**CLUSTERING BASED ON WHALE OPTIMIZATION ALGORITHM FOR IOT OVER WIRELESS NODES**

**ABSTRACT**

The Internet of Things relies heavily on wireless sensor networks (WSNs) (IoT). However, the energy resources of sensor nodes in a WSN-based IoT network are restricted. By grouping nodes into clusters to reduce the transmission distance between sensor nodes and base stations, a clustering protocol offers an effective method for ensuring node energy savings and extending network lifespan (BS). Current clustering protocols, on the other hand, have problems with the clustering mechanism, which has a negative impact on their efficiency. We suggest an enhanced energy-efficient clustering protocol consists of sensor nodes, collecting data from each sensor node is a challenging task, there exists a various types of clustering techniques are present for an efficient clustering. So, we are implementing a new Clustering Routing Algorithm Based on Whale Optimization Algorithm (WOA) which is so better at clustering of sensor nodes without a considerable energy overhead. Rather clustering through depending on the energy of nodes, WOA decides of whether a node will participate or not in the election of a Cluster Head (CH). Thereby, decreasing energy dissipation at a very considerable value.

**Keywords**: Internet of Things, Wireless sensor networks, Whale optimization algorithm, Unequal clustering , Network lifetime

**CHAPTER 1**

**INTRODUCTION**

In the Internet of Things (IoT) domain, each object or thing can use wireless communication to communicate with each other (Machado et al. 2013). Today, IoT has attracted the attention of societies, governments and industries for a wide range of applications, including smart homes, healthcare services, environmental monitoring, smart transportation, smart networks, security, fire detection, finance tracking, smart lighting, etc. (Abdul-Qawy and Srinivasulu 2018). In this context, wireless sensor networks (WSNs) play an important role in widening networks with low-cost smart devices that can be easily installed (Shah et al. 2018). In recent years, WSNs have attracted attention in many applications including environmental monitoring, predicting natural disasters, health monitoring and military applications (Heinzelman et al. 2002; Bozorgi et al. 2017). In these networks, nodes have limited battery and processing strength. Energy is a serious concern in these structures because, in many applications, either network’s nodes are not accessible or cannot be replaced (Bozorgi et al. 2016; Kuila et al. 2013). Since data transmission in wireless communications consumes more energy compared to processing, data routing and data transfer in these networks are of great importance (Kumar and Kumar 2016). Clustering is one of the effective methods for saving energy (Kuila et al. 2013). Clustering is a process that divides nodes into clusters. Each cluster has a cluster head (CH) and several cluster members (CMs). A CM senses the environment’s information and transmits them to the CH. CH collects and integrates the information transmitted by CMs and then transmits information to the base station (BS) (Kuila and Jana 2014; Afsar and Tayarani-N 2014). The clustering problem in IoT/WSN is an NP-hard problem. Computational intelligence has been widely employed for improving these challenges. Different computational intelligence methods including evolutionary algorithms (EAs) were used for routing in WSN (Khalil and Attea 2011). EAs have been used for optimization in many problems. In recent years, algorithms like a genetic algorithm (GA) (Deb 2000), differential evolution (DE) (Brest et al. 2006) and particle swarm optimization (PSO) (Clerc and Kennedy 2002; Eberhart and Shi 2004) were proposed as optimization algorithms (Ma and Simon 2011; Hosseinabadi et al. 2019; Sangaiah et al. 2019). WOA or Whale Optimization Algorithm is a new evolutionary method to solve optimizing problems. This algorithm drew inspiration from the hunting process in whales. Generating a random whale population is the first step of the algorithm. In the next step, WOA functions are used to lead the whales toward prey (the optimum solution). Existing clustering protocols are mainly time-oriented. These protocols are categorized as static, dynamic and hybrid. In the static method, clustering is done once and then CH rotation is performed. As an example,virtual concentric circle band-based clustering (VCCBC) (Kumar et al. 2011) and an energy-efficient protocol with static clustering (EEPSC) (Chaurasiya et al. 2011) are static clustering protocols. In static performance, overhead is low and the stability of the network lasts for a short time. A shortcoming of a static method is because energy is discharged at several nodes (Malathi et al. 2015). In the dynamic method, clustering is done at each round. One of the most well-known dynamic protocols is low-energy adaptive clustering hierarchy (LEACH) (Heinzelman et al. 2002). In a dynamic performance, the lifetime of the network can be improved but its overhead is high (Malathi et al. 2015). The hybrid method, not only improves stability and lifetime but also reduces overhead. Recently, hybrid static–dynamic methods have been proposed for clustering; among them, hybrid unequal clustering with layering protocol (HUCL) (Malathi et al. 2015) can be named. These methods employ both static and dynamic methods for clustering. After a specific number of rounds, clustering and formation of new clusters are performed. Also, CH rotation between clustering rounds and new cluster formation is performed. Against these time-oriented methods, the existence of those that are energy-oriented. Unlike, time-oriented approaches like LEACH (Heinzelman et al. 2002), which are selected at the beginning of each round of CH, in the proposed approach, CHs only change when their energy is not suitable. This way overhead is considerably reduced. Table 1 compares the existing methods. In the following, a different method has been investigated. Assuming that BS knows the network down to details, the clustering process in carried out by base station (Zanireh and Larijani 2015). The clustering process is implemented through WOA. Saving more energy and load balancing are the main objectives in the fitness function. The performance of the algorithm has been analyzed in terms of energy consumption. This way overhead is considerably reduced. Moreover, for transmitting data from CH to BS, energy-aware multi-hop routing is used. Also, a novel mechanism is used which prevents a node from sending conventional control messages like the head message and join message. Therefore, except for one control message at the beginning of the metaround with name status message, nodes do not broadcast any other control messages.

