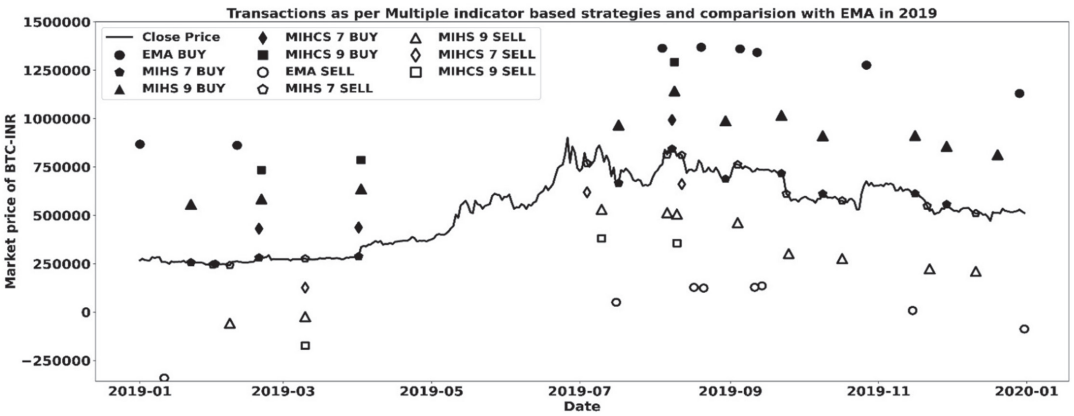


Fig. 18. Analysis of strategies based on multiple indicators and comparison with EMA in 2019

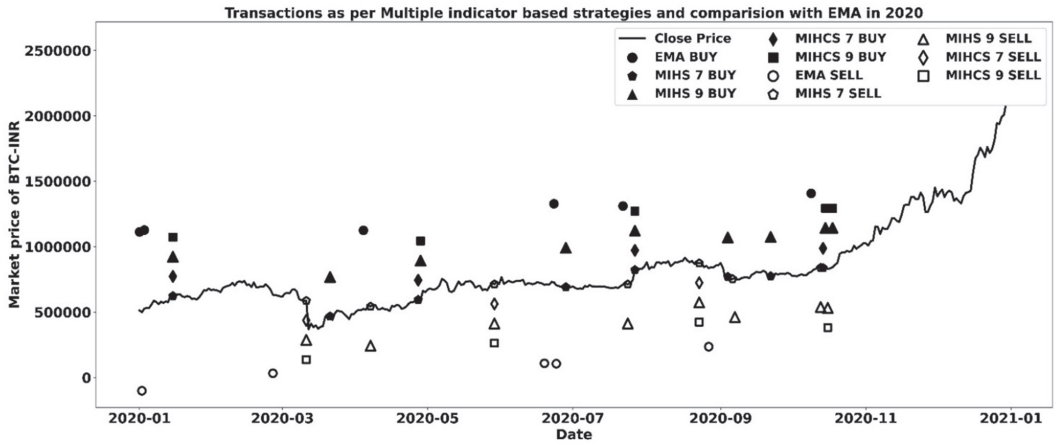


Fig.19. Analysis of strategies based on multiple indicators and comparison with EMA in 2020

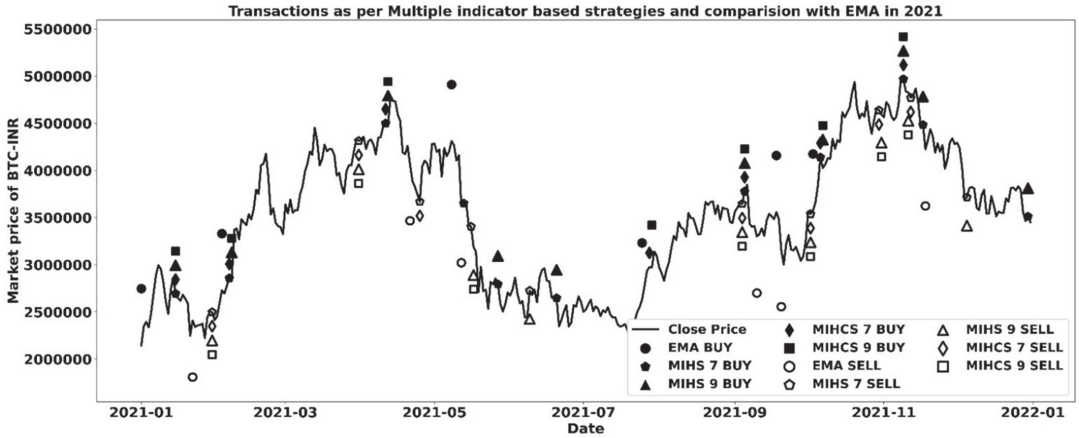


Fig. 20. Analysis of strategies based on multiple indicators and comparison with EMA in 2021

In 2018, MIHCS provided the best performance of -17.25% followed by EMA with -33.10% which shows that the loss protection policy of the MIHCS demonstrated the ability in identifying *Downtrend* with only 3 transactions performed, avoiding losses. MIHS with EMA7 is closely following EMA despite the losses in *Downtrend*, as the net profitable percentage of this algorithm is a little higher when compared to the other cases because of its potential to protect from losses in *Fluctuating* scenarios and deliver profits in *Uptrend*.

In 2019, during the first quarter the market has shown inertia i.e., there are changes around a particular price, during which MIHS with EMA9, and EMA7 tried to capture the *Fluctuating* market scenario, however, these transactions hardly had any impact on the percentage profit. During the second quarter, the market is in *Uptrend* and almost all the strategies considered in Table [3], can capture the *Uptrend* within a window of time, wherein the EMA strategy is the first one to provide the *BUY* signal and MIHCS is the first one to produce the *SELL* signal. All the strategies can capture these market scenarios and more or less almost equal profits, however, in the last quarter of 2019 the market has shown a slow and steady *Downtrend*, during which MIHCS with EMA7 and EMA9 produced only one transaction due to the loss protection policy during the *Downtrend* and able to sustain with the profits, whereas the other strategies performed multiple transactions with consecutive losses that effected the profit percentage.

In 2020, in the first and second quarters, the market has a bit of *Fluctuating* scenario with a hidden *Uptrend*, where EMA performed best because of the hidden *Uptrend*, holding for enough time before selling when compared to MISH which performed multiple transactions in this period to capture the *Fluctuating* scenario neglecting the *Uptrend* to end up with fewer profits. This difference in the profits in the first two quarters made a huge impact as they are included as investments for further transactions causing a huge difference in the profits in the final quarter. MIHCS, because of the loss protection policy in which the *BUY* signal is removed in both *Fluctuating* and *Downtrend* scenarios performed the transactions selectively to end up avoiding the tiny losses/profits and ended with marginal profits because of the difference of investments when compared to the other strategies. Overall, EMA delivered a profit percentage of 370.19% followed by MIHS with EMA7 with a profit percentage of 285.43%. A higher Net profit percentage of 71.43% for MIHS with EMA7 shows that it captures the *Fluctuating* market scenario while there is a hidden *Uptrend* which is the prime reason for this strategy to perform well when compared to MIHCS variants.

In 2021, the market is inconsistent and has seen a steady *Uptrend* and a sudden *Downtrend* which is an ideal scenario for EMA to deliver profits. The MIHS variants although captured the *Uptrend* and saw profits in such cases, they suffer losses during the *Downtrend* because of their quick response of RSI and PSAR involved in these strategies to the fluctuations when compared to the lagging EMA causing more number of transactions with less net profitable percentage. The MIHCS delivered moderate profits due to the delayed *BUY* signal in the *Uptrend* and loss protection policy. Overall, the EMA strategy provides the best result in 2021, with a profit percentage of 84.58%.

In 2022, in the first quarter, the market is in *Fluctuating* scenario where MIHS variants performed fairly well as they are suitable for such cases delivering minor profits. EMA strategy, on the other hand, delivered losses in this timeline. During the only *Uptrend* scenario in the market at the end of the first quarter, EMA is the only strategy to capture the scenario and deliver profits as the multi-indicator strategies suffer with a delayed *BUY* signal. However, these profits obtained became insignificant because of the multiple transactions during the *Downtrend* shot by EMA and MIHS strategies during the second and third quarters of the year. The MIHCS strategies performed relatively better because of the loss protection policy with a profit (loss) percentage of -38.69% for MIHCS with EMA7 and -42.70% for MIHCS with EMA9.

Overall, MIHCS with EMA7 consistently performed over the years because of the loss protection policy, however, still suffers from a slightly delayed *BUY* signal which is because of the lag resulting from EMA with period 7 applied on RSI, and also the profits that can be obtained from *Fluctuating* market are completely ignored. Solving these issues can improve profits further.

7. CONCLUSION AND FUTURE WORK

The current work presents an empirical comparative study of the performance of strategies based on four technical indicators namely Bollinger Bands, EMA, PSAR, and RSI, that have been implemented with popular threshold setups, and tested on the price data of Bitcoin obtained from Yahoo finance for the years 2018-22 individually and also for consolidated for the five years. This analysis has indicated that Trend indicators like EMA, and PSAR with profit percentages of 370 % and 113 % respectively, are much more suitable for principal coins like Bitcoin when compared to the momentum indicators. Proper selection of threshold parameters is very important for the success of momentum indicators and the thresholds that are popular among traders do not provide profits. The strategies have been further analyzed in three scenarios that arise in the market, i.e., *Uptrend*, Downtrend, and Fluctuating. It has been found that strategies using EMA for the BUY signal in *Uptrend*, PSAR for the *SELL* signal in *Downtrend*, and RSI in *Fluctuating* scenario for both *SELL* and *BUY* are found more suitable for obtaining higher profits. Based on such insights, strategies using multiple indicators with a hierarchical approach have been developed,

which identifies the market scenario by applying EMA9 and EMA7 on RSI in two different variants coined as Multi Indicator based Hierarchical strategy (MIHS) with EMA9 and MIHS with EMA7. MIHS with EMA7 has performed better than the EMA strategy with a profitable percentage of 437.48%. MIHS with EMA9 gives a profit percentage of 256.31% which is less when compared to EMA and MIHS with EMA7 strategies because EMA9, when applied on RSI, suffered with a significant lagging. Although MIHS variants have shown the ability to produce profits in *Fluctuating* market scenario, they suffer huge losses because of the delay in the identification of the Downtrend, where the market is misinterpreted as in *Fluctuating* scenario allowed PSAR/RSI to produce *BUY* and consecutive *SELL* signals. To further improve, MIHS variants have been modified by restricting the *BUY* signal only to *Uptrend*, introducing the loss protection policy. These modified strategies have been termed Multi Indicator based Hierarchical Constrained Strategies (MIHCS) with EMA7 and EMA9 in which MIHCS with EMA7 has produced the best profit percentage of 701.77% despite a huge downfall of the market in 2022. Development of a dedicated hardware device that provides *BUY*, *SELL* notifications based on the strategies demonstrated in the current work is considered for Future work.

8. REFERENCES

[1] S. Nakamoto, Bitcoin whitepaper, https://bitcoin.org/bitcoin (accessed: 2023)

[2] Yahoo finance, https://finance.yahoo.com/quote/BTC-USD?p=BTC-USD (accessed: 2023)

[3] S. Hansun, M. B. Kristanda, "Performance analysis of conventional moving average methods in forex forecasting", Proceedings of the International Conference on Smart Cities, Automation & Intelligent Computing Systems, Yogyakarta, Indonesia, 8-10 November 2017, pp. 11-17.

[4] M. J. S. De Souza, D. G. F. Ramos, M. G. Pena, V. A. Sobreiro, H. Kimura, "Examination of the profitability of technical analysis based on moving average strategies in BRICS", Financial Innovation, Vol. 4, No. 1, 2018, pp. 1-18.

[5] P. Praekhaow, "Determination of trading points using the moving average methods", Proceedings of the International Conference for Sustainable Greater Mekong Subregion, Bangkok, Thailand, 1 June 2010.

[6] F. Papailias, D. D. Thomakos, "An improved moving average technical trading rule", Proceedings of the Quantf Research Working Paper Series No. WP01/2014, Physica A: Statistical Mechanics and its Applications, 15 June 2015, pp. 458-469.

[7] M. Tanaka-Yamawaki, S. Tokuoka, "Adaptive use of technical indicators for the prediction of intra-day stock prices", Physica A: Statistical Mechanics and its Applications, Vol. 383, No. 1, 2007, pp. 125-133.

[8] D. F. Gerritsen, E. Bouri, E. Ramezanifar, D. Roubaud, "The profitability of technical trading rules in the Bitcoin market", Proceedings of the Finance Research Letters, Vol. 34, 2020, pp. 101-263.

[9] J. C. P. M'ng, "Dynamically Adjustable Moving Average (AMA) technical analysis indicator to forecast Asian Tigers' futures markets", Physica A: Statistical Mechanics and its Applications, Vol. 509, 2018, pp. 336-345.

[10] J. Chu, S. Chan, Y. Zhang, "High-frequency momentum trading with cryptocurrencies", Research in International Business and Finance, Vol. 52, 2020, pp. 101-176.

[11] M. Resta, P. Pagnottoniand, M. E. De Giuli, "Technical Analysis on the Bitcoin Market: Trading Opportunities or Investors' Pitfall?", Risks, Vol. 8, No. 2, 2020, p. 44.

[12] M. Zatwarnick, K. Zatwarnicki, P. Stolarski, "Effectiveness of the Relative Strength Index Signals in Timing the Cryptocurrency Market", Sensors, Vol. 23, No. 3, 2023, p. 1664

[13] A. UgurSahin, M. Ozbayoglu, "TN-RSI: Trend-normalized RSI Indicator for Stock Trading Systems with Evolutionary Computation", Procedia Computer Science, Vol. 36, 2014, pp. 240-245.

[14] B. Anderson, S. Li, "An investigation of the relative strength index", Banks and Bank Systems, Vol. 10, 2015, pp. 92-96.

[15] J. M. J. Leung, T. T. L. Chong, "An empirical comparison of moving average envelopes and Bollinger bands", Applied Economics Letters, Vol. 10, No. 6, 2003, pp. 339-341.

[16] S. H. M. Yazdi, Z. H. Lashkari, "Technical analysis of forex by parabolic sar indicator", Proceedings of the International Islamic Accounting and Finance Conference, Malaysia, 20-21 November 2012.

[17] J. S. Hunter, "The exponentially weighted moving average", Journal of Quality Technology, Vol. 18, No. 4, 2018, pp. 203-210.

[18] J. W. Wilder, "New concepts in technical trading systems", Trend Research, 1978.

[19] J. Bollinger, "Using bollinger bands, Stocks & Commodities", Vol. 10, No. 2, 1992, pp. 47-51.

Microphone Array Speech Enhancement Via Beamforming Based Deep Learning Network

Original Scientific Paper

Jeyasingh Pathrose
Research Scholar , Department of Electronics and Communication Engineering ,
B.S. Abdur Rahman Crescent Institute of Science and Technology,
Chennai 600048, India
Jeyasingh.p@jasmin-infotech.com

Mohamed Ismail M
Professor and the Dean (Academic Affairs) of B.S. Abdur Rahman Crescent Institute of Science & Technology,
Chennai, 600048 India.
mmismail@crescent.education

Madhan Mohan P
Jasmin Infotech Pvt Ltd,
Chennai, India 600100
madhanmohan.p@jasmin-infotech.com

Abstract *– In general, in-car speech enhancement is an application of the microphone array speech enhancement in particular acoustic environments. Speech enhancement inside the moving cars is always an interesting topic and the researchers work to create some modules to increase the quality of speech and intelligibility of speech in cars. The passenger dialogue inside the car, the sound of other equipment, and a wide range of interference effects are major challenges in the task of speech separation in-car environment. To overcome this issue, a novel Beamforming based Deep learning Network (Bf-DLN) has been proposed for speech enhancement. Initially, the captured microphone array signals are pre-processed using an Adaptive beamforming technique named Least Constrained Minimum Variance (LCMV). Consequently, the proposed method uses a time-frequency representation to transform the pre-processed data into an image. The smoothed pseudo-Wigner-Ville distribution (SPWVD) is used for converting time-domain speech inputs into images. Convolutional deep belief network (CDBN) is used to extract the most pertinent features from these transformed images. Enhanced Elephant Heard Algorithm (EEHA) is used for selecting the desired source by eliminating the interference source. The experimental result demonstrates the effectiveness of the proposed strategy in removing background noise from the original speech signal. The proposed strategy outperforms existing methods in terms of PESQ, STOI, SSNRI, and SNR. The PESQ of the proposed Bf-DLN has a maximum PESQ of 1.98, whereas existing models like Two-stage Bi-LSTM has 1.82, DNN-C has 1.75 and GCN has 1.68 respectively. The PESQ of the proposed method is 1.75%, 3.15%, and 4.22% better than the existing GCN, DNN-C, and Bi-LSTM techniques. The efficacy of the proposed method is then validated by experiments.*

Keywords*: Speech Enhancement, Microphone, Deep Learning, Beamforming, Noise Reduction*

1. INTRODUCTION

Today Speech enhancement (SE) is a pre-processing step in speech recognition that is also required to accommodate the growing demand for higher-quality speech. The speech signal is now used in a variety of systems including speaker identification, speech control, speech-to-text systems, voice over internet protocol (VOIP) accessibility of web applications, and interactive voice response system (IVRS) services. Voice recognition and other speaker activities [1], interaction [2], sound aids [3], and coding of speech all require SE [4]. The SE is a difficult operation when the noisy signal is generated at a lower frequency [5]. The quality of the voice signal should not be sacrificed while designing a speech signal-based system. However, speech signals can be damaged in practice due to a variety of disturbances such as echo, noise in the background, babbling noise babbling sound, and so on. Speech enhancement technology [6] can improve not just the signal-to-noise ratio (SNR) and audio perception of collected speech as well as the resilience of speech

improvement and speaker verification systems. As a result, speech improvement in noisy contexts has gotten a lot of attention.

Speech intelligibility when utilizing in-vehicle speech applications has been impacted by engine noise and other noise sources, such as airflow from electric fans or automobile windows. Inside the car, the reflection of speech waves is employed to communicate particularly between the front and back seat passengers. In addition to in-car disturbances, the quality of speech communication is generally poor. The speech signals are picked up by the microphone and the microphones are placed front seat headrest position.

Hands-free car kits and in-car speech recognition systems increasingly use single-channel noise reduction and beamformer arrays to reduce noise. Microphone array processing focuses on speech improvement and localization, particularly in noisy or reverberant situations [7]. A microphone array is used in the car to increase voice communication quality. [8] A microphone array may gather data in the spatial domain as well as the temporal and frequency domains. The passenger dialogue inside the car, the sound of other equipment, and a wide range of interference effects are major challenges in the task of speech enhancement in-car environment. The noise can be split into interference and desired noise depending on how the noise source and interference path differ from one another. Interference noise makes up the noise in a car. The aim of this paper is to improve speech quality under interference noise conditions. The primary goal of the proposed approach is to improve the speech quality in cars. To overcome this issue, a novel Beamforming based Deep learning Network (Bf-DLN) has been proposed for speech enhancement. The major contribution of the proposed method is;

- Initially, a pre-processing method known as the Least Constrained Minimum Variance (LCMV) is used to pre-process the collected microphone array signals. Consequently, the pre-processed signal is converted into an image using a time-frequency representation.
- The Smoothed Pseudo-Wigner-Ville Distribution (SPWVD) transforms time-domain speech signals into images.
- The convolutional deep belief network (CDBN) receives these transformed signals as input to extract the most pertinent characteristics. By removing the interference source, the desired source is selected using the Enhanced Elephant Heard Algorithm (EEHA).
- The experimental results show that the proposed strategy is effective in removing background noise from the original speech signal.

The rest of the work is organized as follows: Section 2 describes the literature survey, and the problem about the array position inside the car is addressed in section 3. The proposed Source Separation for the car is given in section 4, outcomes are presented in section 5, Section 6 encloses with conclusion and future work.

2. LITERATURE SURVEY

This section outlines the various investigations that have been carried out throughout the year to improve speech signaling. An overview of recent developments in the speech signal is given in this study.

Gentet et al. [9] presented a speech enhancement algorithm it increases the signal-to-noise ratio (SNR). The result showed that the technique has low-frequency noise. Speech intelligibility optimization problem with a fixed perceived loudness restriction is a major drawback.

Alkaher et al. [10] presented the dual microphone speech enhancement for enhancing speech communication in cars. The Pareto optimization decreases the overall speech distortion and relative gain reduction. The result demonstrates the dual-microphone system enhanced howling detection sensitivity. The drawback is that howling sounds may occur even before the speech reinforcement (SR) system reaches instability.

Saleem, N., et al. [11] suggested a Kalman filtering model with an augmented Bidirectional Gated Recurrent Unit (BiGRU) based on residual connections for speech enhancing and recognizing. With the use of the LibriSpeech dataset, the suggested method increased the quality, intelligibility, and word error rates under varied noisy situations by 35.52%, 18.79%, and 19.13%, respectively.

Chuang, S.Y., et al. [12] proposed an improved lite audio-visual speech enhancement (iLAVSE) algorithm for a car-driving scenario. Three stages are involved in the iLAVSE system: data preprocessing, AVSE based on CRNN, and reconstruction. It is also demonstrated that iLAVSE is suitable for real-world scenarios where superior audio-visual sensors might not always be available.

Tao et al. [13] presented the enhanced sound source localization and speech enhancement algorithm which reduce microphone cost and also reduces the complexity. The dual-microphone sound algorithm effectively identifies the sound location, as well as the speech enhancement algorithm, is more resilient and adaptive than the previous method, according to experimental data. The biggest disadvantages are the high cost and high design requirements.

Kothapally, V., et al. [14] proposed a subband spatio-temporal beam former based on DL that can perform speech separation in a care setting with a reduced computation cost and inference time.They showed that the suggested strategy produces improved WER and objective scores using a variety of sub-band configurations.

In Jolad, B and Khanai, R [15], speech signal quality can be improved by using a fractional competitive crowd search algorithm (FCCSA). When the suggested technique is evaluated using the UA speech database, it yields 0.930, 0.933, and 0.934 in terms of accuracy, specificity, and sensitivity.

Qian et al. [16] presented the car speech enhancement system based on a combination of a deep belief network and wiener filtering. The deep belief networks (DBN) parameters are optimized by using the Quantum Particle Swarm Optimization (QPSO) algorithm. The results of the experiment demonstrated that the suggested strategy may successfully reduce the original speech signal's noise signal and improve the speech signal.

Zhou, W., et al. [17], suggested a Meta-reinforcement learning paradigm by concentrating on few-shot learning for improving speech. The experiment's findings demonstrate that in comparison to state-of-the-art DNN-based SE methods under difficult conditions, where the environment noises are varied and the signals are non-stationary, this work achieves at least improvements of 1.3%~12.5% for a single shot and 3.1%~14.3% for a five-shot scenario.