**CHAPTER 2**

**LITERATURE SURVEY**

**[1] Víctor Iglesias, Jesós Grajal, Omar Yeste-Ojeda, “Automatic modulation classifier for military applications,” 19th European Signal Processing Conference, 29 Aug.-2 Sept. 2011:**

Automatic modulation recognition plays an important role in several military and civilian applications. Depending on the application, latency can be the bottleneck for designing an automatic modulation classifier (AMC). In this paper, an AMC based on low complexity signal features to improve latency and percentage of real-time operation is designed for broad-band military applications.

**Summary:** Studied about the Automatic Modulation Classifier (AMC) in military applications.

**[2] Zhechen Zhu and Asoke K. Nandi, “Modulation Classification for Civilian Applications,” Wiley, 2014:**

Automatic Modulation Classification (AMC) has been a key technology in many military, security, and civilian telecommunication applications for decades. In military and security applications, modulation often serves as another level of encryption; in modern civilian applications, multiple modulation types can be employed by a signal transmitter to control the data rate and link reliability.

This book offers comprehensive documentation of AMC models, algorithms and implementations for successful modulation recognition. It provides an invaluable theoretical and numerical comparison of AMC algorithms, as well as guidance on state-of-the-art classification designs with specific military and civilian applications in mind.

**Summary:** Studied about the Modulation Classification for Civilian Applications.

**[3] A. Swami and B. M. Sadler, “Hierarchical digital modulation classification using cumulants,” IEEE Trans. Commun., vol.48, pp. 416- 429, 2000:**

A simple method, based on elementary fourth-order cumulants, is proposed for the classification of digital modulation schemes. These statistics are natural in this setting as they characterize the shape of the distribution of the noisy baseband I and Q samples. It is shown that cumulant-based classification is particularly effective when used in a hierarchical scheme, enabling separation into subclasses at low signal-to-noise ratio with small sample size. Thus, the method can be used as a preliminary classifier if desired. Computational complexity is order N, where N is the number of complex baseband data samples. This method is robust in the presence of carrier phase and frequency offsets and can be implemented recursively. Theoretical arguments are verified via extensive simulations and comparisons with existing approaches.

**Summary:** Studied about Hierarchical digital modulation classification using cumulants.

**[4** **A. K. Nandi and E. E. Azzouz, “Modulation recognition using artificial neural networks,” Signal Processing, pp. 165-175, 1997:**

This paper presents artificial neural networks (ANNs) for the recognition of either analogue or digital modulation types. Computer simulations of different types of band-limited, modulated signals corrupted by band-limited Gaussian noise sequence have been carried out to measure the performance of the ANN approach. The threshold SNR for the recognition of either analogue or digitally modulated signals with average success rate ⩾98% is found to be about 10 dB. Comparisons of results from the ANN approaches and the decision-tree methods are presented. **Summary:** Studied about Modulation recognition using artificial neural networks.

**[5] S. Z. Hsue and S. S. Soliman, “Automatic modulation classification using zero crossing,” IEE Radar and Signal Processing, vol. 137, pp. 459-464, 1990:**

A modulation recogniser that automatically reports modulation types of constant-envelope modulated signals is developed using zero-crossing techniques. The zero-crossing sampler, as a signal conditioner, has the advantage of providing accurate phase transition information over a wide dynamic frequency range. Signal parameters such as zero-crossing variance carrier-to-noise ratio (CNR) and carrier frequency are estimated. Phase difference and zero-crossing interval histograms play the role of features for modulation recognition. The classifier performance is given in the form of a confusion matrix. The simulation results obtained demonstrate that a reasonable average probability of correct classification is achievable for CNR ages; 15 dB**.**

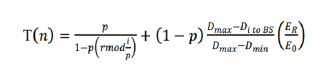
**SUMMARY:** Studied about Automatic modulation classification using zero crossing.

**CHAPTER 3**

**EXISTING METHOD**

Wireless sensor technology is growing rapidly, especially with many new Internet of Things (IoT) applications. In another side, researches are coming out with diversities of approaches to enhance and improve this technology trying to cover the needs in this era. The drawback of sensor Technologies is the low battery and short lifetime. So, most of the following researches considering to sophisticate these weaknesses and suggest different algorithms and approaches overcome these issues. Sharma proposed novel LEACH protocol in the heterogeneous network and compared the simulation results with LEACH Homogeneous system; They chose 100 \* 100 meters area to simulate the protocol. Sharma found that 10 nodes have more energy than the rest of 90 nodes which improves the system lifetime and enhanced wireless sensor network performance.

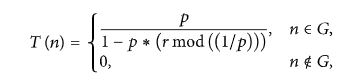
Naveen explored fifteen different types of clustering wireless sensor protocols which considered more in energyefficient and lifetime of the network system.. Prasad simulated LEACH using TDMA routing protocol. Also, they surveyed the previous approaches for selecting CH and improving the WSN performance such as Euclidian Distance from a node to BS, remaining energy and number of nodes in the same cluster. Increasing the number of dead nodes in the cluster would be the reason for shortening the WSN lifetime. Nandi [10] implemented a new protocol for choosing an optimal place for the BS, which overcomes the issues of delivering data and they compared the simulation result with the basic LEACH protocol with TDMA technique. Commonly when the BS located far away from the node, then transmitting data from a node to BS will cost more energy in the node, which leads to reduce the node lifetime and therefore reduce the network lifetime [10]. Moreover, packet delivery time would be reduced when the sink positioned in the center near the nodes [10]. The authors proposed an algorithm called Distance Based Cluster Head (DBCH) which the threshold value measured by the following equation :



where ER is the residual energy of the node for the current round and E0 is the initial energy. This algorithm proposes to select the closest node to the BS as a cluster head. This enhancement considered on two-parameter energy and distance. In addition, it considers the distance from the node to cluster head base station and compared the distance from node cluster head and BS. This study simulated the suggestions on a homogenous network, where all nodes have the same amount of energy. LEACH (Low-Energy Adaptive Clustering Hierarchy) protocol is a basic clustering-based routing protocol for WSNs.

LEACH routing protocol is a WSN routing algorithm designed by Heinzehnan et al. from MIT in the United States, which is the earliest typical hierarchical routing protocol [9]. LEACH protocol adopts the method of distributed CH election, in which some nodes are randomly selected from the network as CHs, and other nodes become cluster member nodes [10]. The CH broadcasts the message that it becomes a CH, and other nodes select the CH with the strongest received signal to join to form a cluster [9]. The cluster member node collects data and transmits it to the CH, which receives data and transmits it to the BS through single-hop communication. The CHs undertake the heavy tasks, including managing the member nodes of the cluster, collecting the data transmitted by the member nodes, data fusion, and intercluster forwarding. Therefore, to balance the energy consumption of nodes, CHs rotate, and the cluster structure is updated periodically. LEACH is a self-adaptive cluster formation protocol. The basic idea of the LEACH protocol is to divide the network into clusters of equal size. The CH rotates periodically, and each cycle is called a “round.” Each round is divided into two stages: the establishment stage of the cluster and the stable transmission stage [10].

In the establishment stage of the cluster, each node generates a random number from 0 to 1, and the threshold T(n) is calculated according to equation ([1](https://www.hindawi.com/journals/jece/2020/8059353/#EEq1)). Then, the random number generated by each node is compared with T(n). If the value is less than T(n), the node is selected as the CH:



where  p is the percentage of CH in all nodes, r is the number of current election rounds,

(r mod(1/p)) is the number of nodes that have been selected in this round, and G is the set of nodes without CHs selected in this round. After the end of each CH selection round, each selected CH broadcasts its message of becoming a CH to other nodes. After receiving the broadcast message, other nodes choose to join a cluster according to the received signal strength and send their joining message to the selected CH [11]. Each CH creates and assigns a TDMA schedule between each member node after its member nodes are joined. Then, end the cluster establishment stage and start the data transmission stage. Node becomes cluster head for the current round if the number is less than threshold T (n). Once node is elected as a cluster head then it cannot become cluster head again until all the nodes of the cluster have become cluster head once. This is useful for balancing the energy consumption. In the second step, non-cluster head nodes receive the cluster head advertisement and then send join request to the cluster head informing that they are members of the cluster under that cluster head. All non-cluster head nodes save a lot of energy by turning off their transmitter all the time and turn it on only when they have something to transmit to the cluster head [2]. In third step, each of the chosen cluster head creates a transmission schedule for the member nodes of their cluster. TDMA schedule is created according to the number of nodes in the cluster. Each node then transmits its data in the allocated time schedule [3].