Table 1. Comparison of existing with the proposed method

Method	Advantage	Disadvantage
Speech intelligibility enhancement method for typical in-car [9]	To improve understanding by employing specialized speech transformation methods without altering the initial SNR	High computational complexity
dual microphone speech enhancement [10]	decreases the overall speech distortion and relative gain reduction using Pareto optimization.	howling sounds may occur even before the speech reinforcement (SR) system reaches instability
Kalman filtering model with augmented Bidirectional Gated Recurrent Unit (BiGRU) [11]	for speech enhancing and recognizing.	Do not eliminate echo
improved lite audio-visual speech enhancement (iLAVSE) algorithm [12]	suitable for real-world scenarios	Failure of the sensor to record the visual signal is another source of low-quality visual data.
enhanced sound source localization and speech enhancement algorithm [13]	reduce microphone cost and also reduces the complexity	high cost and high design requirements
DL-based mel-subband spatiotemporal beamformer [14]	decreased computation cost and inference time	unstable speech signal
Fractional Competitive Crow Search Algorithm-based speech Enhancement Generative Adversarial Network [15]	does not investigate in-vehicle speech recognition	high cost and low performance of the microphone
car speech enhancement system based on a combination of a deep belief network and wiener filtering [16]	eliminate the noise signal of the original speech signal and enhance the speech signal	High computational complexity
Meta-reinforcement learning paradigm by concentrating on few-shot learning [17]	decreased computation cost and complexity	higher complexity with an optimum number of layers
Proposed Beamforming-based Deep Learning Network (Bf-DLN)	improve the speech quality in cars, high performance for microphones, less computation and complexity	A limited number of the dataset is used for training and testing

Based on the literature review, a variety of deep learning techniques for improving speech were suggested. However, speech enhancement systems face challenges from unstable voice signals, poor microphone performance, expensive computing, and the problem of echo cancellation in the presence of background noise. To overcome the above challenges this research proposed a novel Beamforming based Deep learning Network (Bf-DLN) and its detailed process is presented in section 4.

3. PROBLEM FORMULATION

Consider a microphone array with N elements that captures D desirable voice signals. The received signal is corrupted by additive noise, which might be made up of white noise from a point source, a diffuse source, or both. The short-time Fourier transform (STFT) domain is used to construct the speech enhancement problem, with a time frame index x and the frequency index f. The microphone signal can be expressed as

$$P(x,f) = [p_1(x,f), p_2(x,f), p_3(x,f) \dots p_n(x,f)]^t \quad (1)$$

Where t represents transpose. The decomposition of the received signal $p(x, f)$ is given as follows:

$$P(x,f) = \sum_{i=1}^{D} U_{i,j}(x,f) + W_i(x,f), i = 1,2, \dots D \quad (2)$$

where $U_{i,j}(x,f)$ stands for the voice signal from the j^{th} speaker as acquired by the i^{th} microphone and $W_i(x,f)$ stands for background and sensor noise.

Assuming that the i^{th} microphone receives anechoic speech signals $E_j(x,f)$ and a time-invariant ATF $H_{i,j}(f)$ (assuming a static scenario), the observed speech signal can be approximated in the STFT domain as a multiplication of the anechoic signals and i^t hmicrophone, i.e.

$$U_{i,j}(x,f) = H_{i,j}(f) E_j(x,f) \quad (3)$$

Designing a beamformer $Z(x,f)$ to ensure that the output signal is accurate is the problem.

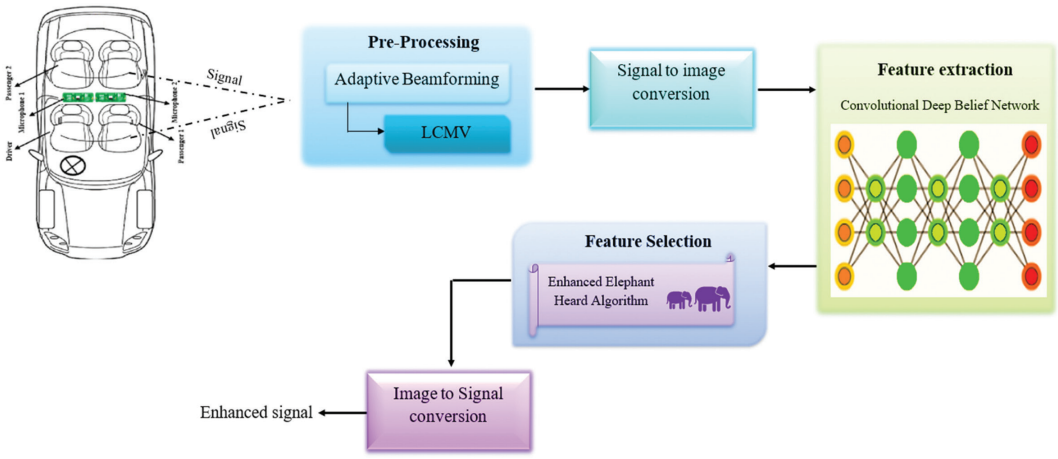
$$L(x,f) = Z^{Ht}(x,f)P(x,f) \tag{4}$$

Where H_t stands for conjugate-transpose, maximizes SNR for each of the S desired sources while preserving a response that is distortion-free for everyone.

4. PROPOSED METHODOLOGY

There In this section, a novel Beamforming based Deep learning Network (Bf-DLN) has been proposed for speech enhancement. The overall block of the proposed methodology is depicted in Fig.1. Initially, the captured microphone signals are pre-processed using an Adaptive beamforming technique named Least Constrained Minimum Variance (LCMV). Pre-processed signals are converted into images via Smoothed Pseudo-Wigner-Ville distribution (SPWVD). Therefore, the convolutional deep belief network (CDBN) is fed these modified images as input in order to extract the most relevant features. Enhanced Elephant Heard Algorithm (EEHA) is used for selecting the desired source by eliminating the interference source.

Fig. 1. Overall block of proposed Methodology



4.1. PRE-PROCESSING VIA ADAPTIVE BEAMFORMING

Beamforming refers to the process by which signals from the microphone array create a beam pattern. Beamforming is a technique for reducing interference signals that come from various noisy directions. The angle and frequency arriving from different directions cancel out the interference signal. It is utilized to increase the volume of speech signals coming from different directions. When beamforming, it generates a signal that is less noisy than the reference microphone's signal while keeping the speech component intact. The LCMV beamformer is a method frequently employed to solve the issue of many desirable speakers' augmentation in noisy environments. It is defined as follows:

$$Z_{LCMV} = \underset{z}{\operatorname{argmin}}\, Z^{Ht}\Phi_w Z, \;\; B^{Ht}Z = k \tag{5}$$

Where, $B = \widetilde{Q_1}, \widetilde{Q_2}, \widetilde{Q_3}, \ldots \widetilde{Q_D}$. The required sources are constrained by an N×D matrix, where $\widetilde{Q_1} = \frac{Q_i}{Q_g}$ for i=1,2,…D and $g\epsilon\{1\ldots.N\}$ is the index of the reference microphone, which is often set to be 1 or the microphone with the best input SNR. The associated restrictions are represented by the vector k=[1 1…1]^t, of length D, and the spatial noise coherence matrix is represented by Φ_w. The well-known formula for solving the problem (5) is given by

$$Z_{LCMV}\Phi_w^{-1}B(B^{Ht}\Phi_w^{-1}B)^{-1}k \tag{6}$$

In response to the necessary speech components, as detected by the microphone, the LCMV beamformer maintains a distortion-free response. When taking into account systems with scattered microphone arrays or arrays with huge apertures, the reference microphone's signal might not be optimal in terms of SNR, or with less speech component power, for all intended speakers. For instance, when one of the desired speakers is placed closest to a microphone $n\epsilon\{1,\ldots,N\}/\{g\}$ in a diffuse noise setting, this is to be expected. SNR for that speaker at the beamformer output is subsequently decreased as a result. The enhanced speech signals are converted into images by using SPWVD, which is briefly discussed in the next section.

4.2. SIGNAL TO IMAGE CONVERSION

Using Smoothed Pseudo-Wigner-Ville distribution, the pre-processed signals are transformed into pictures. At low frequency, Wigner-Ville distributions result in a cross-term and a decrease. SPWVD is used to transform the time-domain filtered speech signals into Time-Frequency Representation (TFR), which overcomes these difficulties. The SPWVD is subjected to independent time and frequency smoothing to attain maximum resolution. The time-frequency resolution of STFT and CWT is problematic, whereas SPWVD has a good resolution. The time-domain signals are transformed into time-frequency representations in order

to follow the spectral domain. TFR stands for time, frequency, and amplitude representation in area simultaneously. To enhance frequency resolution while carrying out the quadratic time-frequency transforming, a sliding window is added to the SPWVD signal in both the frequency and time domains. In both the frequency and time domains, it is possible to separately choose the cross-term lessening window's length and sort. SPWVD has improved time-frequency cluster properties as a result. The mathematical formulation of SPWVD is as follows:

$$\varphi(p,q) = \int_{-\infty}^{+\infty} m(p-p')\varphi(p',q)dp' \quad (7)$$

$$\varphi(p',q) = \int_{-\infty}^{+\infty} n(\tau)\, o\left(p'+\frac{\tau}{2}\right) o^*\left(p'-\frac{\tau}{2}\right) e^{-j2\pi uv} d\tau \quad (8)$$

Where the cross-terms $m(p)$ and $n(p)$ decrease the temporal and frequency domain windows, respectively. It is easy to alter the scales for frequency and temporal domain smoothing. The windows' $m(p)$ and $n(p)$ lengths can be chosen separately.

4.3. FEATURE EXTRACTION USING CONVOLUTIONAL DEEP BELIEF NEURAL NETWORK

Convolutional deep belief networks (CDBNs) are developed from deep belief networks in several ways, including how the building blocks are stacked, the way the network is trained, and even the building blocks themselves. In Fig. 2, the overall structure is depicted, and the main difference is how the building blocks are formed. CRBBMs (Convolutional Boltzmann Machines) are created using Boltzmann machines with convolutional restrictions. The purpose of CRBM is to provide realistically sized images with scaling approaches.

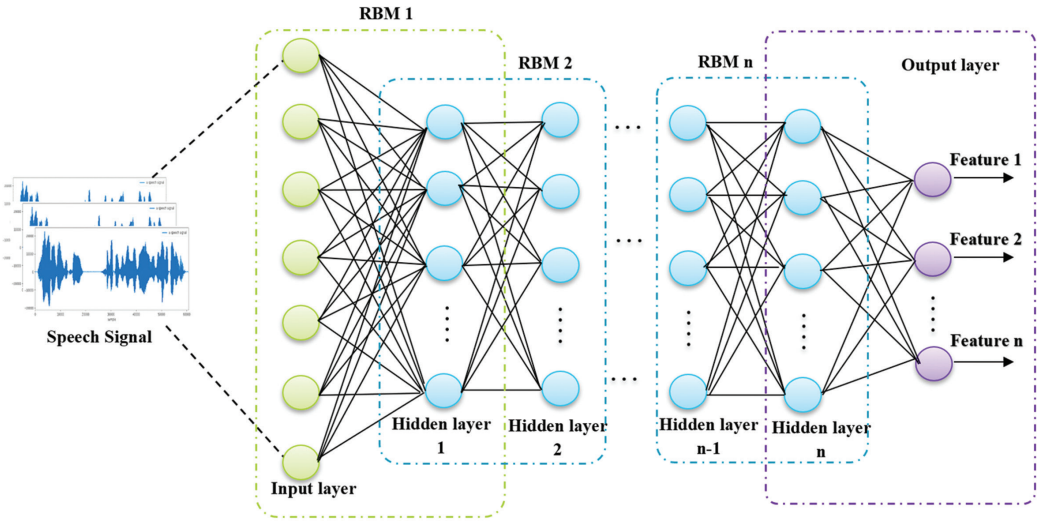


Fig. 2. Architecture of Deep Belief Network

The "convolutional" RBM is an extension of the "regular" RBM in which the weights between the hidden units and the input units are distributed over all locations in the hidden layer. A hidden layer H_l and an input layer I_l make up the two layers of the CRBM. The input units are real-valued or binary-valued, whereas the hidden units are binary-valued.

Assume that the input layer's size is n_{I_l} and the hidden layer's size is n_{H_l}. There are M filters (weights), W_M each of which is convolutional with the input layer, and there are biases b_M for each of the weights as well as c for the input layer. As defined below, the energy function with binary input is

$$E(I_l,H_l) = -\sum_{M=1}^{M}\sum_{y=1}^{n_{H_l}}\sum_{s=1}^{n_w} h_y^s w_s^M i_{y+s-1} - \sum_{M=1}^{M} b_M \sum_{y=1}^{n_{H_l}} h_y^s - c\sum_{x=1}^{n_{I_l}} i_x \quad (9)$$

Accordingly, a CRBM's energy function can be expressed as follows:

$$E(I_l,H_l) = \frac{1}{2}\sum_{x}^{n_{I_l}} i_x^2 - \sum_{M=1}^{M}\sum_{y=1}^{n_{H_l}}\sum_{s=1}^{n_w} h_y^s w_s^M i_{y+s-1} - \sum_{M=1}^{M} b_M \sum_{y=1}^{n_{H_l}} h_y^s - c\sum_{x=1}^{n_{I_l}} i_x \quad (10)$$

Following is a definition of the joint and conditional probability distributions:

$$p(I_l,H_l) = \frac{1}{z}\exp\left(-E(I_l,H_l)\right) \quad (11)$$

$$p(h_{xy}^s = 1 \mid n_{I_l}) = sigmoid((\widetilde{W_M} *_v I)_y + b_M) \quad (12)$$

$$p(i_x = 1 \mid n_{H_l}) = sigmoid(\sum_M (W_M *_f h^M)_y + b_M) \quad (13)$$

Here, $*_v$ denotes valid convolution and $*_f$ denotes full convolution. where, $\widetilde{W_M} = \Delta W_{nw-y+1}^M$. Similar to that of RBM, CRBM is trained using block Gibbs Sampling as an extension of Gibbs Sampling in order to maximize the similarity of distribution between the constructed input layer and the hidden layer and, in that case, obtain the equilibrium state. Convolutional deep belief networks (CRBMs) use probabilistic max-pooling as a fundamental building component.

Calculating the precise gradient for the log-likelihood term is difficult while training convolutional RBMs. Contrastive divergence, however, is an efficient method for approximating the gradient. A sparsity penalty term is added to the log-likelihood goal since a typical CRBM is

significantly overcomplete. The training goal might be stated more precisely as

$$maxi_{W,b,c} = \mathcal{L}_{likelihood}(W,b,c) + \mathcal{L}_{sparsity}(W,b,c) \quad (14)$$

where $L_{sparsity}$ is a penalty term that requires the hidden units to have sparse average activations and $L_{likelihood}$ assesses how well the CRBM approximates the distribution of the input data. The "capacity" of the network can be understood as being limited by this sparsity regularization, which frequently produces feature representations that are simpler to understand. We stack the CRBMs to create a convolutional deep belief network once the parameters for each layer have been trained. Each training involves training the RBM of the lowest layer through each subsequent layer until the top layer is reached. In order to achieve its standard convolutional layer, CBRN uses a 3×3 convolutional kernel with 64 channels. Table 2 displays the CBRN Net hyper parameter.

Table 2. Hyper parameter setting

Parameter	Value
No. of Neurons	512
Learning rate	0.02
Activation function	ReLu
No. of epochs	50
Batch size	100 -250
Dropout	0.03

There are several hyperparameters to tune, including the number of neurons, activation function, optimizer, learning rate, batch size, and epochs. Typically, the connection weights of a neural network serve as its parameters. During the training phase, these traits are revealed in this circumstance. Finally, CDBN feature learning is unsupervised, allowing for extensive use of unlabeled images.

4.4. FEATURE SELECTION VIA ENHANCED ELEPHANT HEAD ALGORITHM

The Elephant Herd Algorithm (EHA) algorithm is modeled by the behavior and way of life of elephants. EHA is a heuristic intelligence system based on elephants' nomadic lifestyles. Elephants exhibit social behavior and have a complicated structure of females and calves. The EHA algorithm selects the most pertinent features from the extracted speech image features.

The number of elephants in this algorithm represents the features that were taken from the input layer; the most pertinent features are the best female elephant of the clan after the matriarch has passed away; and the irrelevant features represent the male elephants with the lowest fitness value. The suggested Enhanced Elephant Herd Algorithm's flowchart is shown in Figure 3. Figure 3 shows the flowchart for the proposed Enhanced Elephant Herd Algorithm.

A group of elephants is made up of several clans, each of which is led by a matriarch who may also have calves or other related females under her care.

Following are some of the algorithm's recommended rules: Elephants live in clans, and each tribe has a set number of elephants. Each clan also has a matriarch, who is the clan's chief (the fittest elephant of the clan). A predetermined number of elephants (worst candidates) must depart the clan each generation, and all the elephants of a clan live together under the authority of the matriarch. Clan update and separation operators make up the two stages of the Elephant Herding Optimization algorithm.

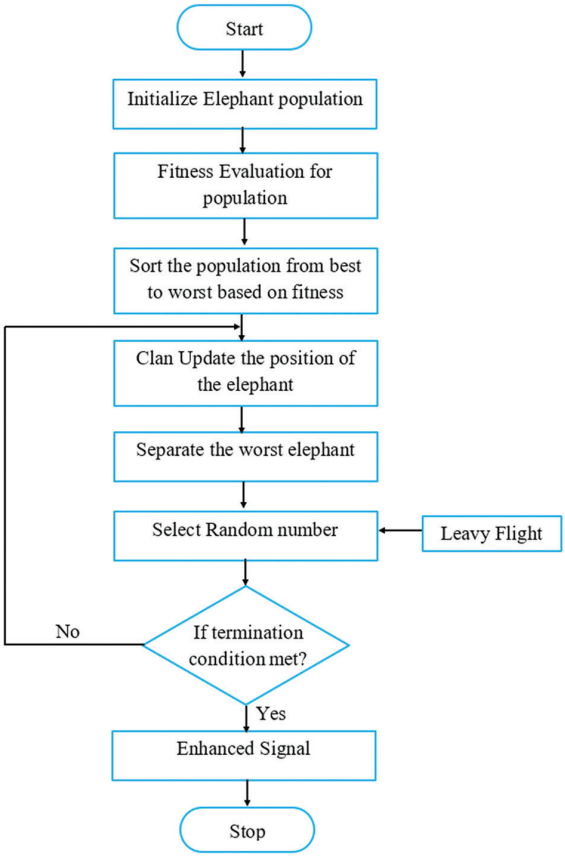


Fig. 3. Flow Diagram of the proposed Enhance EHO

The number of elephants in this algorithm represents the features that were taken from the input layer; the most pertinent features are the best female elephant of the clan after the matriarch has passed away; and the irrelevant features represent the male elephants with the lowest fitness value. The suggested Enhanced Elephant Herd Algorithm's flowchart is shown in Figure 3. Figure 3 shows the flowchart for the proposed Enhanced Elephant Herd Algorithm.

A group of elephants is made up of several clans, each of which is led by a matriarch who may also have calves or other related females under her care. Following are some of the algorithm's recommended rules: Elephants live in clans, and each tribe has a set number of elephants. Each clan also has a matriarch, who is the clan's chief (the fittest elephant of the clan). A predetermined number of elephants (worst candidates) must depart the clan each generation, and all the elephants of a clan live together under the authority of the matriarch. Clan update and

separation operators make up the two stages of the Elephant Herding Optimization algorithm.

The entire population of elephants is initially split up into 'y' clans. Each elephant m_x denotes the new position is influenced by the matriarch m_x. The clan m_x elephant 'y' can be determined using

$$P_{n,m_{x,y}} = P_{m_{x,y}} + \alpha \times \left(P_{best,m_x} - \lambda_{m_{x,j}}\right) \times L \quad (15)$$

where [0,1] is a scaling factor, $P_{best,\ m_x}$ is the location with the best fitness value inside clan "x", and $P_{n,m_{x,y}}$ represent the old and new positions of elephant "y" in clan x, respectively. With a normal distribution and a value between [0, 1], L is a random number. For each clan, the best elephant is determined using

$$P_{n,m_{x,y}} = \beta \times P_{ct,m_x} \quad (16)$$

where $\beta \in [0,1]$ is a scaling factor that defines how the position of the clan leader $P_{n,m_{x,y}}$ will change for the following iteration depending on the effect of the clan center P_{ct,m_x}. Eq. (12) is evaluated to determine a clan center's value:

$$P_{ct,m_x,k} = \frac{1}{N_{m_x}} \times \sum_{y=1}^{N_{m_x}} P_{ct,m_x,k} \quad where\ 1 \le k \le K \quad (17)$$

Where the number of elephants in the clan is signified as N_{m_x}, the k^{th} dimension of an individual elephant. In Eq. (16), the update of the matriarch position is related to the information of all members of the clan.

The worst solution individuals are replaced by randomly initialized individuals during the separation procedure. It expands the population of elephants and enhances their capacity for exploration. The least valuable elephants in each tribe are relocated to the position indicated by

$$P_{w,m_x} = P_{Min} + (P_{Max} - \lambda_{Min} + 1) \times L \quad (18)$$

where P_{w,m_x} is the position with the worst fitness value in clan 'x'; P_{Min} and P_{Max} are the upper and lower bound of the elephant's position, respectively; L is a random number with a normal distribution in the range [0, 1].

The slower convergence rate that results from using random numbers is due to problems like lack of exploitation and random replacement of the poorest person. To address this problem, the LF mode is used with the EHO. The LF is represented by,

$$LF(L) = \begin{cases} 1 & L < 1 \\ (L)^{-F} & L \ge 1 \end{cases} \quad (19)$$

Equations 18 and 19 are combined to progress EHA with LF as follows:

$$P_{w,m_x} = P_{Min} + (P_{Max} - \lambda_{Min} + 1) \times LeF(L) \quad (20)$$

As a result, the EEHA specifies the following as the most important features:

$$RF_x = \{rf_1, rf_2, rf_3, \dots\dots, rf_n\} \quad (21)$$

The features that were randomly chosen are then added together after the inputs given are multiplied by the feature vectors. The mathematical representation of the input layer is,

$$I_x = \sum_{x=1}^{n} RF_x w_x + B_x \quad (22)$$

The input features are shown as RF_x, the weight values are shown as w_x, and the bias value is shown as B_x where the IL is shown as I_x. The enhanced image of a speech signal is converted into signals by using the Inverse SPWVD technique.

5. RESULT AND DISCUSSION

These sections provide details of the experiment and discuss the simulation's results. In the following section, the effectiveness of a speech enhancement system based on the proposed Beamforming-based Deep Learning Network (Bf-DLN) methodology is assessed. The study includes three types of speech: speech with ambient noise, speech with a source of interference, and speech with both sources of interference and ambient noise. The required interference speech is used as an input source, and MATLAB is used to display the findings.

5.1. EXPERIMENTAL SETUP

Two unidirectional microphones in a silent car were used to record recorded speech at an 8kHz sample rate to create the inputs. Then, to create realistic in-vehicle noisy speech signals, actual in-car noise that was recorded using the identical setup with the car driving in a typical motorway condition was mixed with the clean speech. The center of the vehicle is where the two microphones are situated. The desired source data and interference data are provided by two speakers, SP1 and SP2, respectively. Both microphones are mounted on a car's rear unit layout, 0.1 meters apart from the speakers. The center of the back unit holds the microphone array. The intended source is set to speaker 1, and the interfering source is set to speaker 2.

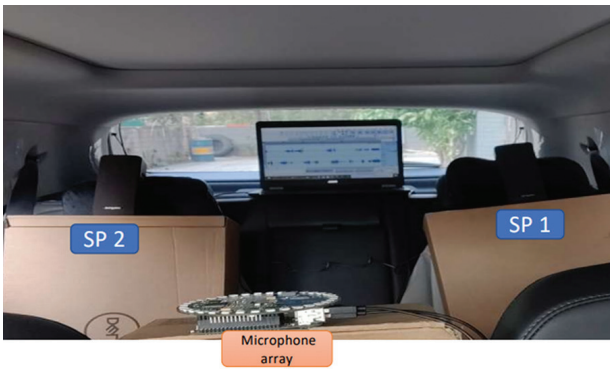


Fig. 4. Realtime data collection setup

With various combinations of input sources at various angles, 30 sets of data in all were obtained.

The parameters used in the suggested method are shown in Table 1. 16 KHz sampling is used for the speech. With segSNR values roughly ranging from -8 dB to -3 dB, noise degrades the signals under various circumstances.

Table 3. Simulation Parameters

Characteristic	Parameter
No. of sources	2
Source classifications	source of speech
Number of mics	2
Sampling rate	8kHz
Size of the FFT window	512

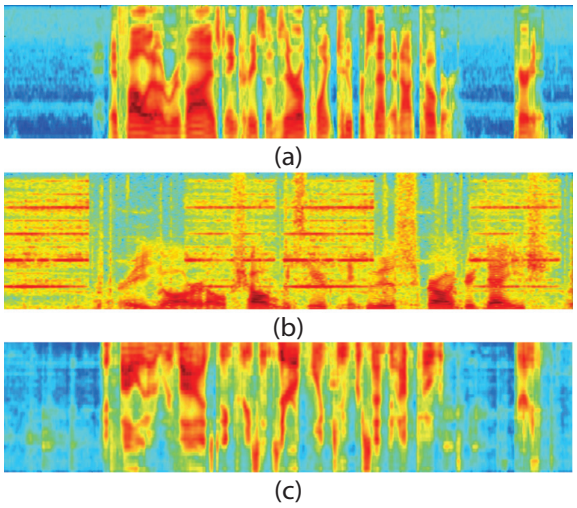


Fig 5. Spectrograms (a) clean image (b) Noisy image (c) Signal enhanced by the proposed method

Figure 5 displays the spectrograms for (a) clear speech, (b) noisy speech and (c) signal increased by the suggested approach. By comparing (b) and (c), it is clear that the suggested Bf-DLN effectively enhances the noise components by demonstrating the method's efficacy.

5.2. EXPERIMENTAL RESULT

The effectiveness of the proposed technique is assessed in comparison to existing methods using the perceptual assessment of speech quality (PESQ), short-time objective intelligence (STOI), segmental SNR improvement (SSNRI), and signal-to-distortion ratio (SDR). The comparative analysis section evaluates the effectiveness of both the existing and the suggested technique. The effectiveness of the various current approaches is evaluated in this section. In addition, we provide performance under various speech and noise SNRs to provide a more complete evaluation of speech quality. Performance scores such as PESQ, STOI, SSNRI, and low SDR reflect improved performance.

Increased speech signal quality and intelligibility are a benefit of GCN [18], but it predicts lower sound quality when used with non-parallel data. On the other hand, DNN-C [19] has the advantage of improving the corresponding noisy input, and all of the channel-wise enhanced outputs are fed into a DNN fusion model to produce a practically clean signal. However, it performs poorly for improving the signal. Additionally, Bi-LSTM [20] can extract local and global characteristics and achieves competitive results; its main drawback is a high computational cost.

Table 4. Comparison of proposed with an existing method for Microphone 1

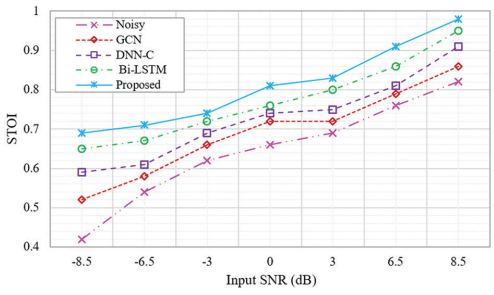
Methods	PESQ	STOI	SSNRI	SDR
Noisy	1.54	0.57	0.00	5.14
GCN [18]	1.68	0.62	7.51	3.84
DNN-C [19]	1.75	0.69	9.52	5.29
Two-stage Bi-LSTM [20]	1.82	0.72	10.45	7.15
Proposed	1.92	0.79	11.5	9.24

Table 5. Comparison of proposed with an existing method for Microphone 2

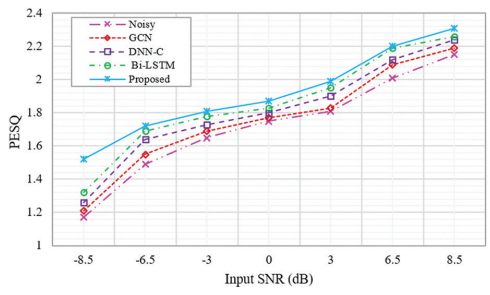
Methods	PESQ	STOI	SSNRI	SDR
Noisy	1.72	0.668	0.00	3.18
GCN [18]	1.54	0.672	5.12	4.15
DNN-C [19]	1.81	0.752	7.15	6.47
Two-stage Bi-LSTM [20]	1.86	0.842	9.157	7.15
Proposed	1.99	0.954	12.45	9.15

Increased speech signal quality and intelligibility are a benefit of GCN [18], but it predicts lower sound quality when used with non-parallel data. On the other hand, DNN-C [19] has the advantage of improving the corresponding noisy input, and all of the channel-wise enhanced outputs are fed into a DNN fusion model to produce a practically clean signal. However, it performs poorly for improving the signal. Additionally, Bi-LSTM [20] can extract local and global characteristics and achieves competitive results; its main drawback is a high computational cost.

The results show that the proposed method performs better than other GCN [18], DNN-C [19], and Bi-LSTM [20] methods when evaluated at various SNRs, proving the model's efficacy. Tables IV and V show the averaged STOI, PESR, SSNRI, and SDR scores for the input signal and enhanced output signal of the two-microphone used in the proposed method. The proposed models outperform the current methods in terms of STOI, PESQ, SSNRI score, and SDR.



(a)



(b)

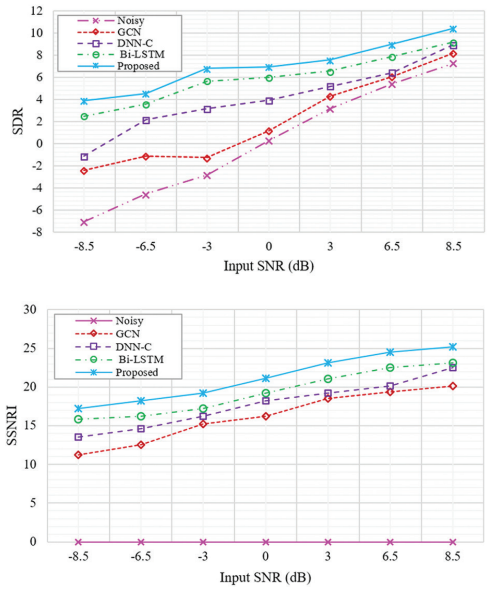


Fig. 6. Performance of (a) STOI, (b) PESQ, (c) SDR, and (d) SSNRI metrics for different SNR levels

In Fig. 6, we analyze the outcomes at various SNR levels. Results for a linear array with two microphones at seven different SNR levels [-8.5, -6.5, -3, 0, 3, 6.5, 8.5] (dB). are displayed. Generally speaking, negative SNR values yield greater performance increases than positive ones. We can see that the proposed Bf-DLN approach improves SDR over the noisy case by more than 8.25 dB for -8.5 dB input SNR. Only 3.48 dB of SDR improvement is seen for the input SNR with the highest values. Additionally, we note that the proposed model outperforms GCN, DNN-C, and Bi-LSTM across the board for SNR values. The proposed model has 0.95, 1.54, and 4.26 improvement on STOI, PESQ, and SDR, respectively. The performance of the SSNIR also improved compared to the existing techniques.

6. CONCLUSION

In this paper, a novel Beamforming based Deep learning Network (Bf-DLN) has been proposed for speech enhancement. Initially, the captured microphone array signals are pre-processed using an Adaptive beamforming technique named Least Constrained Minimum Variance (LCMV). Consequently, the proposed method uses a time-frequency representation to transform the pre-processed data into an image. Time-domain speech signals are converted into pictures using the smoothed pseudo-Wigner-Ville distribution (SPWVD). These converted images are given as input to the convolutional deep belief network (CDBN) for extracting the most relevant features. Enhanced Elephant Heard Algorithm (EEHA) is used for selecting the desired source by eliminating the interference source. The experimental result demonstrates the effectiveness of the proposed strategy in removing background noise from the original speech signal. The proposed strategy outperforms existing methods in terms of PESQ, STOI, SSNRI, and SNR. Results indicate the superiority of our approach when compared to prior state-of-the-art methods.

7. REFERENCES

[1] J. Benesty, "Fundamentals of speech enhancement", Springer, 2018.

[2] B. K. Khonglah A. Dey, S. R. Prasanna, "Speech enhancement using source information for phoneme recognition of speech with background music", Circuits, Systems, and Signal Processing, Vol. 38 No. 2, 2019, pp. 643-663.

[3] Z. X. Li, L. R Dai, Y. Song, L. McLoughlin, "A conditional generative model for speech enhancement", Circuits, Systems, and Signal Processing, Vol. 37, No. 11, 2018, pp. 5005-5022.

[4] P. Malathi, G. R. Suresh M. Moorthi N. R. Shanker, "Speech Enhancement via Smart Larynx of Variable Frequency for Laryngectomee Patient for Tamil Language Syllables Using RADWT Algorithm", Circuits, Systems, and Signal Processing, Vol. 38, No. 9, 2019, pp. 4202-4228.

[5] T. K. Dash, S. S. Solanki, G. Panda, "Improved phase aware speech enhancement using bio-inspired and ANN techniques", Analog Integrated Circuits and Signal Processing, Vol. 102, No. 3, 2020, pp. 465-477.

[6] J. Yang, B. Xia, Y. Shang, W. Huang, C. Mi, "Improved battery parameter estimation method considering operating scenarios for HEV/EV applications", Energies, Vol. 10, No. 1, 2016, p. 5.

[7] S. Gannot, E. Vincent, S. Markovich-Golan, "A consolidated perspective on multimicrophone speech enhancement and source separation", IEEE/ACM Transactions on Audio, Speech, and Language Processing, Vol. 25, No. 4, 2017, pp. 692-730.

[8] M. Tammen, S. Doclo, "Deep multi-frame MVDR filtering for single-microphone speech enhancement", Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, Toronto, ON, Canada, 6-11 June 2021, pp. 8443-8447.

[9] E. Gentet, B. David, S. Denjean, G. Richard, V. Roussarie, "Speech intelligibility enhancement by equalization for in-car applications", Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, Barcelona, Spain, 4-8 May 2020, pp. 6934-6938.

[10] Y. Alkaher, I. Cohen, "Dual-Microphone Speech Reinforcement System with Howling-Control for In-Car Speech Communication", Frontiers in Signal Processing, Vol. 2, 2022, p. 819113.

[11] N. Saleem, J. Gao, M. I. Khattak, H. T. Rauf, S. Kadry, M. Shafi, "Deepresgru: residual gated recurrent neural network-augmented kalman filtering for speech enhancement and recognition", Knowledge-Based Systems, Vol. 238, 2022, p. 107914.

[12] S. Y. Chuang, H. M. Wang, Y. Tsao, "Improved lite audio-visual speech enhancement", IEEE/ACM Transactions on Audio, Speech, and Language Processing, Vol. 30, 2022, pp. 1345-1359.

[13] T. Tao, H. Zheng, J. Yang, Z. Guo, Y. Zhang, J. Ao, Y. Chen, W. Lin, X. Tan, "Sound Localization and Speech Enhancement Algorithm Based on Dual-Microphone", Sensors, Vol. 22, No. 3, 2022, p. 715.

[14] V. Kothapally, Y. Xu, M. Yu, S. X. Zhang, D. Yu., "Deep Neural Mel-Subband Beamformer for in-Car Speech Separation", Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, Rhodes Island, Greece, 4-10 June 2023.

[15] B. Jolad, R. Khanai, "An approach for speech enhancement with dysarthric speech recognition using optimization based machine learning frameworks", International Journal Of Speech Technology, 2023, pp. 1-19.

[16] L. Qian, F. Zheng, X. Guo, Y. Zuo, W. Zhou, "Vehicle Speech Enhancement Algorithm Based on TanhDBN", Proceedings of the IEEE 3rd International Conference of Safe Production and Informatization, Chongqing City, China, 28-30 November 2020, pp. 434-438.

[17] W. Zhou, R. Ji, J. Lai, "MetaRL-SE: a few-shot speech enhancement method based on meta-reinforcement learning", Multimedia Tools and Applications, 2023, pp. 1-20.

[18] S. S. Wang, Y. Y. Liang, J. W. Hung, Y. Tsao, H. M. Wang, S. H. Fang, "Distributed microphone speech enhancement based on deep learning", arXiv:1911.08153, 2019.

[19] P. Tzirakis, A. Kumar, J. Donley, "Multi-channel speech enhancement using graph neural networks", Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, Toronto, ON, Canada, 6-11 June 2021, pp. 3415-3419.

[20] X. Shen, Z. Liang, S. Li, Y. Jiang, "Multichannel Speech Enhancement in Vehicle Environment Based on Interchannel Attention Mechanism", Journal of Advanced Transportation, Vol. 2021, 2021, pp. 1-9.

Amazigh Spoken Digit Recognition using a Deep Learning Approach based on MFCC

Original Scientific Paper

Hossam Boulal
Sidi Mohamed Ben Abdellah University of Fez
Multidisciplinary faculty of Taza, LSI Laboratory
Taza, Morocco.
hossam.boulal@usmba.ac.ma

Mustapha Abarkan
Sidi Mohamed Ben Abdellah University of Fez
Multidisciplinary faculty of Taza, LSI Laboratory
Taza, Morocco.
mustapha.abarkan@usmba.ac.ma

Mohamed Hamidi
Mohamed First University of Oujda,
Multidisciplinary faculty of Nador, Team of modeling and scientifc computing
Nador, Morocco.
m.hamidi@ump.ac.ma

Jamal Barkani
Sidi Mohamed Ben Abdellah University of Fez
Multidisciplinary faculty of Taza, LSI Laboratory
Taza, Morocco.
jamal.barkani@usmba.ac.ma

Abstract *– The field of speech recognition has made human-machine voice interaction more convenient. Recognizing spoken digits is particularly useful for communication that involves numbers, such as providing a registration code, cellphone number, score, or account number. This article discusses our experience with Amazigh's Automatic Speech Recognition (ASR) using a deep learning-based approach. Our method involves using a convolutional neural network (CNN) with Mel-Frequency Cepstral Coefficients (MFCC) to analyze audio samples and generate spectrograms. We gathered a database of numerals from zero to nine spoken by 42 native Amazigh speakers, consisting of men and women between the ages of 20 and 40, to recognize Amazigh numerals. Our experimental results demonstrate that spoken digits in Amazigh can be recognized with an accuracy of 91.75%, 93% precision, and 92% recall. The preliminary outcomes we have achieved show great satisfaction when compared to the size of the training database. This motivates us to further enhance the system's performance in order to attain a higher rate of recognition. Our findings align with those reported in the existing literature.*

Keywords*: CNN, ASR, Deep learning, digit, Amazigh, speech*

1. INTRODUCTION

Amazigh is a North African language that is spoken by 50% of Moroccans. There are 4 vowels, 27 consonants, 2 semi-consonants, and 33 graphemes in the Amazigh language [1], This refers to the Tifinagh letters used in Morocco. Automatic Speech Recognition (ASR) is a challenging task in the Amazigh language due to its morphological complexity, language barrier, dialects, and resource limitations. ASR now has a wide range of applications that are both diverse and challenging, especially when combined with deep learning, a branch of machine learning [2]. It incorporates several techniques and features that mimic human thought and behavior. Deep learning algorithms modify the input values through a hierarchy of nonlinear transformations, and as an output, they create statistical models that are capable of independently anticipating future events [3]. Deep learning techniques require more processing power and a large amount of training data [2]. Artificial neural networks (ANN) [4], convolutional neural networks (CNN) [5], and recurrent neural networks (RNN) [6] are only a few examples of the numerous types of neural networks. The most popular deep neural network for understanding and interpreting visual data is a convolutional neural network [2].

In this work, we present the creation and evaluation of an Amazigh speech recognition system based on the MFCC and CNN approach. Our study focuses on harnessing the potential of these techniques to develop an accurate and efficient system for recognizing Amazigh speech. The focal point of this paper is our CNN model, which is designed to process input images of MFCC features that are extracted from audio signals. Our approach to addressing the complexity of audio classification involves reframing it as an image classification problem, using plots of the MFCC features as inputs to the CNN model. This innovative strategy is a key contribution to our work.

The rest of the paper is organized as follows: Sect. 2 presents the related works. Section 3 presents our methodology and system preparation. Section 4 shows the system's results and performance. Finally, Sect. 5 concludes the paper.

2. LITERATURE REVIEW

Several studies have recently examined Moroccan Amazigh speech recognition systems, with a particular emphasis on the Interactive Voice Response (IVR) system [7]. This system was evaluated by the authors for its ability to identify speaker-independent Amazigh digits in loud environments at different decibel levels. The researchers discovered that digits containing the consonant "S" were the most affected.

An automated voice recognition system for Amazigh has been developed by Telmem and Ghanou [8]. The authors trained and assessed the system for different Gaussian values ranging from 1 to 256 by using HMMs with 3 and 5 states. After training with 128 Gaussian mixture models and 5 HMM state numbers, the system performed best. They achieved a system rate of 90%.

Ghazi and Dawi [9] utilized Amazigh numerical construction principles in developing ASR. Their strategy involved generating associated numbers from an unrelated single number ranging from 1 to 10. This method enabled the expansion of the Amazigh accent in ASR. The researchers concluded that the results were satisfactory, taking into account the size of the learning database.

Satori and El Haoussi [10] have evaluated the independence of the speech recognition system through the use of "Alphadigits", a speech database containing Amazigh numbers and letters pronounced by native speakers. They built the system using CMU Sphinx, which is an HMM-based approach. The best results were achieved when they trained the system with 16 GMMs and 5 HMMs, with an accuracy rate of 92.89%.

Ghazi et al. [11] developed an Amazigh ASR based on the transcription of the Tifinaghe alphabet, which was authorized by the Royal Institute of Culture Amazigh (IRCAM). They compared the efficacy of dynamic programming and HMM approaches. Despite the limited number of speakers and the size of the dataset, the HMM method outperformed dynamic programming. This highlights the effectiveness of probabilistic techniques and stochastic models in speech recognition.

Zealouk O. et al. [12] conducted an assessment of the Sphinx toolset for open-source voice recognition. They specifically compared the use of Pocketsphinx and Sphinx-4 decoders, examining the tradeoff between identification accuracy and computational cost. Notably, the evaluation was carried out in the Amazigh language. Their system was trained to detect the first 10 Amazigh digits using 5 Hidden Markov model states (HMMs) with 16 Gaussian mixture models (GMMs) and the Mel frequency spectral coefficients (MFCCs). According to the gathered testing findings, the Pocketsphinx toolbox recorded the highest recognition rate.

Hamidi M. et al. [13] developed a safe VoIP network-based interactive Amazigh speech system. They examine interactive voice response (IVR) solutions for managing backup and firewall responsibilities in their research. Based on HMM automatic speech recognition, a voice platform was created. To increase security, the biological voiceprint is employed for identification and management tasks. The Amazigh language's 10 starting numerals, 33 alphabets, and five words have all been taught to the voice recognition system. They experimented with the network's remote voice control administration system by altering the IVR and ASR system parameters. According to their study, the system works best for admin tests when trained with three HMMs and eight GMMs. The non-admin recognition rate, however, is less than 5%, demonstrating the system's security.

Lounnas K. et al. [14] reported a series of research using a hybrid methodology to improve spoken digit recognition for under-resourced languages of the Maghreb region. The value of including a dialect identification module in an Automatic Speech Recognition (ASR) system has been shown in earlier research. To enable the ASR system to recognize numbers spoken in many languages, they trained their hybrid system on the Moroccan Arabic Dialect (MAD), Algerian Arabic Dialect (AAD), and Moroccan Amazigh Dialect (MAD) in addition to Modern Standard Arabic. They investigated two deep learning models and five machine learning classifiers, the second of which uses two pre-trained models: residual deep neural networks (RDN), and the first of which is based on convolutional neural networks (CNN) (Resnet50 and Resnet101). The CNN model outperforms the other recommended methods, increasing the spoken digit recognition system's performance by 20% for both Moroccan and Algerian languages, according to the data.

3. METHODOLOGY

There are six essential steps in the process for CNN-based digit identification in the Amazigh language, which is depicted in Fig. 1.

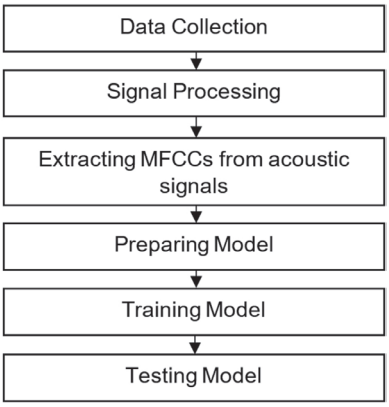


Fig. 1. Steps for proposed model

3.1. DATA COLLECTION

42 native Amazigh speakers were selected for this study to take part in the data collection. All of the volunteers that spoke were between the ages of 20 and 40. The audio database was recorded using the Audacity software. The numbers from 0 to 9 were spoken ten times each by each speaker. Table 1 displays the 0 to 9 digits of the dataset in Amazigh with English transcription. 4200 occurrences (42 speakers * 10 digits * 10 occurrences) were used for the proposed speech recognition model. We have chosen to divide the data into three categories: 60% for training, 20% for validation, and 20% for testing to assure the speaker-independent component.

Table 1. Dataset of digits (0 to 9) in Amazigh and English representation.

Amazigh digits	English transcription	Tifinagh transcription	Number of syllables	Syllables
ILEM	ZERO	ⵉⵍⵎ	2	VCVC
YAN	ONE	ⵢⴰⵏ	1	CVC
SIN	TWO	ⵙⵉⵏ	1	CVC
KṚAḌ	THREE	ⴽⵔⴰⴹ	1	CCVC
KKUẒ	FOUR	ⴽⴽⵓⵥ	1	CCVC
SEMMUS	FIVE	ⵙⵎⵎⵓⵙ	2	CVCCVC
ṢḌIṢ	SIX	ⵚⴹⵉⵚ	1	CCVC
SA	SEVEN	ⵙⴰ	1	CV
TAM	EIGHT	ⵜⴰⵎ	1	CVC
TẒA	NINE	ⵜⵥⴰ	1	CCV

3.2. SIGNAL PROCESSING

Each speaker's recordings were saved as a ".wav" file. To ensure that all of the signal's digits were recorded, the right phrase was kept in the database, and the wrong ones were repeated until reliable and secure recordings were created, each utterance was repeated throughout the recording session.

Speakers were told to record all digits in one voice recording file (with 10 repetitions) to make the work of recording easier. The output files were cleaned up and reduced to a single-word signal (one repetition of each word in each audio file).

3.3. EXTRACTING MFCCS

We approach the challenge of categorizing audio as an image classification task in our method. To be employed with CNN models, audio recording data must be represented in the visual domain, that is the reason we opted to utilize MFCC (Mel Frequency Cepstral Coefficients [15-17].

Mel-Frequency Cepstral Coefficients (MFCC) is a widely used technique in digital signal processing and speech recognition. MFCC allows for the extraction of audio signal features that can be utilized in various signal processing applications and speech recognition tasks. The core concept behind MFCC is based on the observation that the human auditory system is attuned to logarithmically-spaced frequency bands, rather than linearly-spaced ones. To better simulate this experience, MFCC maps the audio spectrum onto the mel-scale, a logarithmic scale that more closely resembles how humans perceive sound. The first step in the MFCC extraction process is to preprocess the audio signal by removing noise and artifacts using methods like windowing and filtering. The signal is then split into brief frames that typically last 20 to 30 milliseconds. For each frame, a power spectrum is computed using a Fast Fourier Transform. A filter bank made up of triangle filters that overlap and are evenly placed on the Mel scale is then used to translate the power spectrum into the Mel scale. The discrete cosine transform (DCT) is then applied to the resulting mel-scaled spectrum to produce the MFCCs, a collection of coefficients that describe the signal's spectral envelope. Typically, just the first 10 to 20 MFCC coefficients are employed for speech recognition tasks since these are the ones that contain the majority of the data necessary to distinguish between various phonemes and words.

In our case, we took 13 Coefficients, plus Delta 1 and Delta 2, a total of 39 in each frame of 31.25 ms. Table 2 displays the parameters that were used to generate the MFCC.

Table 2. Librosa parameter values for MFCC generation

Parameters	Values
Sample rate	16000
Window length	512
hop length	128
Overlap	384
Window function	Hann
FFT Length	512

Below is an illustration of the grayscale MFCC spectrogram, generated using the librosa.feature.mfcc function from the librosa Python library for audio analysis [10].

Fig. 2. MFCC Spectrogram of a Digit Spoken in Amazigh

When we usually see an MFCC spectrogram (Fig. 2), we see it in color, the role of these colors is to facilitate the reading and understanding for the viewer, and it is not necessary, the spectral information can be represented using only black and white. The use of grayscale will be of great benefit at the level of computational cost, as using color, the input matrix dimension will be (256×256×3), but using Grayscale representation, the input dimension will be only (256×256×1).

3.4 PREPARING MODEL

The dataset consists of 4200 instances from both male and female speakers. It consists of 10 recordings of the numbers 0 through 9 being uttered. The dataset was recorded as 16000 Hz monophonic 16-bit audio files in .wav format using the Audacity program. Google Colab is used to construct and execute neural networks and acquire GPU and TPU as a runtime environment. The usage of it is free. The Librosa Python Library is used to extract spectrograms from audio files.

Training, validation, and testing data are split up into three groups with the ratios 60:20:20. The dataset is divided based on speakers, using 25 speakers for training, 8 speakers for validation, and 9 speakers for testing in order to have all the audio samples for the same speaker in only one split (training, or validation, or testing). It was carried out utilizing the train_test_split function of the Python "sklearn" module (see Fig. 3).

We preserved the same number of speakers across all datasets while reducing the number of repetitions for each number; for example, in the case of 100%, there are 10 iterations, 42 speakers, and 4200 total samples. While in the case of 70%, there is the same number of speakers (42), but there are only 7 repeats of each number, therefore there will be 2940 samples in total. Thus, if the percentage is 40%, there will be 1680 samples in total.

The Keras interface and TensorFlow library are used to implement our model in Python.

There are ten classes, numbered from 0 to 9, for each spectrogram picture. The following step is to separate the data into training, validation, and testing sets. Using sequential types, we built a layer-by-layer model in Keras.

The first convolutional layer (C1), which has 96 kernels with symmetrical shapes of sizes (11×11) and stride settings of (2×2) pixels, receives input pictures from the spectrogram during the training phase. A second convolution layer (C2) with 256 kernels of (5×5) is added after a maximum pooling layer (P1) of size (3×3) and a stride of (2×2). The second Maximum Pooling layer (P2) is identical to the first (P1). The last maximum pooling layer (P3) has a size of (2×2) and a final convolutional layer (C3) with 384 kernels of (3×3). Between the convolutional layers and the maximum pooling layers, batch normalization is implemented.

Fully connected layers (FC) of sizes 1024 (FC1), 1024 (FC2), and 10 (FC3) are positioned after the feature learning phase. Except for the last dense layer (FC3), which uses a Softmax activation function, all convolutional layers and dense layers employ the activation function ReLu (Rectified Linear Activation). A dropout layer with a value of 0.2 dropout rate is put after the Fully Connected layers (FC1) and (FC2). Fig. 4 depicts our CNN architecture.

Once all of the model's parameters have been specified, three parameters are utilized during compilation. Loss, metrics, and optimizers are all involved in the compilation process. SGD (Stochastic Gradient Descent) is incorporated from Keras with a momentum of 0.9 since it is an iterative method for maximizing the objective function. Sparse categorical cross-entropy is the name of the loss function that is used to estimate the performance of the model. The third parameter, the accuracy metric, is used to show the model's accuracy on the validation data.

3.5 TRAINING MODEL

A model is trained using training data, validation as test data, and the fit () function. The total number of data model cycles is determined by epochs. Up to a point, better model performance can be achieved by adding more epochs. The recommended model was used to analyze the neural network's performance across 250 epochs.

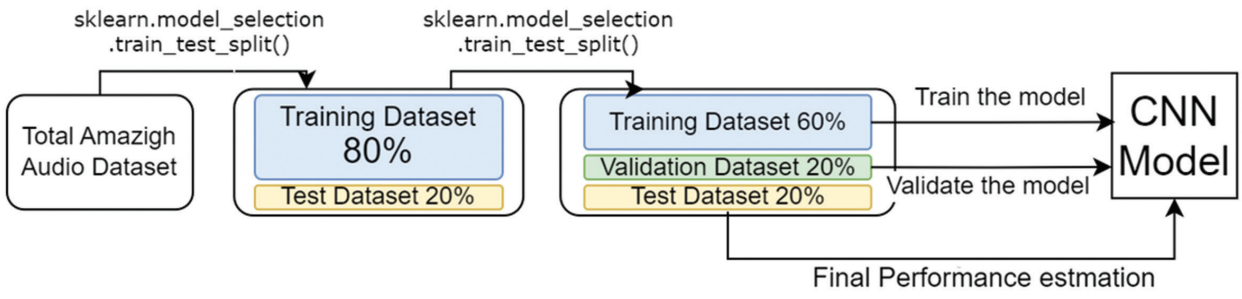


Fig. 3. Training, validation and test division of the dataset

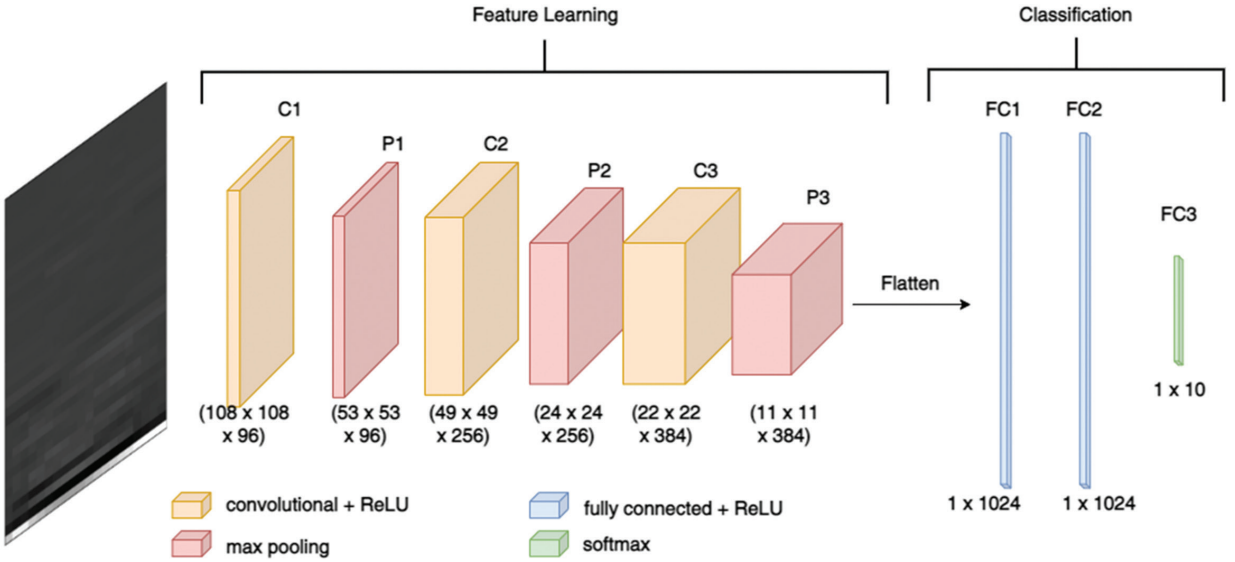


Fig. 4. Convolutional Neural Network

To determine test accuracy, one uses the model evaluation function. Sklearn's classification report and confusion matrix were used. To get information on the accuracy, recall, and f1-score of each test digit, the classification report was invoked.

4. RESULT AND DISCUSSION

The model is run three times with datasets of 1680, 2940, and 4200 digits, respectively. The outcomes of the outcome analysis utilizing several data sets are displayed in Table 3. The 4200 dataset values yielded the greatest results; the accuracy after the first iteration was 88.13%, with 89% precision and 89% recall. With 2940 samples in the second iteration, accuracy was 91.25%, precision was 91%, and recall was 91%. The third iteration produces 91.75% accuracy, 93% precision, and 92% recall with 4200 occurrences.

Table 3. Result analysis with various dataset sizes.

Experiment No.	Data set size	No. of Epochs	Batch size	Accuracy %	Precision %	Recall %
1	1680	250	32	88.13	89	88
2	2940	250	32	91.25	91	91
3	4200	250	32	91.75	93	92

The analysis demonstrates that when dataset sizes grow, recognition accuracy grows as well.

For a Dataset of 1680, 320 samples in testing (32 samples per digit):

Table 4. Classification report 40% dataset.

	precision	recall	f1-score
ILEM	90	88	89
YAN	76	81	79
SIN	94	97	95
KRAD	79	97	87
KKUZ	97	88	92
SEMMUS	97	91	94
ṢDIṢ	100	100	100
SA	88	69	77
TAM	93	84	89
TẒA	74	88	80

In the first experiment, the number "SDIS" had the greatest recall (100%), followed by "SIN", and "KRAD" (97% each of them). In addition, the number "SA" gave the worst recall (69%) (see Table 4).

As for Precision, the number "SDIS" had a perfect classification (100%), "KKUZ" and "SEMMUS" got second place with 97%, while numbers "TZA", "YAN" and "KRAD" got the worst (74%, 76%, and 79%). In this experiment, a significant difference was observed between precision and Recall, concerning the digit "SA" (19%) and digit "KRAD" (18%) (see Fig. 5).

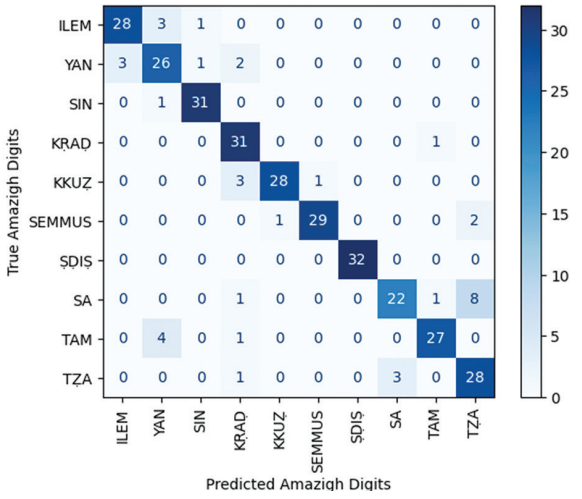


Fig. 5. The Confusion matrix of our CNN model with the 40% of the dataset.

For a Dataset of 2940, 560 samples in testing (56 samples per digit):

Table 5. Classification report 70% dataset.

	precision	recall	f1-score
ILEM	94	80	87
YAN	79	88	83
SIN	88	100	93
KRAD	97	100	98
KKUZ	100	100	100
SEMMUS	100	96	98
ṢDIṢ	97	100	98
SA	84	77	80
TAM	96	88	92
TẒA	81	84	82

In the second experiment, and with the increase of data to 70%, a change occurred in the classification. "SDIS" has no longer the only perfect recall, "KKUZ", "KRAD", and "SIN" had also 100% of recall, the worst recall is still the commentator with numbers "ILEM" and "SA" (80% and 77%). Looking at the precision, the number "KKUZ" and "SEMMUS" have a perfect precision (100%), and the number "YAN" got the worst precision (79%) although his precision improved compared to the first experience. We noticed here that the biggest difference between Precision and Recall was with the number "ILEM" with a difference of (14%) (see Table 5 and Fig. 6).

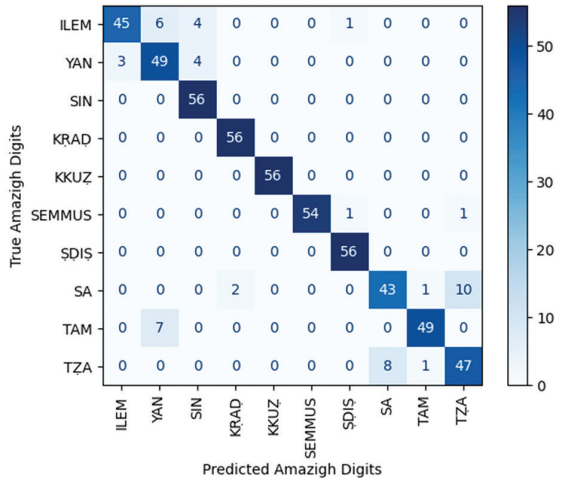


Fig. 6. The Confusion matrix of our CNN model with the 70% of the dataset

For a Dataset of 4200, 800 samples in testing (80 samples per digit):

Table 6. Classification report 100% dataset

	precision	recall	f1-score
ILEM	98	74	84
YAN	72	89	80
SIN	87	99	92
KRAD	99	100	99
KKUZ	98	100	99
SEMMUS	100	97	99
ṢDIṢ	100	100	100
SA	87	84	85
TAM	100	88	93
TẒA	84	88	86

In the third experiment, what happened in the second experiment was confirmed, "SDIS", "KRAD", and "KKUZ" have perfect recall classification (100%), followed by "SIN" (99%), these top four, the same were the first in the previous experiment. The worst recalls are still like the previous experiment with an exchange in order, "SA" with 84%, and "ILEM" with 74%. Also in terms of Precision, the last five orders remained the same as in the previous experiment, while "SDIS", "SEMMUS" and "TAM" got perfect precision (100%) (see Table 6 and Fig. 7).

The numbers "SA" and "TZA" were frequently mentioned when discussing the worst classifications. The confusion matrix in Tables 4, 5, and 6 explains why "SA" and "TZA" received low ratings. We observe that our system often confuses the two digits "SA" and "TZA". We speculate that this may be due to their similar pronunciation.

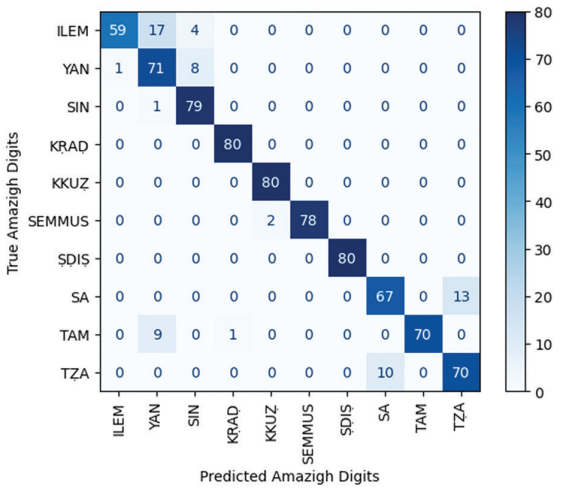


Fig. 7. The Confusion matrix of our CNN model with the 100% of the dataset

Our realized study has been contrasted with other studies that sought to identify speech in the Amazigh language. Table 7 presents a range of results obtained from prior studies. The Amazigh voice recognition system was created by Telmem, M., and Ghanou, Y [18]. It was built on MFCC and CNN, and it was able to recognize 33 letters of the Amazigh language with an accuracy of up to 93.90% from 4620 samples (80:10:10). Satori and El Haoussi (2014) [11] made use of a voice database that included native Amazigh speakers' utterances of Amazigh numerals and letters. This system was built using a system based on HMMs and CMU Sphinx, with feature extraction performed using MFCC. The greatest performance, with a score of 92.89%, came from the recognition results employing 16 Gaussian mixed models and five fixed HMM states. With the use of Pocketsphinx, Zealouk O. et al. [13] developed a system consisting of 5 HMMs and 16 GMMs. They also took advantage of the MFCC's feature extraction capabilities, which resulted in a top classification rate of 92.14% for Amazigh digits. Using a Raspberry Pi, Barkani, F. et al. [19] research Amazigh speech recognition and devel-

op a system based on MFCCs, 3 HMMs, and 16 GMMs. Words in Amazigh were given to the algorithm, and it performed at its best (90.43%).

Table 7. Results of different approaches

Ref	year	Approach	Best parameters	Accuracy
Our Work	2023	MFCC + CNN	----------	91.75%
[13]	2022	MFCC + HMM + GMM	5 HMMs + 16 GMMs	92.14%
[18]	2021	MFCC + CNN	----------	93.90%
[19]	2020	MFCC + HMM + GMM	3 HMMs + 16 GMMs	90.43%
[11]	2014	MFCC + HMM + GMM	5 HMMs + 16 GMMs	92.89%

5. CONCLUSION

In this study, the first 10 Moroccan Amazigh digits were used to assess the ASR speaker-independent system. Our system, which uses MFCC Spectrograms as the basis for feature extraction, was built using the convolutional neural network. As part of this endeavor, the first 10 Amazigh language digits were developed as a speech database called "Amazigh digits," which was used in the system's training and testing stages. The findings we have provided show that our Amazigh ASR system is independent of the speaker and that it agrees with the findings of [13][18][19][11]. Our obtained results show that the best-achieved accuracy is 91.75%.

ACKNOWLEDGEMENT

We want to thank two specialists who have spared no effort in any assistance related to the Amazigh language. We would like to thank Dr. Younas LOUKILI and Mr. Youssef Kouraich, Both teachers of the Amazigh language, and employees of the Ministry of National Education in Morocco.

6. REFERENCES

[1] A. Boukous, "The planning of Standardizing Amazigh language The Moroccan Experience", Iles d imesli, Vol. 6, No. 1, 2014, pp. 7-23.

[2] A. A. Abdullah, M. M. Hassan, Y. T. Mustafa, "A review on bayesian deep learning in healthcare: Applications and challenges", IEEE Access, Vol. 10, 2022, pp. 36538-36562.

[3] J. H. Tailor, R. Rakholia, J. R. Saini, K. Kotecha, "Deep Learning Approach for Spoken Digit Recognition in Gujarati Language", International Journal of Advanced Computer Science and Applications, Vol. 13, No. 4, 2022.

[4] S. Bilgaiyan, S. Mishra, M. Das, "Effort estimation in agile software development using experimental validation of neural network models", International Journal of Information Technology, Vol. 11, No. 3, 2019, pp. 569-573.

[5] V. Jain, A. Jain, A. Chauhan, S. S. Kotla, A. Gautam, "American sign language recognition using support vector machine and convolutional neural network", International Journal of Information Technology, Vol. 13, 2021, pp. 1193-1200.

[6] D. Chhachhiya, A. Sharma, M. Gupta, "Designing optimal architecture of recurrent neural network (LSTM) with particle swarm optimization technique specifically for educational dataset", International Journal of Information Technology, Vol. 11, 2019, pp. 159-163.

[7] M. Hamidi, H. Satori, O. Zealouk, K. Satori, "Amazigh digits through interactive speech recognition system in noisy environment", International Journal of Speech Technology, Vol. 23, No. 1, 2020, pp. 101-109.

[8] M. Telmem, Y. Ghanou, "Estimation of the optimal HMM parameters for amazigh speech recognition system using CMU-Sphinx", Procedia Computer Science, Vol. 127, 2018, pp. 92-101.

[9] A. El Ghazi, C. Daoui, N. Idrissi, "Automatic Speech Recognition for tamazight enchained digits", World Journal Control Science and Engineering, Vol. 2, No. 1, 2014, pp. 1-5.

[10] H. Satori, F. ElHaoussi, "Investigation Amazigh speech recognition using CMU tools", International Journal of Speech Technology, Vol. 17, 2014, pp. 235-243.

[11] A. El Ghazi, C. Daoui, N. Idrissi, M. Fakir, B. Bouikhalene, "Système de reconnaissance automatique de la parole Amazigh à base de la transcription en alphabet Tifinagh", Revue Méditerranéenne des Télécommunications, Vol. 1, No. 2, 2011.

[12] Z. Ouissam, H. Mohamed, S. Hassan, "Investigation on speech recognition Accuracy via Sphinx toolkits", Proceedings of the 2nd International Conference on Innovative Research in Applied Science, Engineering and Technology, Meknes, Morocco, 3-4 March 2022, pp. 1-6.

[13] M. Hamidi, H. Satori, O. Zealouk, K. Satori, N. Laaidi, "Interactive voice response server voice network administration using hidden markov model speech recognition system", Proceedings of the Second World Conference on Smart Trends in Systems, Security and Sustainability, London, UK, 30-31 October 2018, pp. 16-21.

[14] K. Lounnas, M. Abbas, M. Lichouri, M. Hamidi, H. Satori, H. Teffahi, "Enhancement of spoken digits recognition for under-resourced languages: case of Algerian and Moroccan dialects", International Journal of Speech Technology, Vol. 25, No. 2, 2022, pp. 443-455.

[15] Q. Zhou et al. "Cough recognition based on mel-spectrogram and convolutional neural network", Frontiers in Robotics and AI, Vol. 8, 2021, p. 580080.

[16] A. Ahmed, Y. Serrestou, K. Raoof, J.-F. Diouris, "Empirical Mode Decomposition-Based Feature Extraction for Environmental Sound Classification", Sensors, Vol. 22, No. 20, 2022, p. 7717.

[17] K. W. Gunawan, A. A. Hidayat, T. W. Cenggoro, B. Pardamean, "A transfer learning strategy for owl sound classification by using image classification model with audio spectrogram", International Journal on Electrical Engineering and Informatics, Vol. 13, No. 3, 2021, pp. 546-553.

[18] M. Telmem, Y. Ghanou, "The convolutional neural networks for Amazigh speech recognition system", TELKOMNIKA (Telecommunication Computing Electronics and Control), Vol. 19, No. 2, 2021, pp. 515-522.

[19] F. Barkani, H. Satori, M. Hamidi, O. Zealouk, N. Laaidi, "Amazigh speech recognition embedded system", Proceedings of the 1st International Conference on Innovative Research in Applied Science, Engineering and Technology, Meknes, Morocco, 16-19 April 2020, pp.

A Robust Cardiovascular Disease Predictor Based on Genetic Feature Selection and Ensemble Learning Classification

Original Scientific Paper

Sadiyamole P. A.
Research Scholar, Department of Computer Science,
Karpagam Academy of Higher Education
Coimbatore 21, India
sadiya.pa@gmail.com

Dr. S. Manju Priya
Professor, Department of CS, Karpagam Academy of Higher Education
Coimbatore 21, India
smanjupr@gmail.com

Abstract *– Timely detection of heart diseases is crucial for treating cardiac patients prior to the occurrence of any fatality. Automated early detection of these diseases is a necessity in areas where specialized doctors are limited. Deep learning methods provided with a decent set of heart disease data can be used to achieve this. This article proposes a robust heart disease prediction strategy using genetic algorithms and ensemble deep learning techniques. The efficiency of genetic algorithms is utilized to select more significant features from a high-dimensional dataset, combined with deep learning techniques such as Adaptive Neuro-Fuzzy Inference System (ANFIS), Multi-Layer Perceptron (MLP), and Radial Basis Function (RBF), to achieve the goal. The boosting algorithm, Logit Boost, is made use of as a meta-learning classifier for predicting heart disease. The Cleveland heart disease dataset found in the UCI repository yields an overall accuracy of 99.66%, which is higher than many of the most efficient approaches now in existence.*

Keywords*: Cardiovascular disease prediction; Deep learning techniques; Genetic algorithms; Adaptive Neuro-Fuzzy Inference System; Multi-Layer Perceptron; Radial Basis Function; Logit Boost*

1. INTRODUCTION

The World Health Organization (WHO) considers coronary artery problems as the significant reason for people's death worldwide. Their estimation shows about 32% of the world's population is deceased yearly due to CVDs [1] Moreover, according to the Centers for Disease Control and Prevention (CDC) in the United States of America, one civilian dies every second succumbing to CVD [2]. In addition, the country spent nearly $407.3 billion from 2018 to 2019 on heart diseases. The American Heart Association's statistics for heart disease and stroke in 2023 estimate that the yearly direct costs of CVD increased from $103.5 billion in 1996–1997 to $251.4 billion in 2018–2019 [3]. So, CVDs have an impact not only on human fitness but also on the financial side and the expense of countries. A crucial way to decrease this record is to detect CVDs early. But, the identification of CVDs with traditional medical examinations is quite challenging, takes up a great deal of time, and is expensive. Thus, the early determination of heart disease in developing countries is risky due to the shortage of conventional examination tools and medical practitioners [4,5].

The growth in the artificial intelligence field, mainly in the areas of data mining and machine learning procedures, can be well utilized in clinical research. These techniques can be used to effectively diagnose heart diseases in a benignly and cost-effectively manner. In the literature, there are numerous cardiovascular disease diagnostic algorithms proposed by different researchers over the years. To extract needed knowledge in an organized manner from high-dimensional unorganized data, various data mining techniques have been proposed [6, 7]. However, the obtained data may still contain redundant and irrelevant features that can confuse for classification algorithms, resulting in high time complexity and poor performance. Incorporating a good feature selection process is necessary for any heart disease predictive system. Furthermore, among the available classification algorithms, deep learning methodologies are currently considered one of the researcher's favorite disciplines. Therefore, this work proposes a robust CVD prediction

algorithm using the evolutionary power of genetic algorithms along with an ensemble of three established deep learning neural networks such as Multilayer Perceptron (MLP), Radial Basis Function (RBF), and Adaptive Neuro-Fuzzy Inference System (ANFIS). The findings of the experiment demonstrate that, when compared to current approaches, this suggested ensemble learning system has a strong competence in predicting cardiovascular illness.

The rest of this work is put together as follows: An overview of the literature on alternative heart disease prediction techniques is presented in Section 2. A comprehensive explanation of the suggested methodology can be found in Section 3. The experimental findings are discussed in Section 4 along with a comparison to existing methods. The conclusion is presented in Section 5.

2. RELATED WORKS

Over the years of research on the automated diagnosis of cardiovascular diseases, many researchers have suggested different data mining and classification algorithms. Many researchers have tried to incorporate Genetic Algorithms (GA) into heart disease predictions to select the most relevant attributes. The significance of the GA is to categorize the relevant features from the raw data set that could represent the whole more accurately. The fitness function used in this GA identifies the best attributes from the whole dataset.

Gokulnath and Shantharajah [8] proposed GA-based Support Vector Machines (SVM) to diagnose the presence of heart diseases. Their results claimed that the SVM classifier attained only 83.70% accuracy without GA, while the same classifier improved its performance to 88.34% with GA aid. [9] presented a Genetic-based Crow Search Algorithm (GCSA) for selecting relevant features from the dataset and deep convolutional neural networks (DCNN) for the classification. They claimed that the proposed GCSA could increase the classification accuracy by achieving more than 94% in comparison with the other feature selection approaches. Also, Kanwal et al. [10] proposed a heart disease classification algorithm that utilized GA for feature selection. Their classification included different machine learning algorithms like Deep Learning (DL), Naive Bayes (NB), Neural Network (NN), Support Vector Machine (SVM), and Logistic Regression (LR). Their method achieved 92% accuracy.

A similar type of procedure was presented in [11]. They also used GA for feature selection. The performance was obtained using three different classifiers, namely, Nave Bayes (NB), clustering, and Decision Tree (DT). Their experiments show that NB performs consistently with and without the attribute selection procedure, while DT shows better performance after feature selection using GA. They obtained 99.2% accuracy with features selected using GA and DT for classification. Another author, Durga Devi [12], also used genetic algorithms for feature reduction. Furthermore, they used Radial Basis Function (RBF) for diagnosing heart diseases and achieved an accuracy of 85.48%.

Jothi Prakash and Karthikeyan [13] proposed a combination of GA and linear discriminant analysis (LDA) for feature selection and combined the classification results of MLP, NN, DT, and SVM using the ensemble bagging technique. The highest accuracy achieved by their procedure is 93.65% for the Statlog dataset. A hybrid deep learning technique was designed for predicting CVD by Kishore and Jayanthi [14]. To predict CVD, the attribute weights for neural network initialization in Artificial Neural Network (ANN) are performed using Analytic Hierarchy Processing (AHP), and Multilayer Back Propagation Neural Network (MLBPNN) is designed using the Gradient Descent algorithm. Their system obtained an average accuracy of 94.15%. Sharma and Parmar [15] employed an optimized DNN using Talos that produced 90.78% accuracy without any feature extraction procedures.

Mohan et al. [16] presented a novel technique called the Less Error Classifier for the dimensionality reduction of CVD. They produced an accuracy of 88.7% with their HRFLM (Hybrid Random Forest with a Linear Model). Ali et al. [17] proposed a CVD monitoring and prediction system based on ensemble deep learning and feature fusion. The feature combination procedure consisted of missing data filtering, normalization, information gain-based feature selection, and conditional probability-based feature weighting. The system produced an accuracy of 98.5%. Javeed et al. [18] exploited a feature selection method that uses a floating window with adaptive size (FWAFE). Additionally, they used two types of prediction structures: ANN and DNN. They obtained the best results with the DNN system of 93.33%.

Khourdifi and Bahaj [19] utilized an FCBF (Fast Correlation-Based Feature Selection) algorithm to select relevant features from the CVD dataset. The system consisted of classification methodologies based on various classification algorithms such as KNN, SVM, NB, RF, and MLP approaches optimized by Particle Swarm Optimization (PSO) combined with the Ant Colony Optimization (ACO) method. They obtained an accuracy of 99.65% using the FCBF, PSO, and ACO optimized KNN classifiers. Baksh [20] proposed a cluster-based improved deep GA for solving the problem of CVD diagnosis. They developed stochastic gradient boosting with recursive feature elimination (SGB-RFE) for dimensionality reduction, followed by the adaptive Harris Hawk optimization algorithm. Finally, they exploited EDGA for the classification purpose, achieving 99.77% on the UCSF heart disease database. Rani et al. [21] exploited a combination of GA and a recursive feature elimination method for feature selection and a hybrid system consisting of SVM, NB, LR, RF, and Adaboost classifiers for prediction. The system performed best with the RF classifier, giving 86.6% overall accuracy.

Once the literature on CVD is analyzed, it could note that most of the accepted procedures for CVD predic-

tion include a dimensionality reduction technique and a learning mechanism for classification. The dimensionality of the medical dataset is huge so, there is a need for an appropriate parameter selection mechanism to restrict the number of features and to increase classification accurateness. Also, a machine learning procedure is necessary to progress the working performance of the system predictions. In addition, with the modern developments in classification practices such as ensemble learning, there could exist a means of further refinement in terms of accuracy, efficiency, and robustness of the predictions. Evolutionary or metaheuristic attitudes open a wide range of benefits over old-fashioned approaches as they encompass least domain-specific information. For attaining a reduced dimensional robust attribute space, GA is exploited in this work. Furthermore, when compared to a single classification algorithm, ensemble techniques usually produce more accurate results. Consequently, the suggested method employs the ensemble approach to categorization.

3. PROPOSED METHODOLOGY

The sole intention of this research project is to confer a robust prediction method for CVD prediction. So, here the model proposes a system consisting of three major stages; data preprocessing, feature selection, and classification. But the proposed system could be accepted only if it produces acceptable results, so we carried out the experiments on the Cleveland heart- dataset from the UCI repository [22], which is a benchmark in heart disease prediction research. The comprehensive structure of the proposed system is shown in Fig. 1.

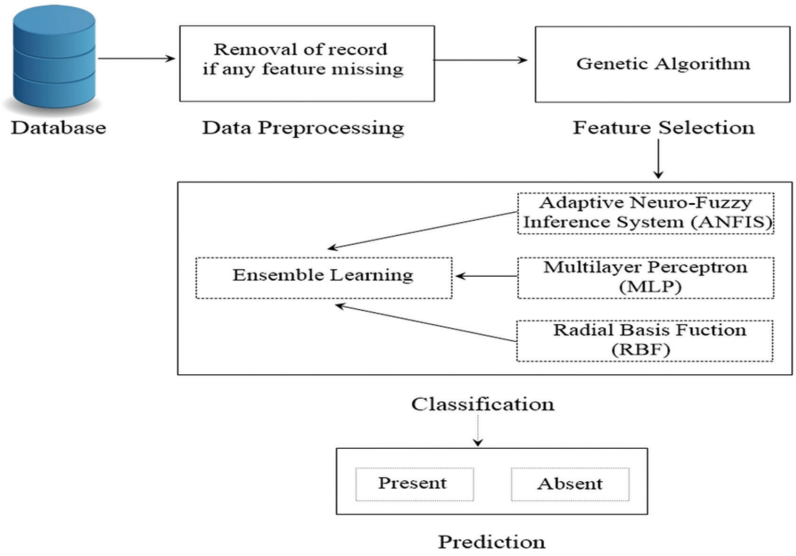


Fig 1. Structure of the Proposed CVD Prediction System

3.1. DATASET AND PRE-PROCESSING

The Cleveland Heart Disease Database formerly consisted of 303 instances with 75 parameters. But a subset of the database with 14 parameters is made available and utilized in most of the research, so this work is also using the same method in this experiment. The age and sex of the data providers along with other disease-related symptoms, such as type of chest pain, blood pressure level, cholesterol level, sugar level, etc., are some of the features included in the database. The 14 included features and their description are given in Table 1.

Once the dataset is analyzed, we notice that out of 303 clinical records, 6 records were missing with some feature values. Those 6 records with missing values were removed in the pre-processing stage as we acknowledged that these records could reduce model performance.

3.2 FEATURE SELECTION

One of the widely used Genetic Algorithms (GA) is employed for eliminating irrelevant and redundant features. Darwin's natural selection is the basic idea behind GA; that is, it works on the principle that the fittest shall survive. It is one of the commonly used feature selection methods [23, 24]. The major characteristics of GA that make it a favorite of researchers; beginning with a set of arguments, GA has the potential to reach a closer optimal solution through a quick convergence, a simple fitness function is only utilized for evaluation, and it supports parallel programming.

3.1. DATASET AND PRE-PROCESSING

The Cleveland Heart Disease Database formerly consisted of 303 instances with 75 parameters. But a subset of the database with 14 parameters is made available and utilized in most of the research, so this work is also using the same method in this experiment. The age and sex of the data providers along with other disease-related symptoms, such as type of chest pain, blood pressure level, cholesterol level, sugar level, etc., are some of the features included in the database. The 14 included features and their description are given in Table 1.

Table 1. Overall structure of the database

Attribute Number	Feature	Explanation
i.	Age	The age completed on the last birthday
ii.	Gender details	Male = 1; Female = 0
iii.	Chest ache is categorized into four types.	Typical angina = 1, atypical angina = 2, non-angina pain = 3, asymptomatic = 4
iv.	Level of BP at resting mode.	Represented anywhere from 94 mm/Hg to 200 mm/Hg
v.	Serum cholesterol level in mg/dl	Represented anywhere from 126 mg/dl to 564 mg/dl
vi.	Sugar levels on fasting >120 mg/dl	True is 1 and false is 0.
vii.	Electrocardiographic results at resting mode.	Hypertrophy equals 2, ST-T wave abnormalities equals 1, and the normal equals 0.
viii.	The highest possible heart rate was noted.	Anywhere from 71 to 202
ix.	Angina brought by exercise	1 represents "yes", and 0 represents "no".
x.	Generated ST depression vs. at rest.	Up-sloping equals 1, Flat equals 2, and down-sloping equals 3.
xi.	Peak exercise slope	SPE ranging from 0 to 6.2
xii.	Major vessel count (0–3) colored by fluorescence	From 0 to 3.
xiii.	Thallium, the status of the heart.	Normal is 3, the fixed defect is 6, and the reversible defect is 7
xiv.	Target value	The presence of heart disease equals 1, and no heart disease 0

Once the dataset is analyzed, we notice that out of 303 clinical records, 6 records were missing with some feature values. Those 6 records with missing values were removed in the pre-processing stage as we acknowledged that these records could reduce model performance.

3. 2. FEATURE SELECTION

One of the widely used Genetic Algorithms (GA) is employed for eliminating irrelevant and redundant features. Darwin's natural selection is the basic idea behind GA; that is, it works on the principle that the fittest shall survive. It is one of the commonly used feature selection methods [23, 24]. The major characteristics of GA that make it a favorite of researchers; beginning with a set of arguments, GA has the potential to reach a closer optimal solution through a quick convergence, a simple fitness function is only utilized for evaluation, and it supports parallel programming.

Any GA encompasses three basic stages: selection, crossover, and mutation.

i. Initially, a set of random populations is created from the data.

ii. Iterated until a generation is formed that satisfies the given fitness threshold value.

a. A fitness function is designed to find the fittest population. We used kNN to estimate the accuracy, then the error was evaluated to find the fitness of each population.

b. In the selection phase, a selected set of features known as chromosomes is placed in a mating pool and generates the next generation. Features in the mating pools are combined to form better generations.

c. The mutation is used to make sure that the next generation is not like the present ones.

3. 3. CLASSIFICATION

In this part, the final strategy of the proposed technique, that is, classification is discussed. The model is aiming to prognosticate the existence and non-existence of heart disease automatically from some of its related parameters. So, here it is dealing with a binary classification algorithm. From the literature analyzed, ensemble models are more reliable and robust in comparison with basic individual methods [25, 26]. Fundamentally, the idea of ensemble learning is that the data is trained using more than a single learning model, and these models are used as the input to the ensemble learning strategy. We have incorporated learning models such as ANFIS, MLP, and RBF to model our ensemble learning procedure.

3.3.1. Adaptive Neuro-Fuzzy Inference System (ANFIS)

Initially, the Adaptive Neuro-Fuzzy Inference System (ANFIS) was included in the scheme. J.S. Roger Jang created the ANFIS in 1993, and is extensively considered as a universal estimator or Takagi-Sugeno fuzzy system [27, 28]. It works with the combined characteristics of ANN and fuzzy inference systems (FIS). Fig. 2 puts forward the basic architecture of the ANFIS classification methodology. From Fig. 3, we could understand that the ANFIS basic architecture consists of 5 layers; Fuzzification, Multiplication, Normalization, De-fuzzification, and Summation.

In the ANFIS model, each rule forms an output based on a linear combination of an input variable and a constant. So, the ultimate output is formed by taking the weighted average of all the outputs from each rule. The following definitions apply to the IF-THEN rules for a Takagi-Sugeno system with two inputs.

$$R1 = a1p + b1q + c1, \text{ if } p = X1 \text{ and } q = Y1 \quad (1)$$

$$R2 = a2p + b2q + c2, \text{ if } p = X2 \text{ and } q = Y2 \quad (2)$$

Here p and q are the inputs in the feature set; Xi and Yi are the linguistic labels; ai, bi, and ci are the consequent parameters; $R1$ and $R2$ are the output fuzzy membership functions.

The fuzzification layer is in charge of mapping the given inputs in the range of 0–1 with the fuzzy sets set up on membership functions (MFs) described using the universal bell function. Every node 'i' in the input

layer is an adaptive node whose node function is represented as follows:

$$O_{1,i} = \mu_{Xi}(p), \quad \text{where } i = 1,2,\ldots \tag{3}$$

$$O_{1,j} = \mu_{Yi}(q), \quad \text{where } j = 1,2,\ldots \tag{4}$$

Where $\mu_{Xi}(p)$, and $\mu_{Yi}(p)$ are the membership functions of the linguistic labels Xi and Yj respectively.

The multiplication layer estimates the number of fuzzy rules to be involved in the system. The output of the multiplication layer defines the dismissal strength of IF-THEN rules which is stated as follows:

$$O_{2,i} = w_i = \mu_{Xi}(p), x\, \mu_{Xi}(q) \tag{5}$$

Where i=1, 2,…, and w_i denotes the firing strength of i-th rule.

The neurons in the normalizing layer are fixed and are normalized using the weights of all the neurons in the layer. The following equation is utilized to calculate the node output is calculated.:

$$O_{3,i} = \overline{w}_i = w_i / (\sum_i w_i), \quad \text{where } i = 1, 2,\ldots \tag{6}$$

In the de-fuzzification layer, every node is an adaptive node that embraces the resulting parameters of the model. The output node can be obtained as follows:

$$O_{4,i} = \overline{w}_i z_i = w_i (a_i p + b_i q + c_i), \quad \text{where } i = 1,2,... \tag{7}$$

The final summation layer appends all the outputs from the previous layer to draw the final output of the ANFIS architecture as follows:

$$O_{5,i} = \sum_i \overline{w}_i z_i = \sum_i w_i z_i / \sum_i w_i, \quad \text{where } i = 1, 2,... \tag{8}$$

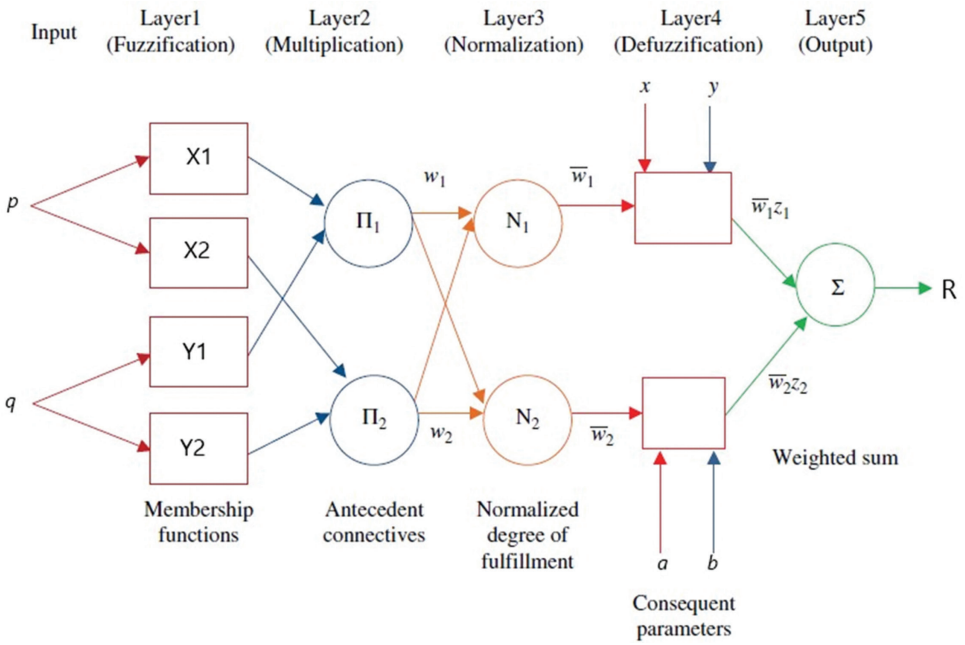


Fig 2. The basic structure of ANFIS

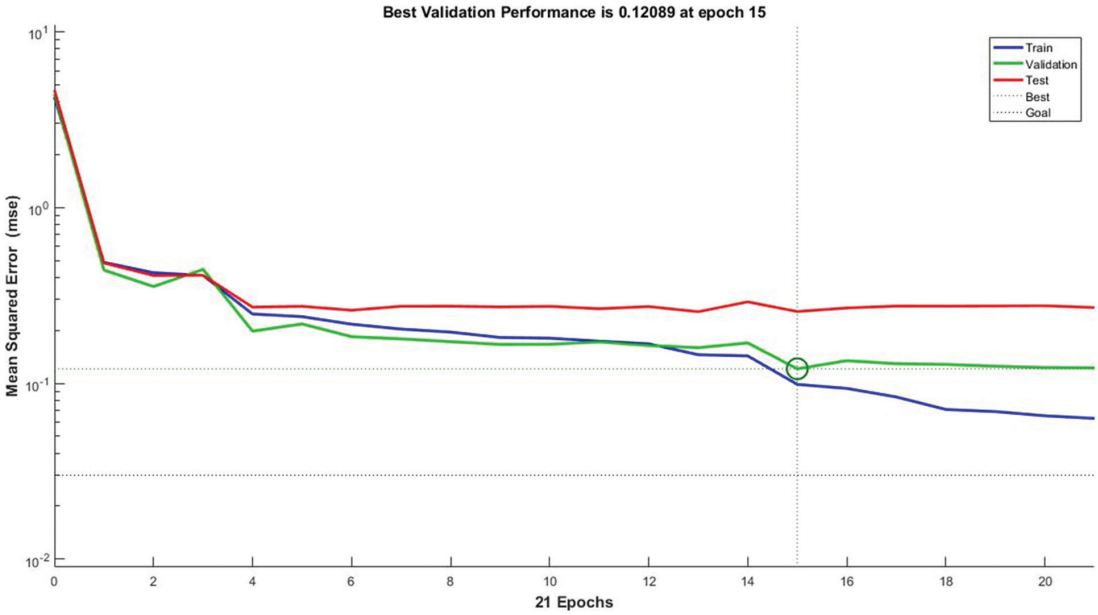


Fig 3. Performance graph of MLP model

3.3.2. Multi-Layer Perceptron (MLP)

The multilayer perceptron is a widely used feed-forward neural network due to its faster execution, easier implementation, and need for comparatively lesser training values [29, 30]. Generally, MLP encompasses three sequential layers, namely, the input layer, the hidden layer, and the output layer. The hidden layer is accountable for processing and executing output from the input layer to the output layer. The number of neurons should be analyzed according to the subject and data distribution to solve the issues of bad generalization and overfitting.

Here hidden layer with 7 neurons has been used for experimental purposes. The input signals xi that each neuron j in the hidden layer receives are multiplied by the corresponding connection weights wij before summing them together as follows:

$$Z_i = f(\sum w_{ji} x_i) \quad (9)$$

where in *f* is a function of activation using the weighted sums of the inputs. If the hyperbolic tangent function is taken as the activation function, which can be estimated as follows

$$f = tansig(x) = 2 / (1+ exp(-2*x)) -1 \quad (10)$$

The performance graph of MLP classification based on Mean Squared Error (MSE) at 21 epochs of the system is presented in Fig. 3.

3.3.3 Radial Basis Function (RBF)

Another widely used ANN method is the Radial Basis Function (RBF), which also consists of three layers similar to the MLP network [31, 32]. Each neuron in the hidden layer contains a radial basis function as the activation function, while each neuron in the input layer contains a certain predictor variable. A weighted sum of the hidden layer's outputs, which is a Gaussian function (Ψ), is included in the output layer and is shown below.

$$\Psi i = exp(-(X-Ci^2)/(2\sigma^2)) \quad (11)$$

here *ci* is the i-th neuron's centre vector and σ is a vector representing basis width. The output of the i-th RBF network can be computed using the following:

$$y_i = \sum_{q=1}^{p} W_{iq} \Psi_q \quad (12)$$

where Wiq is the weight of the *q-th* hidden neuron to the *i-th* output. The performance graph based on the MSE of the train set of our suggested system is presented in Figure 4.

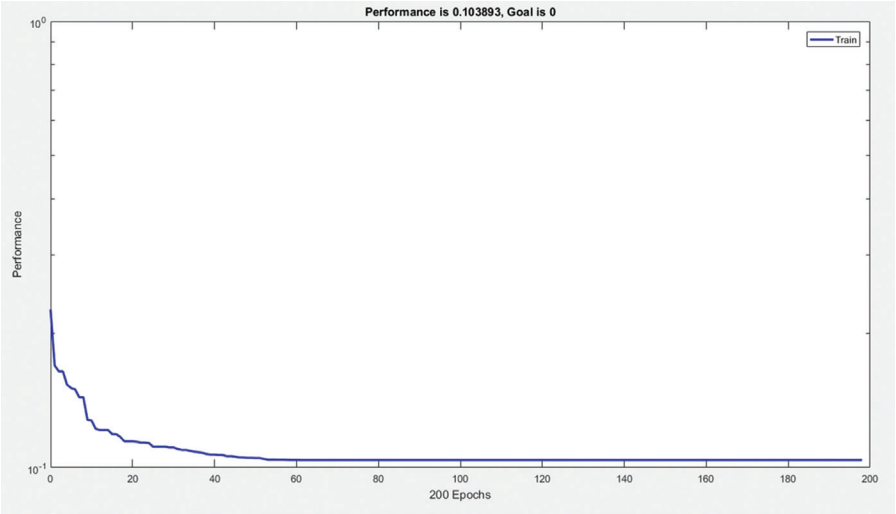


Fig. 4. Performance graph of the RBF model

3.3.4. Ensemble Learning

One of the most advanced strategies for addressing the concerns on classification and prediction problems are ensemble learning methods. It works on the principle that a combination of predictions could lead to better predictions [33]. Ensemble learning techniques provide benefits such as reduced overfitting, reduced finishing with local minimums, reduced impact of the curse of dimensionality, best-fit data, and improved class balances [34, 35].

There are different ways to form an ensemble learning mechanism and we have used the infamous boosting strategy to combine the results of previously mentioned learning methods.Boosting models carry out predictions by placing weak learners in a sequential manner and they reduce overfitting by re-weighting misclassified data of classifiers. We have used a boosting algorithm known as LogitBoost as a meta-learning classifier [13, 36, 37]. LogitBoost has proved better at handling noisy data by minimizing bias and variance. The main concept of the logitboost algorithm is described below:

i. Set the initial weight to $w_i = 1/N$, for $i = 1, 2, \ldots N$, and probability estimate $p(x_i)=1/2$.

ii. Iterate for $m = 1, 2, \ldots M$:

   a. Calculate the responses and weights below:
   $z_i = (yi^* - p(x_i))/(p(x_i)(1-p(x_i))$
   $w_i = p(x_i)(1-p(x_i))$

   b. Compute the function $f_m(x)$ by a weighted least square regression of zi to *xi* using weights *wi*.

c. Update $F(x) \leftarrow F(x)+1/2\, f_m(x)$ and $p(x) \leftarrow (e^{F(x)}) / (e^{F(x)} + e^{-F(x)})$.

iii. Output the classifier $sign[F(x)] = sign[\sum^{M}_{m=1} f_m(x)]$.

The flow of data in our ensemble learner is shown in Fig. 5.

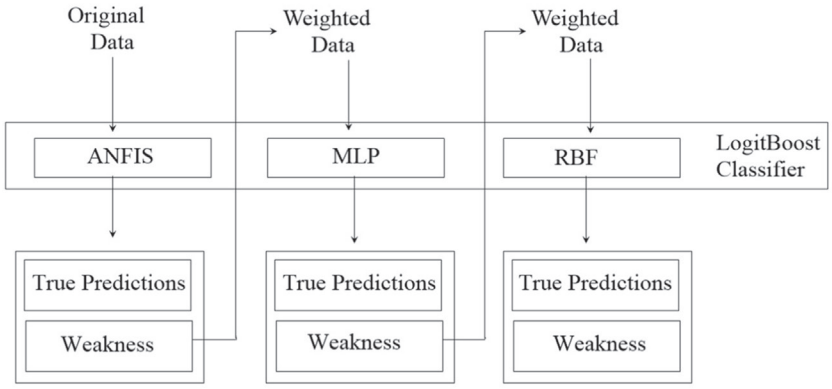


Fig. 5. The overall structure of proposed ensemble learning architecture

4. EXPERIMENTS AND RESULTS

Here, the experiments and tests that were conducted to establish the performance superiority of the suggested CVD diagnostic model will be discussed. This model has used MATLAB 2019 an environment to perform the system experiments with UCI-Cleveland heart disease datasets (discussed in Section 3.1). This technique has employed commonly used performance metrics like accuracy, specificity, sensitivity, precision, and f-measure to scale the results [21, 38]. The analysis of these metrics is performed using a confusion matrix, which produces a total outcome of four measures; true positive *(T_P)*, true negative *(T_N)*, false positive *(F_P)*, and false negative *(F_N)*. We could analyze the needed measures from these parameters with the following equations:

$$Accuracy = (T\_P+T\_N) / (T\_P+T\_N+F\_P+F\_N)$$

$$Specificity = T\_N / (T\_N+F\_P)$$

$$Sensitivity = T\_P / (T\_P+F\_N)$$

$$Precision = T\_P / (T\_P+F\_P)$$

$$F\text{-}measure = (2*T\_P) / (2*T\_P+F\_N+F\_P)$$

The results are validated by K-fold cross-validation with K = 10. Performance was assessed with the k=1 data for training and the balance one for testing, k times, with different sets for training and testing each time. Initially, it was experimented without using any feature selection strategy on the pre-processed data. It was noted that the ensemble model performed pretty well even without any feature reduction techniques. Ensemble learning outperformed the use of singular learning techniques as well. The performance of ANFIS is also acceptable in certain cases. Then it progressed to use GA for feature elimination and found a huge rise in performance metrics, so as inferred from the literature, the use of feature selection is highly useful in the case of CVD diagnosis databases.

Here the number of iterations is used as the factor of generation approval and set the number of iterations is 50. Additionally, the cross-over probability has set at 0.25 and the mutation probability at 0.01. The convergence graph for our fitness value vs.the number of iterations is presented in Fig. 6. We have utilized the K-NN algorithm to estimate the cost of the attributes selected in the genetic algorithm. The final set of features consisted of six attributes, numbered i, ix, x, xi, xiii, and xiv.

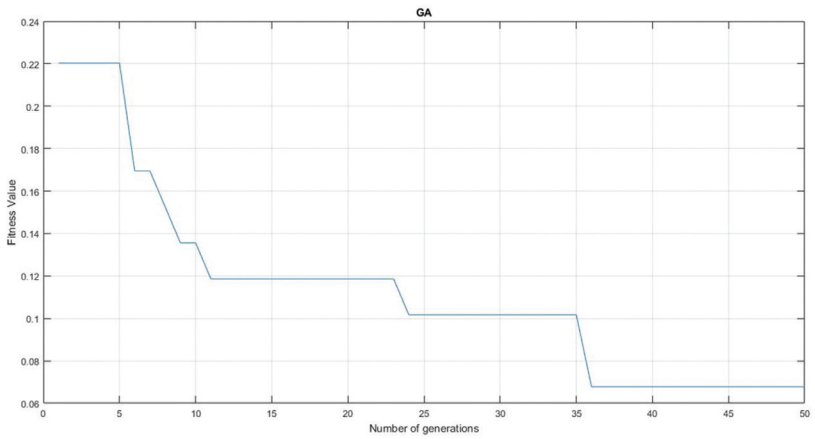


Fig. 6. Graph for Fitness Value VS. Number of Iterations in Genetic Algorithm

The model has obtained a rise of about 10% in most of the techniques with the introduction of GA. The overall accuracy of the ensemble learning was 92.31%, which has shifted to 99.66% with the incorporation of GA. If we take individual techniques of classifications, ANFIS saw a rise from 86.35% to 92.15%, MP hiked from 73.03% to 83.65%, and RBF accuracy changed from 70.79% to 83.17%. The complete experimental results are shown in Table 2. Additionally, for better understanding, we have presented a graphical representation of the comparative analysis of performances based on accuracy in Fig. 7.

Table 2. Performance of the Proposed System with and without Genetic Algorithm for feature selection

Method		Accuracy	Specificity	Sensitivity	Precision	F-Measure
Without GA	ANFIS	86.35	82.53	86.80	88.91	89.63
	MLP	73.03	82.14	68.85	89.36	77.78
	RBF	70.79	7.14	100	70.11	82.43
	Ensemble	92.31	97.73	82.89	95.45	88.73
With GA	ANFIS	92.13	77.50	97.53	94.05	95.76
	MLP	83.65	90.91	71.05	81.82	76.06
	RBF	83.17	92.19	68.75	84.62	75.86
	Ensemble	99.66	100	99.27	100	99.63

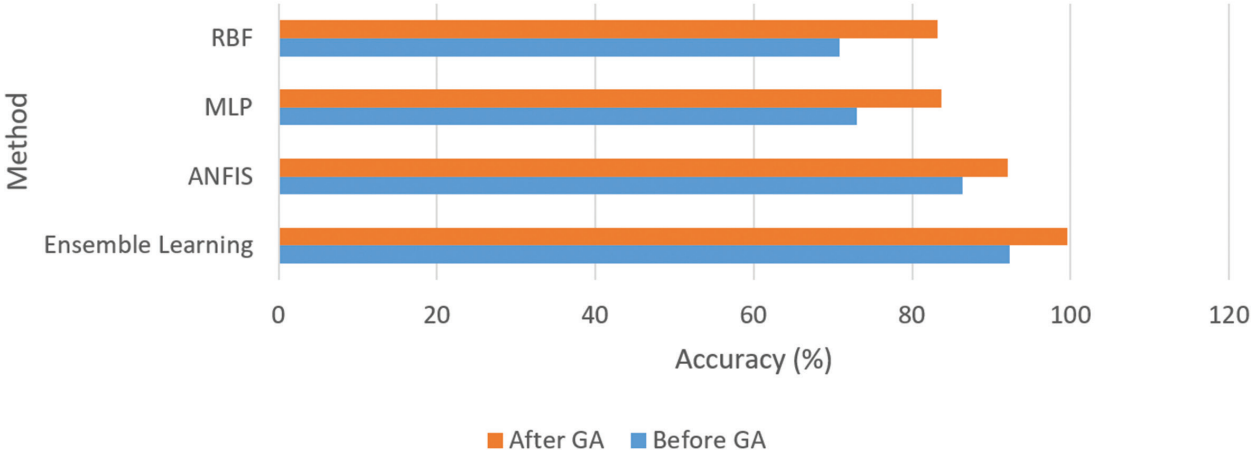


Fig 7. The Comparative Analysis of the Different Algorithms Before and After GA, Based on Accuracy

Furthermore, we have compared the performance of the suggested system against previous literature on the subject of interest. Table 3 presents the comparative analysis of the same. Here, it has included 10 recent works from the past few years in the literature on CVD prediction and diagnosis for comparison purposes. In addition, some comparisons have been made with only those works that provided results on the Cleveland heart disease database to ensure the worthiness of comparison. The results exhibit the superiority of the proposed system over the others based on the accuracy of the system's performance.

Table 3. The Comparison of the proposed system performance against the state-of-art techniques

Author	Feature Selection	Classification	Overall Accuracy
Gokulnath and Shantharajah [8]	GA-SVM	SVM	88.34%
Kanwal et al. [10]	GA	NN	89.00%
Durga Devi [12]	GA	RBF	85.48%
Kishore and Jayanthi [14]	MCDM	ANN	94.15%
Sharma and Parmar [15]	Talos	DNN	90.78%
Mohan et al. [16]	Less Error Classifier	HRFLM	88.70%
Ali et al. [17]	Conditional probabilistic approach	Ensemble deep learning model, and ontology-based recommendation	98.50%
Javeed et al. [18]	FWAFE	DNN	93.33%
Khourdifi and Bahaj [19]	FCBF, PSO and ACO	KNN	99.60%
Rani et al. [21]	GA and Recursive Feature Elimination	RF	86.60%
Proposed Method	GA	Ensemble Learning	99.63%

We could note that both [10] and [11] used a similar feature selection approach as ours, but the results projects differently, this can be due to the difference in the fitness functions used within the GA which has resulted in a different set of features selected in all the cases. Also, the choice of classifiers is another main reason for this disparity. The only system that comes close to the proposed model's performance is [19], but they used a lot of optimization strategies that could add complexities to the system.

5. CONCLUSION

Presently, the cause of the highest mortality rate in the world is cardiovascular disease. Early detection of these diseases could have a high impact on reducing these estimations. Therefore, in this research project, it is aimed to propose a robust automated heart failure prediction system. This method is not trying to replace doctors or heart specialists using the system, but to aid them with a quick diagnosis of the disease. Also, it is being sensed that these systems could be of great use in remote or rural areas where there is a lack of these specialists. In this research work, a GA-based ensemble learning technique that could predict the presence of heart diseases from commonly used clinical data is proposed. An overall accuracy of 99.6% was achieved with the proposed system. The proposed investigational out-turn reveals that the system surpasses some of the existing state-of-the-art works

6. REFERENCES

[1] "Cardiovascular diseases (CVDs)", https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds) (accessed: 2022)

[2] Centers for Disease Control and Prevention, "heart disease facts", https://www.cdc.gov/heartdisease/facts.htm (accessed: 2022)

[3] C. W. Tsao et al. "Heart disease and stroke statistics-2023 update: A report from the American Heart Association", Circulation, Vol. 147, No. 8, pp. e93-e621, 2023.

[4] L. A. Allen, "Decision making in advanced heart failure: a scientific statement from the American Heart Association: A scientific statement from the American heart association", Circulation, Vol. 125, No. 15, 2012, pp. 1928-1952.

[5] H. Yang, J. M. Garibaldi, "Automatic detection of protected health information from clinic narratives", Journal of Biomedical Informatics, Vol. 58, 2015, pp. S30-S38.

[6] J. Jonnagaddala, S.-T. Liaw, P. Ray, M. Kumar, N.-W. Chang, H.-J. Dai, "Coronary artery disease risk assessment from unstructured electronic health records using text mining", Journal of Biomedical Informatics, Vol. 58, 2015, pp. S203-S210.

[7] P. Melin, I. Miramontes, G. Prado-Arechiga, "A hybrid model based on modular neural networks and fuzzy systems for classification of blood pressure and hypertension risk diagnosis", Expert Systems with Applications, Vol. 107, 2018, pp. 146-164.

[8] C. B. Gokulnath, S. P. Shantharajah, "An optimized feature selection based on genetic approach and support vector machine for heart disease", Cluster Computing, Vol. 22, No. S6, 2019, pp. 14777-14787.

[9] S. M. Nagarajan, V. Muthukumaran, R. Murugesan, R. B. Joseph, M. Meram, A. Prathik, "Innovative feature selection and classification model for heart disease prediction", Journal of Reliable Intelligent Environments, Vol. 8, 2022, pp. 333-343.

[10] S. Kanwal, J. Rashid, M. W. Nisar, J. Kim, A. Hussain, "An effective classification algorithm for heart disease prediction with genetic algorithm for feature selection", Proceedings of the Mohammad Ali Jinnah University International Conference on Computing, 2021.

[11] M. Anbarasi, E. Anupriya, N. Iyengar, "Enhanced prediction of heart disease with feature subset selection using genetic algorithm", International Journal of Engineering Science and Technology, Vol. 2, No. 10, 2010, pp. 5370-5376.

[12] A. Durga, D. M. Phil, "Enhanced prediction of heart disease by genetic Algorithm and RBF Network", International Journal of Advanced Information in Engineering Technology, Vol. 2, No. 2, 2015.

[13] V. Jothi Prakash, N. K. Karthikeyan, "Enhanced evolutionary feature selection and ensemble method for cardiovascular disease prediction", Interdisciplinary Sciences: Computational Life Sciences, Vol. 13, No. 3, 2021, pp. 389-412.

[14] A. H. N. Kishore, V. E. Jayanthi, "Neuro-fuzzy based medical decision support system for coronary artery disease diagnosis and risk level prediction", Journal of Computational and Theoretical Nanoscience, Vol. 15, No. 3, 2018, pp. 1027-1037.

[15] S. Sharma, M. Parmar, "Heart diseases prediction using deep learning neural network model", Interna-

tional Journal of Innovative Technology and Exploring Engineering, Vol. 9, No. 3, 2020, pp. 2244-2248.

[16] S. Mohan, C. Thirumalai, G. Srivastava, "Effective heart disease prediction using hybrid machine learning techniques", IEEE Access, Vol. 7, 2019, pp. 81542-81554.

[17] F. Ali et al. "A smart healthcare monitoring system for heart disease prediction based on ensemble deep learning and feature fusion", Information Fusion, Vol. 63, 2020, pp. 208-222.

[18] A. Javeed, S. S. Rizvi, S. Zhou, R. Riaz, S. U. Khan, S. J. Kwon, "Heart risk failure prediction using a novel feature selection method for feature refinement and neural network for classification", Mobile Information Systems, Vol. 2020, 2020, pp. 1-11.

[19] Y. Khourdifi, M. Bahaj, "Heart disease prediction and classification using machine learning algorithms optimized by particle swarm optimization and ant colony optimization", International Journal of Intelligent Systems, Vol. 12, No. 1, 2019, pp. 242-252.

[20] A. A. Bakhsh, "High-performance in classification of heart disease using advanced supercomputing technique with cluster-based enhanced deep genetic algorithm", Journal of Supercomputing, Vol. 77, No. 9, 2021, pp. 10540-10561.

[21] P. Rani, R. Kumar, N. M. O. S. Ahmed, A. Jain, "A decision support system for heart disease prediction based upon machine learning", Journal of Reliable Intelligent Environments, Vol. 7, No. 3, 2021, pp. 263-275.

[22] Heart disease dataset, https://archive.ics.uci.edu/ml/machine-learning-databases/heart-disease/ (accessed: 2022)

[23] S. Sayed, M. Nassef, A. Badr, I. Farag, "A Nested Genetic Algorithm for feature selection in high-dimensional cancer Microarray datasets", Expert Systems with Applications, Vol. 121, 2019, pp. 233-243.

[24] A. K. Das, S. K. Pati, A. Ghosh, "Relevant feature selection and ensemble classifier design using bi-objective genetic algorithm", Knowledge and Information Systems, Vol. 62, No. 2, 2020, pp. 423-455.

[25] M. A. Ganaie, M. Hu, A. K. Malik, M. Tanveer, P. N. Suganthan, "Ensemble deep learning: A review", Engineering Applications of Artificial Intelligence, Vol. 115, No. 105151, 2022, p. 105151.

[26] Y. Cao, T. A. Geddes, J. Y. H. Yang, P. Yang, "Ensemble deep learning in bioinformatics", Nature Machine Intelligence, Vol. 2, No. 9, 2020, pp. 500-508.

[27] J.-S. R. Jang, "ANFIS: adaptive-network-based fuzzy inference system", IEEE Transactions on Systems, Man, and Cybernetics, Vol. 23, No. 3, 1993, pp. 665-685.

[28] S. Chopra, G. Dhiman, A. Sharma, M. Shabaz, P. Shukla, M. Arora, "Taxonomy of adaptive Neuro-Fuzzy Inference System in modern engineering sciences", Computational Intelligence and Neuroscience, Vol. 2021, 2021, p. 6455592.

[29] U. Orhan, M. Hekim, M. Ozer, "EEG signals classification using the K-means clustering and a multilayer perceptron neural network model", Expert Systems with Applications, Vol. 38, No. 10, 2011, pp. 13475-13481.

[30] H. Taud, J. F. Mas, "Multilayer Perceptron (MLP)", Geomatic Approaches for Modeling Land Change Scenarios, Springer International Publishing, 2018, pp. 451-455.

[31] E. Meng Joo, W. Shiqian, L. Juwei, T. Hock Lye, "Face recognition with radial basis function (RBF) neural networks", IEEE Transactions on Neural Networks and Learning Systems, Vol. 13, No. 3, 2002, pp. 697-710.

[32] E. Assareh, M. Biglari, "A novel approach to capture the maximum power from variable speed wind turbines using PI controller, RBF neural network and GSA evolutionary algorithm", Renewable and Sustainable Energy Reviews, Vol. 51, 2015, pp. 1023-1037.

[33] M. Bulmer, "Galton's law of ancestral heredity", Heredity, Vol. 81, 1998, pp. 579-585.

[34] T. G. Dietterich, "An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization", Machine Learning, Vol. 40, 2000, pp. 139-157.

[35] R. Polikar, "Ensemble based systems in decision making", IEEE Circuits and Systems Magazine, Vol. 6, No. 3, 2006, pp. 21-45.

[36] I. Palit, C. K. Reddy, "Scalable and parallel boosting with MapReduce", IEEE Transactions on Knowledge and Data Engineering, Vol. 24, No. 10, pp. 1904-1916, 2012.

[37] M. H. Kamarudin, C. Maple, T. Watson, N. S. Safa, "A LogitBoost-based algorithm for detecting known and unknown web attacks", IEEE Access, Vol. 5, 2017, pp. 26190-26200.

[38] P. Rani, R. Kumar, A. Jain, R. Lamba, "Taxonomy of machine learning algorithms and its applications", Journal of Computational and Theoretical Nanoscience, Vol. 17, No. 6, 2020, pp. 2508-2513.

Sprinkler Irrigation Automation System to Reduce the Frost Impact Using Machine Learning

Original Scientific Paper

Ricardo Yauri
Universidad Tecnológica del Perú,
Faculty of Engineering, Lima, Perú
boncer99@gmail.com
Universidad Nacional Mayor de San Marcos,
Faculty of Systems and Informatic Engineering,
Lima, Perú, ryaurir@unmsm.edu.pe

Oscar Llerena
Seoul National University of Science and Technology
Computer Science and Engineering
Seoul, South Korea
oscar.llerena.c@gmail.com

Jorge Santiago
Universidad Nacional Mayor de San Marcos
Faculty of Electronic and Electrical Engineering,
Lima, Perú
jorge.santiago3@unmsm.edu.pe

Jean Gonzales
Universidad Nacional Mayor de San Marcos
Faculty of Electronic and Electrical Engineering,
Lima, Perú
jean.gonzales2@unmsm.edu.pe

Abstract *– Frosts reduce the ambient temperature to the freezing point of water, affecting the agricultural sector and the integrity of plant tissues, severely damaged by freezing, destroying plant cells. In addition, losses are generated in the economy due to the death of cattle due to cold, hunger, diseases, etc. Latin America is a region that depends, to a considerable extent, on its crops for its consumption and export, so frost represents an urgent problem to solve, considering that in Perú the area of agriculture is not technical. Among the methods most used by farmers is anticipated irrigation, through automatic learning techniques, which allows predicting the behavior of a variable based on previous historical data. In this paper, sprinkler irrigation is implemented in crops exposed to frost, using an automated system with machine learning techniques and prediction models. Therefore, three types of models are evaluated (linear regression, random forests, and decision trees) to predict the occurrence of frosts, reducing damage to plants. The results show that the protection activation indicator from 1.1°C to 1.7°C was updated to decrease the number of false positives. On the three models evaluated, it is determined that the most accurate method is the Random Forest Regression method, which has 80.91% reliability, absolute mean error, and mean square error close to zero.*

Keywords*: Frost, machine learning, sprinkler irrigation, random forests, linear regression, decision trees*

1. INTRODUCTION

Frosts generate a drop in ambient temperature to levels below the freezing point of water, which a problem that affects the agricultural sector. When this phenomenon occurs, the environmental condition becomes critical for the integrity of plant tissues, severely damaged by freezing, generating ice inside the tissues and destroying plant cells [1, 2]. This problem usually occurred all over the world seriously affecting the economy. In our country, this negatively affects agriculture and livestock, causing loss of crop areas, and death of livestock due to cold, hunger, diseases, etc. [3]. Likewise, it affects the health of families due to the low temperature seasons that occur every year [4]. It is for this reason that it is necessary to implement a crop protection system against frost, especially due to climate change [5].

Latin America is a region that depends to a considerable extent on its crops for its consumption and export, so the frost problem represents a problem to be solved by the authorities currently, who have made little effort to mitigate this problem. Likewise, in Peru, agriculture is generally not technified, due to the lack of technical information and training from the state. It should be noted that the regions most affected in the country by frost are above 3000 meters above sea level [5, 6]. Likewise, despite the economic losses suffered by farmers in Peru, no government proposals have yet been put forward to solve this problem.

In the year 2022, the "Multisectoral Plan against Frost and Cold 2022-2024" was approved to reduce the vulnerability of the population to these phenomena recently. This plan seeks to reduce the damage, however, they leave aside the prevention policy to protect

the population and their crops [7]. In the same way, different government and state agencies have identified 1,367 districts exposed to the occurrence of frost out of a total of 1,873 districts that exist in the country. To reduce the effect of frost, different techniques have been used (burning of weeds, use of hormonal products) [8] where early irrigation is the most used and with the best results. However, sprinkler irrigation is more efficient because it ensures frost protection down to -7°C [9]. Another widely used technique is hereditary mechanisms of resistance to water stress due to drought and frost, but, in developing countries, these techniques are complex and little known by farmers [10]. There are also proposals for the development of natural fibers that help mitigate frost which functions as thermal insulation [11].

It is necessary to take advantage of the great advances in the field of automatic learning, to predict the behavior of a variable based on previous historical data [12]. For this there are techniques based on embedded artificial intelligence using machine learning [13, 14]. This contributes to the design of an intelligent system to predict the occurrence of frosts. The model uses historical data of climatological input variables such as temperature, humidity, and pressure, which can be easily obtained from the SENAMHI website.

For all the above, this research proposes the following research question: How is it possible to implement an automated system to support the reduction of the impact of frost on crops?

For this reason, sprinkler irrigation is implemented in crops exposed to frost, using an automated system that will work together with machine learning techniques to predict the occurrence of frost in crops and reduce damage to plants and crops, avoiding great economic losses. As a contribution to the care of crops in agriculture, this system contributes to providing a solution to technical irrigation to improve efficiency in production and water savings [15, 16]. In addition, the application of the predictive system and irrigation by spraying will benefit the agricultural industry in Peru, by mitigating frost damage in crop fields.

This paper has several sections: Section 2 shows the related works. Section 3 describes the concepts and technologies used in automated sprinkler risk systems. Section 4 describes the implementation of the system. Finally, the results are shown in section 5, and the conclusions in section 6.

2. RELATED WORK

A flow of literature work was followed that started with the definition of the research objectives and then an exhaustive search of research papers using keywords related to sprinkler irrigation, frost, and machine learning. Studies were evaluated to identify key findings and gaps in the existing literature. Subsequently, the concepts, theories and results found were synthesized in the literature review, which provided the basis for the choice of methodology. Finally, the selected studies were cited, and included in the bibliography to support the statements and conclusions of this paper.

Frosts are a major problem in the agricultural sector in Peru and the world currently because each year thousands of hectares of crops are affected, which translates into millions of dollars in losses. However, there are different methods to mitigate this phenomenon, highlighting sprinkler irrigation. Vargas [17], shows the control of low temperatures in crops through sprinkler irrigation in the Bolivian altiplaNo. This research demonstrates how water, applied by spraying, allows crops to be protected from frost, taking advantage of the latent heat from freezing water, protecting the plant down to -8 °C. Likewise, the article presents a mathematical model to determine the value of the spray flow.

In the paper developed by Huayta [18] shows the efficiency of sprinkler irrigation is verified to reduce the effects of frost in two varieties of quinoa. The author used the Arduino platform to read the thermometer and turn on the electric pump. Likewise, parameters are considered to verify the protective effect, such as maximum leaf width, maximum leaf length, plant height, main stem diameter, panicle diameter, grain diameter, seed yield per plant, and days of physiological maturity. The results show that when quinoa crops are under the protection of sprinkler irrigation, they grow more than others.

Regarding review papers for frost monitoring, Zho [19], described that frost damage to crops generates economic losses for producers, making it necessary to use a digital infrastructure for the development of agriculture. In some papers, like Hansen et al. [20] and Lu et al.[21], an intermittent control strategy is used using a modified model that included the start and stop time of the system and the adjustment of the water application rate using a simulation with different values of airflow velocity, air temperature , air humidity and spray water temperature, being suitable for protection against frost, and automatically regulating the water application rate.

Some technologies used in the papers focus on the Internet of Things, cloud computing, Machine Learning, and the Soft Computing framework for designing sustainable agricultural systems. These technologies make it possible to obtain information in real time, process and analyze it, incorporate intelligence in decision-making, and use time series data as described in the Cadenas [22]. In addition, in Talsma [23], is used in machine learning techniques and deep neural network models for frost prediction. These technologies can be integrated to diagnose and quantify environmental stress in the winter wheat crop due to climate change through remote and proximal sensing, vegetation index analysis, and the use of statistical and machine learning techniques, to evaluate abiotic and biotic stress in winter wheat. They also allow for obtaining early and accurate information on the environmental impact on crops as described in the paper of Skendžić [24].

Among these solutions are cyber-physical systems (CPS) that consist of sensors, intelligence (prediction), and actuators. In addition, prediction systems are mentioned in the paper of Zhou [25] and Kim [26], which describe how techniques with machine learning and neural networks contribute to the management, control, and prediction of frost. For integrated frost prediction systems, it is necessary to integrate distinct types of sensors.

3. SPRINKLER AND FROST IRRIGATION

3.1. FROST IN AGRICULTURE

In meteorological terms, frosts occur when the air temperature drops to 0ºC or less. However, there is agrometeorological frost, which is the drop in air temperature to critical levels of crops and kills plant tissues, with temperatures above 0ºC [27]. Frost causes damage when ice forms inside the tissues of plants, destroying of their cells. The damage is due to ice crystals that form in the protoplasm of cells, while indirect damage occurs with ice formed in the extracellular space. In both cases, cell damage affects the plants completely or part of it, reducing the quality of the product [1].

According to this paper [28], it is described that there are two types of frosts: radiative and advective. Radiative frosts are related to the leakage of heat accumulated during the day during the night. This happens when the night sky is clear, without wind and the air has a low temperature. This cooling generates thermal inversion due to height, depending on the local topography and weather conditions. Advective frosts occur due to the movement of cold air covering extensive areas of territory, which can last for several hours. These frosts are characterized by having drier and colder air, causing greater damage to plants. Fig. 1 shows the difference between these two types of frost, observing their variation with respect to temperature and altitude for each case.

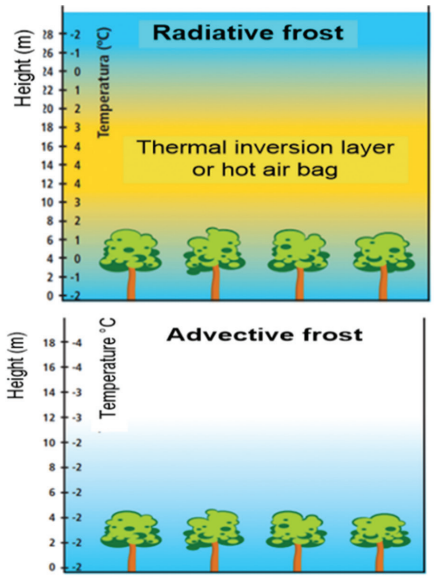


Fig. 1. Radiative and advective frost [28]

3.2. IRRIGATION BY ASPERSION

Sprinkler irrigation is an integral method to deliver pressurized water to the soil uniformly [15]. This method consists of applying intense and uniform rain on the crop, looking for a layer of water to cover each part of the plant. For this purpose, this system is made up of: the pumping group, pipes with hydrants, pipes carrying emitters (irrigation branches), and emitters [29].

The efficiency of the sprinklers is related to their quality, and they are mostly made of plastic. Modern sprinkler irrigation technologies have an innovation-oriented demand and are developing in the direction of high efficiency, energy saving, and enhanced intelligence. New sprinkler irrigation technology is combined with agricultural machinery today [15].

3.3. SENSORS

For the implementation of the irrigation system, it is necessary to add a series of sensor components.

- LM35. Analog sensor that measures temperature, is simple to use, economical, and has a small dimension. It is capable of measuring temperature in a range from -55 ºC to 150 ºC. [30] (Fig. 2).
- DHT11. It is a low-cost digital relative humidity and temperature sensor. It consists of a thermistor and a capacitive humidity sensor to measure the air. This component is used in applications related to monitoring in agriculture, etc. [31].
- BMP180. The BMP 180 barometric pressure sensor combines a wide pressure measurement range, a wide temperature range, high accuracy, and low power consumption [32] (Fig. 3).

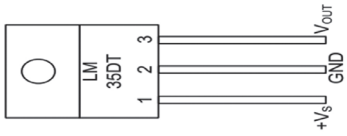


Fig. 2.
LM35 Sensor [30]

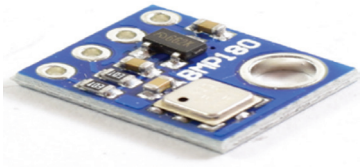


Fig. 3.
Sensor BPM180

4. PROPOSED SYSTEM

The system performs the acquisition of the readings of the meteorological sensors and controls the performance of the sprinklers. The hardware development stage was through a V-methodology while the CRISP methodology was used for the implementation and evaluation of the learning algorithms. Likewise, to have an easy monitoring of the crop and a reliable response of the system, it is proposed to divide the project into two parts: end-devices and base station as shown in Fig. 4:

- End Devices. It is composed of weather sensors connected to the Arduino module with LoraWan data transmission, to send the information to the base station. In addition, it has a protection stage, for its deployment in a crop field. The power part of the system, it has a solar panel and a DC battery for 24-hour operation.
- Base station. It is built based on a Raspberry microcomputer with the Ubuntu operating system. The prediction algorithm will be implemented using three different machine learning techniques. These models will have data received by the base station as inputs and perform the frost prediction, determining the moment in which the protection system will be deployed in the end-devices.

The proposed system is divided into two stages. The first stage is responsible for making the prediction and the second stage does the monitoring and activation. Each of these will be detailed below.

4.1. PREDICTIVE STAGES

In this stage, a series of steps are followed to develop the frost prediction algorithm as shown in Fig. 5, which are: Data cleaning, Training, input of variables, Model construction, and Evaluation. In the Training stage, a previously cleaned data set is used to use regression techniques, decision trees, and random forests. During training, the models learn to recognize patterns and relationships between input variables and frost occurrence using Python's Scikit-Learn tool. In the variable entry stage, the temperature data is provided, and the probability of frost occurrence is estimated. This involves calculating specific thresholds or limits to determine when a frost is considered likely. Finally, in the Evaluation stage, the model's predictions are analyzed using performance metrics, such as the hit rate. This allows you to evaluate the effectiveness of the prediction algorithm and adjust or improvements.

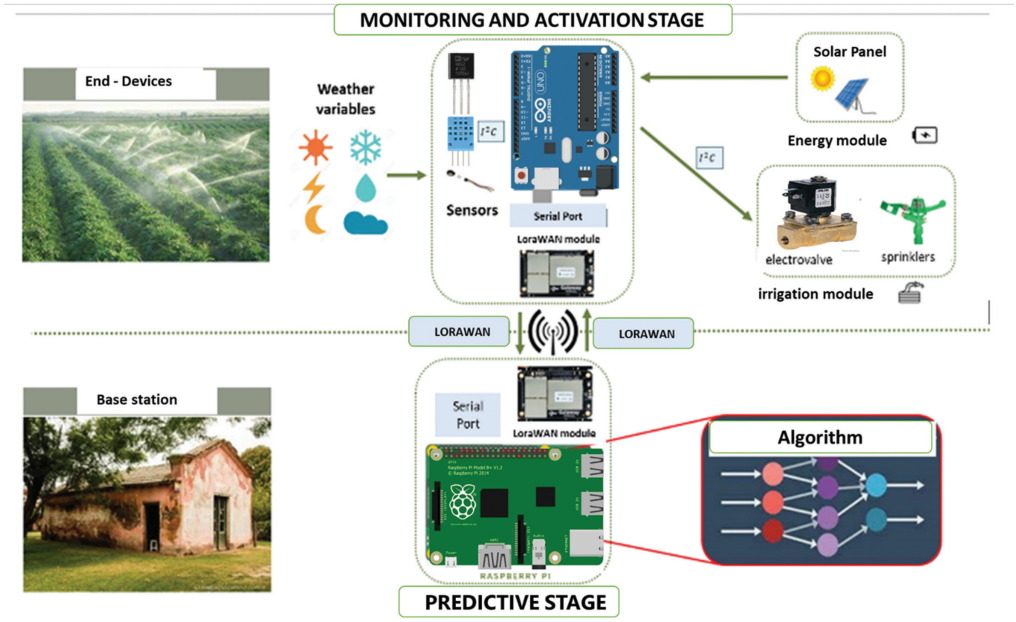


Fig. 4. Frost protection system design

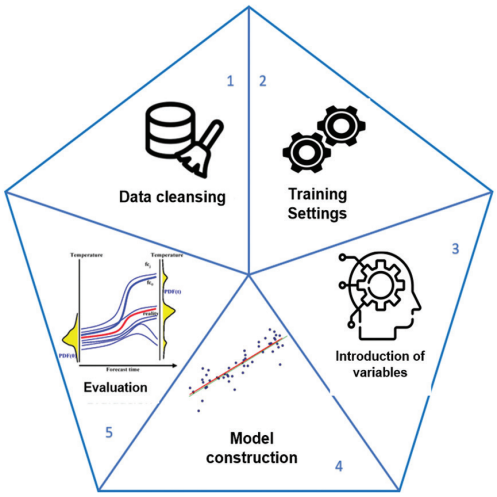


Fig. 5. Predictive Stage Processes

4.2. MONITORING AND ACTIVATION STAGE

This section describes the process of transmission of weather variables to reach the Raspberry module and its input to the prediction models. For this, the following steps are followed:

- The data of the climatological variables are acquired through the sensors, and they are used for the frost and climate forecast.
- The variables are stored in the Arduino hardware for later transmission.
- The data is sent to the Raspberry PI to enter the predictive models.
- Subsequently, a drive stage is implemented, and the response of the frost predictive models is graphically displayed, to activate the actuators

controlled by the Arduino module (Fig. 6). The drive stage performs the following steps:

- Using the algorithm based on the trained predictive models, frosts are predicted, sending data from the Raspberry to the Arduino module using a LoraWAN network.
- The Arduino drives the actuator to deploy the protection system.
- The electrovalve is activated so that it can allow the passage of water towards the sprinklers, starting the irrigation.

Finally, Fig. 7 shows the stages described above integrated into the system. The interaction between the base station and the end-devices is observed through a LoraWAN communication system, and the specific location of the prediction algorithm (base station) and actuators in the Arduino (end-device).

4.3. ALGORITHM DEVELOPMENT

Meteorological data were obtained from the NASA service (Data Access Viewer), which was subjected to a preparation process through filtering. In the first stage, the parameter values of the model are adjusted (number of epochs, number of layers, etc.) for training. Then a validation process is carried out with different data, entering, for example, information for May and June 2022, and frosts are predicted in July. This result can be verified with the real data of the occurrence of frost, obtaining the precision of the prediction. After this, the algorithm is implemented in a Raspberry, which has a low computational load, because it only performs the inference process. Fig. 8 shows the flowchart that indicates the conditions to be met to display the predicted data.

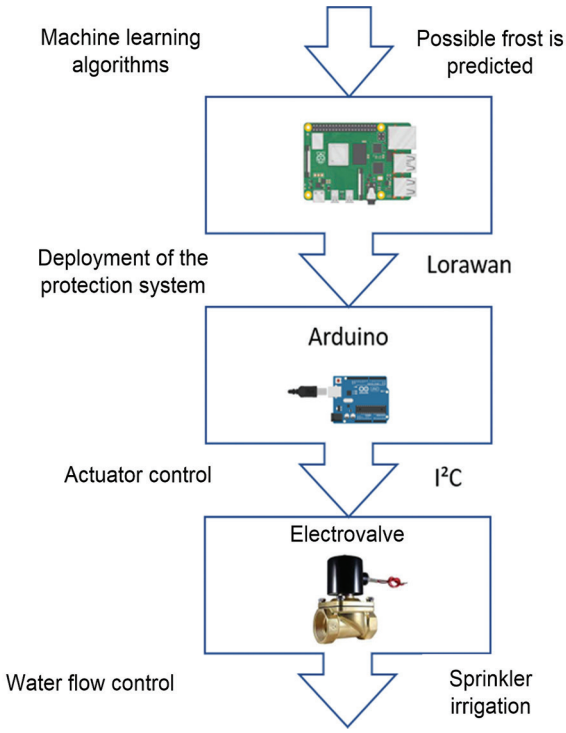


Fig. 6. Drive Stage Steps

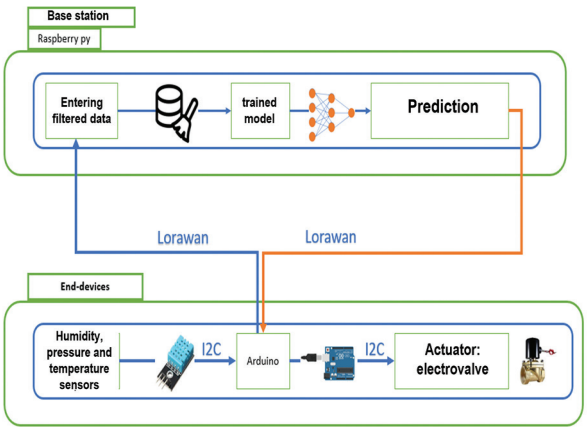


Fig. 7. Integrated system

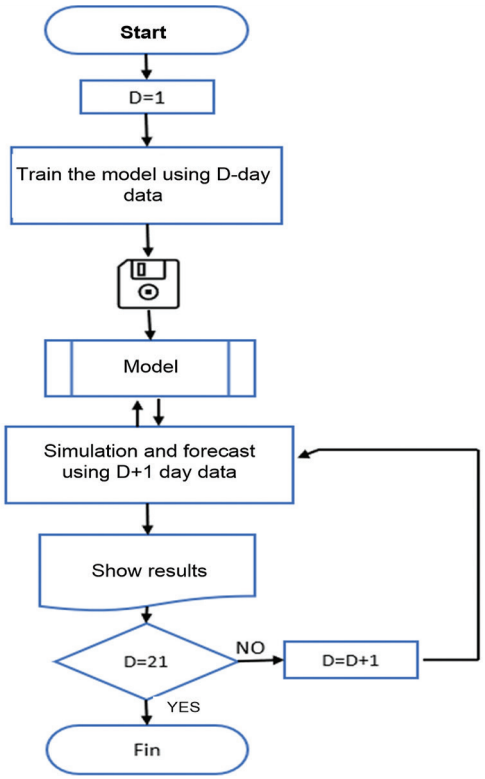


Fig. 8. Inference Process Flowchart

5. RESULTS

The end-devices need more protection because the red lights, the solar panel and the pressure, temperature, and humidity sensors are exposed, as well as the Arduino module and the power stage of the system. In this prototype, a heat-resistant, hermetic-sealable plastic box is used as the casing as shown in Fig. 9.

To obtain the temperature history in the region, the NASA tool that offers historical temperature data for a large part of the planet Earth was used. In this case, a marker was placed in the town of Ataura, where climatological variables were analyzed for their subsequent treatment. Fig. 10 shows the NASA tool used to obtain historical climate data (maximum temperature, minimum temperature, humidity, precipitation, pressure, and wind speed).

With the support of computational tools, such as Python and its sklearn library, the algorithm necessary to predict temperatures was developed. This temperature prediction system is related to frost, establishing that if the temperature drops below 1.1°C, the protection system is activated.

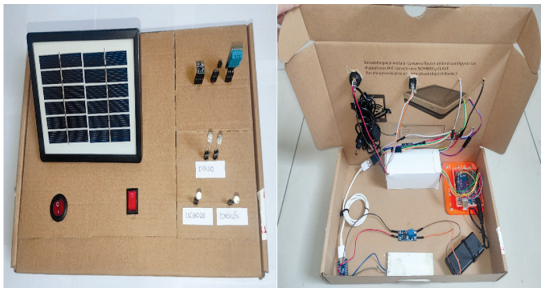


Fig. 9. End Device Prototype

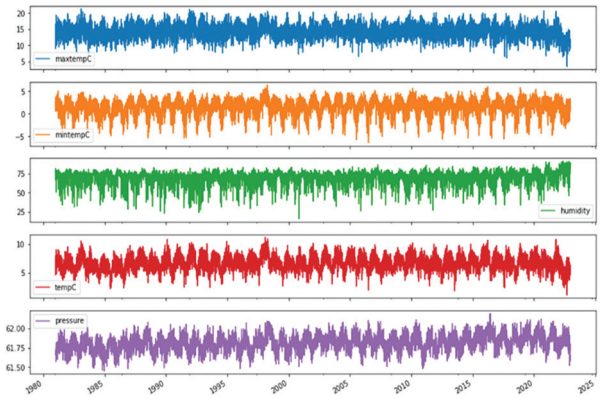


Fig. 10. Meteorological data (1981-2022)

The results compare the behavior of the three types of machine learning algorithms (linear regression, Decision Tree Regression, and Random Forest Regression). The first evaluation is conducted with the linear regression method to predict the temperature values as observed in Table 1, where there are date values, environmental variables, and the activation of the frost control system.

Table 1. Linear Regression Temperature Data

DATE	TEMP. Real (°C)	TEMP. Forecast (°C)	Difference	Activate
2022-06-16	-2.31	-1.88	-0.43	YES
2022-06-24	-2.47	-0.65	-1.82	YES
2022-06-27	-1.59	-2.41	0.82	YES
2022-07-08	-1.81	-0.93	-0.88	YES
2022-07-18	-0.64	0.74	-1.38	YES
2022-07-27	1.1	1.59	-0.49	NO
2022-07-29	-2.92	-1.58	-1.34	YES
2022-08-02	1.26	1.5	-0.24	NO
2022-08-10	-1.01	0.72	-1.73	YES
2022-08-14	-1.35	0.92	-2.27	YES
2022-08-24	0.23	1.22	-0.99	NO
2022-08-25	-1.57	0.22	-1.79	YES
2022-08-28	0.36	0.44	-0.08	YES
2022-09-01	-1.63	0.67	-2.3	YES
2022-09-08	0.54	0.7	-0.16	YES
2022-09-17	-0.27	0.85	-1.12	YES
2022-09-19	1.6	0.3	1.3	YES
2022-09-27	1.58	2.36	-0.78	NO
2022-10-03	0.6	1.63	-1.03	NO
2022-10-06	1.2	2.23	-1.03	NO
2022-10-12	1.62	2.15	-0.53	NO
2022-10-20	-0.6	0.98	-1.58	YES
2022-10-25	1.87	2.64	-0.77	NO

For the evaluation of the models, the average value of the absolute error (MAE) is used, which is a measure of the average discrepancy between the predicted values and the real values in a data set. This evaluation metric is efficient in regression problems through the following equation, where the calculation for "n" real and predicted values is observed:

$$MAE = \sum_{i=0}^{n} \frac{|real\ value_i - predict\ value_i|}{n} \quad (1)$$

While the MAE calculates the mean of the absolute differences, the ECM Mean Squared Error calculates the mean of the squared differences and penalizes the largest errors more, as observed in equation 2:

$$ECM = \sum_{i=0}^{n} \frac{(real\ value_i - predict\ value_i)^2}{n} \quad (2)$$

The absolute error value and square error are calculated for three types of algorithms. Subsequently, these values are averaged, obtaining the results in Table 2. Of the three methods used, the Decision Tree Regression method is the most accurate since its mean absolute error and its mean square error are closer to zero.

Table 2. Mean absolute error and mean square error

	Lineal regression	Decision Tree Regression	Random Forest Regression
Mean absolute error	1.08	1.12	0.87
Root mean square error	4.55	2.08	1.09

Furthermore, we can use the mathematical indicators to obtain the experimental results based on the parameters of the confusion matrix using false positives (FP) where the model incorrectly classifies an instance as positive when it is negative. False negatives (FN) correspond to an erroneous classification of an instance as negative when it is positive. From this we can calculate the Total False Positives (FPR) using the Total Negatives (TN) parameter with the following relationship: FPR = FP/(FP+TN). And on the other hand, we can have the false negative rate (FNR) using the true positives (TP) to obtain the relationship: FNR = FN/(TP+FN).

The number of false positives and false negatives activates the frost protection system. (The critical temperature for ignition is 1.1°C). As can be seen in Table 3, there are four protection failures in linear regression, 7 in the Decision Tree Regression method, and 4 in the Random Forest Regression method.

False positives are less harmful than false negatives since in the case of false negatives they do not activate the protection against frost, therefore the crop would be damaged. For this reason, the value of the protection activation indicator was increased to 1.7°C, reducing the prediction errors (Table 4).

Table 3. False positives and false negatives with 1.1°C indicator

	false positive	false negative
Regression lineal	1	3
Decision tree regression	2	5
Random forest regression	2	2

Table 4. False positives and false negatives with 1.7°C indicator

	false positive	false negative
Regression lineal	1	0
Decision tree regression	2	2
Random forest regression	1	0

Fig. 11 shows the difference between the actual value and the predicted value with the Random Forest Regression method. Finally, to monitor the operation of the system, the efficiency of the prediction algorithm, and the predictions of frost occurrence, an Android application was developed. This application allows you to display updated data in real time (Fig. 12).

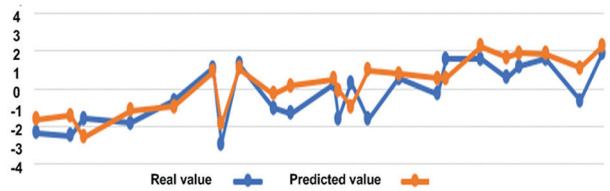


Fig. 11. Comparison between actual and predicted value.

Fig. 12. WARU application (frost monitoring)

6. CONCLUSIONS

The automation of the deployment of the frost protection system was successfully implemented, starting with the reading of the sensors in end-devices and then analyzing the data in the base station. This station issues an order to activate the solenoid valve to activate sprinkler irrigation. In addition, the deployment of the protection system is conducted by turning on a led, to indicate the activation of the opening of the solenoid valve that allows the passage of water towards the sprinklers.

The results of this research provide a specific approach for the deployment of the frost protection system, describing the sensor reading process until the activation of the solenoid valve, while researchers such as Skendžić [24] only broadly describe the impact of climate change on crops and offers a more general vision of monitoring techniques in agriculture. In addition, this paper focuses on practical and technical aspects, unlike works such as Talsma [23], which only show prediction models and algorithms without practical integration in the field or an overview of the benefits of sprinkler irrigation in the case of Yan [15].

The frost detection algorithm was successfully developed, entering historical meteorological data which influences the accuracy of the different models, for which a greater amount of meteorological data is needed to improve the prediction system. The results of the prediction algorithm have an acceptable error range, where the protection activation indicator is increased from 1.1°C to 1.7°C to reduce it. These changes turn out to be positive, as they decrease false negatives from 3 to 0 in the linear regression method, from 5 to 2 in the Decision Tree Regression method, and from 3 to 0 in the Random Forest Regression method.

On the three models evaluated, the Random Forest Regression method has better reliability with 80.91% accuracy and mean absolute error and mean square error close to zero. In addition, it has zero incidences of false negatives, avoiding damaging crops in the event of a frost.

7. REFERENCES:

[1] L. Martínez B., A. Ibacache G., and L. Rojas P., "Efectos de las heladas en la agricultura", https://biblioteca.inia.cl/handle/20.500.14001/7167 (accessed: 2023)

[2] G. Patricia, Z. Sanchéz, E. M. Ceballos, A. L. Ruiz, "Estrategias sociales y ecológicas de resiliencia al cambio climático implementadas por los agricultores del municipio de Marinilla (Colombia)", Agroecology, Vol. 8, No. 1, 2013.

[3] F. H. Solís, W. Iván, V. Terán, "Efecto de las heladas en la región de Cajamarca", Servicio Nacional de

Meteorología e Hidrología del Perú, http://repositorio.senamhi.gob.pe/handle/20.500.12542/118 (accessed: 2023)

[4] M. Muguerza, N. Arce, M. Muguerza, N. Arce, "Heladas y friajes: un acercamiento a la calidad informativa y encuadres en la cobertura periodística de los cibermedios en el Perú", Desde el Sur, Vol. 14, No. 1, 2022, p. e0005.

[5] J. A. Ruiz-Corral et al. "Impacto del cambio climático sobre la estación de crecimiento en el estado de Jalisco, México", Rev. Mex. ciencias agrícolas, Vol. 7, No. SPE13, 2016, pp. 2627-2638.

[6] G. Corina, Q. Mosquera, "Análisis del riesgo por heladas en zonas alpaqueras al sur del Perú", http://diposit.ub.edu/dspace/handle/2445/186279 (accessed: 2023)

[7] Presiderncia del consejo de Ministros, "Plan Multisectorial ante Heladas y Friaje", Lima, Peru, 2022.

[8] H. Mauricio, R. Angula, J. Antonio, N. Rodríguez, "Acción de las poliaminas en la protección de papa criolla (solanum phureja ev. 'yema de huevo') contra las heladas", https://repositorio.unal.edu.co/handle/unal/34238 (accessed: 2023)

[9] C. C. Castellanos Mestizo, "Sistema para la protección de cultivos soterrados y terrestres contra las heladas en la Sabana de Bogotá para el año 2050", Universidad Antonio Nariño, 2021.

[10] J. G. Ortega, M. C. Jury, A. A. Fernández, V. T. Silene, "Selección de cultivares de papa (Solanum tuberosum L.) resistentes a sequía y heladas en Bolivia", Revista Latinoamericana de la Papa, Vol. 24, No. 2, 2020, pp. 17-34.

[11] O. R. Peña Ramírez, R. E. Roman Enciso, "Diseño de un aislante térmico a base de fibras naturales para mitigar el impacto de las heladas en la comunidad de Cupisa", Universidad Peruana de Ciencias Aplicadas, 2018.

[12] W. Muñoz-Herrera, O. Fernando-Bedoya, M. Edilberto-Rincón, W. Muñoz-Herrera, O. Fernando-Bedoya, M. Edilberto-Rincón, "Aplicación de redes neuronales para la reconstrucción de series de tiempo de precipitación y temperatura utilizando información satelital", Escuela de ingenieria de Antioquia, Vol. 17, No. 34, 2020, pp. 73-88.

[13] R. Yauri, R. Acosta, M. Jurado, M. Rios, "Evaluation of Principal Component Analysis Algorithm for Locomotion Activities Detection in a Tiny Machine Learning Device", Proceedings of the IEEE Engineering International Research Conference, Lima, Peru, 27-29 October 2021.

[14] R. Yauri, R. Espino, "Edge device for movement pattern classification using neural network algorithms", Indonesian Journal of Electrical Engineering and Computer Science, Vol. 30, No. 1, 2023, p. 229.

[15] H. Yan, X. Hui, M. Li, Y. Xu, "Development in sprinkler irrigation technology in China*", Irrigation and Drainage, Vol. 69, No. S2, 2020, pp. 75-87.

[16] C. P. Molina Ticona, "Optimización del secado de la hoja de quinua (Chenopodium quinoa willd.) por ultrasonido-infrarrojo-vacío, y evaluación de capacidad antioxidante, fenoles totales y flavonoides totales", Universidad Nacional de Juliaca, 2022.

[17] M. R. Vargas, R. Chipana R., "Control de bajas temperaturas mediante el riego por aspersión fijo en el Altiplano Central de Bolivia", Revista de Investigación e Innovación Agropecuaria y de Recursos Naturales, Vol. 2, No. 1, 2015, pp. 55-67.

[18] R. G. Pineda Huayta, "Evaluación de helada, en quinua (Chenopodium quinoa Willd) en dos variedades con riego por aspersión, en la comunidad de Huarza del distrito de Pucara - Lampa Puno", Universidad Nacional Del Altiplano, 2017.

[19] I. Zhou, J. Lipman, M. Abolhasan, N. Shariati, D. W. Lamb, "Frost Monitoring Cyber-Physical System: A Survey on Prediction and Active Protection Methods", IEEE Internet of Things Journal, Vol. 7, No. 7, 2020, pp. 6514-6527.

[20] B. D. Hansen et al. "Current status of and future opportunities for digital agriculture in Australia", Crop and Pasture Science, 2022.

[21] Y. Lu, Y. Hu, C. Zhao, R. L. Snyder, "Modification of water application rates and intermittent control for sprinkler frost protection", Transactions of the ASABE, Vol. 61, No. 4, 2018, pp. 1277-1285.

[22] J. M. Cadenas, M. C. Garrido, R. Martínez-España, "A Methodology Based on Machine Learning and Soft Computing to Design More Sustainable Agriculture Systems", Sensors, Vol. 23, No. 6, 2023, p. 3038.

[23] C. J. Talsma, K. C. Solander, M. K. Mudunuru, B. Crawford, M. R. Powell, "Frost prediction using machine learning and deep neural network models", Frontiers in Artificial Intelligence, Vol. 5, 2023.

[24] S. Skendžić, M. Zovko, V. Lešić, I. Pajač Živković, D. Lemić, "Detection and Evaluation of Environmental Stress in Winter Wheat Using Remote and Proximal Sensing Methods and Vegetation Indices—A Review", Diversity, Vol. 15, No. 4, 2023.

[25] I. Zhou, J. Lipman, M. Abolhasan, N. Shariati, "Minute-wise frost prediction: An approach of recurrent neural networks", Array, Vol. 14, 2022.

[26] H. Kim, J. M. Kim, S. Kim, "Frost Forecasting considering Geographical Characteristics", Advances in Meteorology, Vol. 2022, 2022.

[27] L. Alfaro Lozano, D. E. Marin Sanchez, P. Porras, "Índice para la estimación de la rigurosidad de la temporada de bajas temperaturas en la región andina del Perú. Nota Técnica 005 SENAMHI-DGM-2015", Servicio Nacional de Meteorología e Hidrología del Perú, Lima, Peru, 2015.

[28] Fundación para la Innovación Agraria, "Heladas : tipos, medidas de prevención y manejos posteriores al daño. Guía y uso del sitio", Santiago, Spain, 2016.

[29] J. M. Tarjuelo Martin-Benito, "El riego por aspersión y su tecnología", Mundi-Prensa, 1999.

[30] P. Dani, P. Adi, A. Kitagawa, "Performance Evaluation WPAN of RN-42 Bluetooth based (802.15.1) for Sending the Multi-Sensor LM35 Data Temperature and RaspBerry pi 3 Model B for the Database and Internet Gateway", International Journal of Advanced Computer Science and Applications, Vol. 9, No. 12, 2018.

[31] W. Gay, "Advanced Raspberry Pi", Apress, 2018.

[32] LabFerrer, "Sensor de presión atmosférica", https://www.lab-ferrer.com/sensor-de-presion-atmosferica-sb-100/ (accessed: 2023)

INTERNATIONAL JOURNAL OF ELECTRICAL AND COMPUTER ENGINEERING SYSTEMS

Published by Faculty of Electrical Engineering, Computer Science and Information Technology Osijek,
Josip Juraj Strossmayer University of Osijek, Croatia.

About this Journal

The International Journal of Electrical and Computer Engineering Systems publishes original research in the form of full papers, case studies, reviews and surveys. It covers theory and application of electrical and computer engineering, synergy of computer systems and computational methods with electrical and electronic systems, as well as interdisciplinary research.

Topics of interest include, but are not limited to:

- Power systems
- Renewable electricity production
- Power electronics
- Electrical drives
- Industrial electronics
- Communication systems
- Advanced modulation techniques
- RFID devices and systems
- Signal and data processing
- Image processing
- Multimedia systems
- Microelectronics
- Instrumentation and measurement
- Control systems
- Robotics
- Modeling and simulation
- Modern computer architectures
- Computer networks
- Embedded systems
- High-performance computing
- Parallel and distributed computer systems
- Human-computer systems
- Intelligent systems
- Multi-agent and holonic systems
- Real-time systems
- Software engineering
- Internet and web applications and systems
- Applications of computer systems in engineering and related disciplines
- Mathematical models of engineering systems
- Engineering management
- Engineering education

Paper Submission

Authors are invited to submit original, unpublished research papers that are not being considered by another journal or any other publisher. Manuscripts must be submitted in doc, docx, rtf or pdf format, and limited to 30 one-column double-spaced pages. All figures and tables must be cited and placed in the body of the paper. Provide contact information of all authors and designate the corresponding author who should submit the manuscript to https://ijeces.ferit.hr. The corresponding author is responsible for ensuring that the article's publication has been approved by all coauthors and by the institutions of the authors if required. All enquiries concerning the publication of accepted papers should be sent to ijeces@ferit.hr.

The following information should be included in the submission:

- paper title;
- full name of each author;
- full institutional mailing addresses;
- e-mail addresses of each author;
- abstract (should be self-contained and not exceed 150 words). Introduction should have no subheadings;
- manuscript should contain one to five alphabetically ordered keywords;
- all abbreviations used in the manuscript should be explained by first appearance;
- all acknowledgments should be included at the end of the paper:
- authors are responsible for ensuring that the information in each reference is complete and accurate. All references must be numbered consecutively and citations of references in text should be identified using numbers in square brackets. All references should be cited within the text;
- each figure should be integrated in the text and cited in a consecutive order. Upon acceptance of the paper, each figure should be of high quality in one of the following formats: EPS, WMF, BMP and TIFF;
- corrected proofs must be returned to the publisher within 7 days of receipt.

Peer Review

All manuscripts are subject to peer review and must meet academic standards. Submissions will be first considered by an editor-in-chief and if not rejected right away, then they will be reviewed by anonymous reviewers. The submitting author will be asked to provide the names of 5 proposed reviewers including their e-mail addresses. The proposed reviewers should be in the research field of the manuscript. They should not be affiliated to the same institution of the manuscript author(s) and should not have had any collaboration with any of the authors during the last 3 years.

Author Benefits

The corresponding author will be provided with a .pdf file of the article or alternatively one hardcopy of the journal free of charge.

Units of Measurement

Units of measurement should be presented simply and concisely using System International (SI) units.

Bibliographic Information

Commenced in 2010.
ISSN: 1847-6996
e-ISSN: 1847-7003

Published: semiannually

Copyright

Subscription Information

The annual subscription rate is 50€ for individuals, 25€ for students and 150€ for libraries.

Postal Address

Faculty of Electrical Engineering,
Computer Science and Information Technology Osijek,
Josip Juraj Strossmayer University of Osijek, Croatia
Kneza Trpimira 2b
31000 Osijek, Croatia

IJECES Copyright Transfer Form

(Please, read this carefully)

This form is intended for all accepted material submitted to the IJECES journal and must accompany any such material before publication.

TITLE OF ARTICLE (hereinafter referred to as "the Work"):

COMPLETE LIST OF AUTHORS:

The undersigned hereby assigns to the IJECES all rights under copyright that may exist in and to the above Work, and any revised or expanded works submitted to the IJECES by the undersigned based on the Work. The undersigned hereby warrants that the Work is original and that he/she is the author of the complete Work and all incorporated parts of the Work. Otherwise he/she warrants that necessary permissions have been obtained for those parts of works originating from other authors or publishers.

Authors retain all proprietary rights in any process or procedure described in the Work. Authors may reproduce or authorize others to reproduce the Work or derivative works for the author's personal use or for company use, provided that the source and the IJECES copyright notice are indicated, the copies are not used in any way that implies IJECES endorsement of a product or service of any author, and the copies themselves are not offered for sale. In the case of a Work performed under a special government contract or grant, the IJECES recognizes that the government has royalty-free permission to reproduce all or portions of the Work, and to authorize others to do so, for official government purposes only, if the contract/grant so requires. For all uses not covered previously, authors must ask for permission from the IJECES to reproduce or authorize the reproduction of the Work or material extracted from the Work. Although authors are permitted to re-use all or portions of the Work in other works, this excludes granting third-party requests for reprinting, republishing, or other types of re-use. The IJECES must handle all such third-party requests. The IJECES distributes its publication by various means and media. It also abstracts and may translate its publications, and articles contained therein, for inclusion in various collections, databases and other publications. The IJECES publisher requires that the consent of the first-named author be sought as a condition to granting reprint or republication rights to others or for permitting use of a Work for promotion or marketing purposes. If you are employed and prepared the Work on a subject within the scope of your employment, the copyright in the Work belongs to your employer as a work-for-hire. In that case, the IJECES publisher assumes that when you sign this Form, you are authorized to do so by your employer and that your employer has consented to the transfer of copyright, to the representation and warranty of publication rights, and to all other terms and conditions of this Form. If such authorization and consent has not been given to you, an authorized representative of your employer should sign this Form as the Author.

Authors of IJECES journal articles and other material must ensure that their Work meets originality, authorship, author responsibilities and author misconduct requirements. It is the responsibility of the authors, not the IJECES publisher, to determine whether disclosure of their material requires the prior consent of other parties and, if so, to obtain it.

- The undersigned represents that he/she has the authority to make and execute this assignment.
- For jointly authored Works, all joint authors should sign, or one of the authors should sign as authorized agent for the others.
- The undersigned agrees to indemnify and hold harmless the IJECES publisher from any damage or expense that may arise in the event of a breach of any of the warranties set forth above.

_____ _____

Author/Authorized Agent **Date**

CONTACT
International Journal of Electrical and Computer Engineering Systems (IJECES)
Faculty of Electrical Engineering, Computer Science and Information Technology Osijek
Josip Juraj Strossmayer University of Osijek
Kneza Trpimira 2b
31000 Osijek, Croatia
Phone: +38531224600,
Fax: +38531224605,
e-mail: ijeces@ferit.hr