In the data transmission stage, each member node sends data to the CH within its allocated period, and the CH transmits data to the BS after data fusion. Therefore, CHs consume more energy than member nodes. LEACH ensures that all nodes are equally likely to act as CHs employing cycle circulation so that the nodes consume energy in a relatively balanced manner. However, factors such as residual energy of nodes and distance from the BS are still not considered. The randomness of the CH election may lead to the death of the CH far away from the BS due to the rapid exhaustion of energy, which affects the survival time of the whole network.



Figure :Hierarchical or cluster based routing

Low Energy Adaptive Clustering Hierarchy (LEACH) protocol is a TDMA based MAC protocol. The principal aim of this protocol is to improve the lifespan of wireless sensor networks by lowering the energy consumption required to create and maintain Cluster Heads.

LEACH is a hierarchical protocol in which most nodes transmit to cluster heads, and the cluster heads aggregate and compress the data and forward it to the base station (sink). Each node uses a [stochastic](https://en.wikipedia.org/wiki/Stochastic) algorithm at each round to determine whether it will become a cluster head in this round. LEACH assumes that each node has a radio powerful enough to directly reach the base station or the nearest cluster head, but that using this radio at full power all the time would waste energy.

Nodes that have been cluster heads cannot become cluster heads again for P rounds, where P is the desired percentage of cluster heads. Thereafter, each node has a 1/P probability of becoming a cluster head again. At the end of each round, each node that is not a cluster head selects the closest cluster head and joins that cluster. The cluster head then creates a schedule for each node in its cluster to transmit its data.

All nodes that are not cluster heads only communicate with the cluster head in a TDMA fashion, according to the schedule created by the cluster head. They do so using the minimum energy needed to reach the cluster head, and only need to keep their radios on during their time slot.

LEACH also uses [CDMA](https://en.wikipedia.org/wiki/Code_division_multiple_access) so that each cluster uses a different set of CDMA codes, to minimize interference between cluster. The operation of LEACH protocol consists of several rounds with two phases in each [3] [4]: Set-up Phase and Steady Phase.

In the Set-up phase the main goal is to make cluster and select the cluster head for each of the cluster by choosing the sensor node with maximum energy. Leach protocol is a typically representation of hierarchical routing protocol. It is self-adaptive and self-organized [2]. Leach protocol uses round as unit, each round is made up of cluster set-up stage and steady state storage for the purpose of reducing unnecessary energy costs. Phases of leach protocol are as follows: A. Set-up phase In the set-up phase, the main goal is to make cluster and select the cluster head for each of the cluster by choosing the sensor node with maximum energy [3]. Set-up phase has three fundamental steps: 1. Cluster head advertisement 2. Cluster set up 3. Creation of transmission schedule During the first step cluster head sends the advertisement packet to inform the cluster nodes that they have become a cluster head on the basis of the following formula:

Steady Phase which is comparatively longer in duration than the set-up deals mainly with the aggregation of data at the cluster heads and transmission of aggregated data to the Base station. In steady phase, cluster nodes send their data to the cluster head. The member sensors in each cluster can communicate only with the cluster head via a single hop transmission. Cluster head aggregates all the collected data and forwards data to the base station either directly or via other cluster head along with the static route defined in the source code. After predefined time, the network again goes back to the set-up phase. The LEACH protocol adopts the concept of clustering and periodic data collection, which can reduce the data transmission between the nodes and the BS. Therefore, this protocol can not only reduce the energy loss, but also can extend the network lifetime. In addition, the CH uses the method of data aggregation, which can reduce correlated data locally. This method can also optimize the amount of data in the network and reduce energy consumption. Moreover, the time division multiple access (TDMA) schedule used by LEACH allows the member nodes to go into sleep mode, and this mechanism holds back the collision between clusters and extends the sensors’ battery life



Figure:Leach phases

A cluster head in the LEACH protocol is not stabilized; LEACH is established over the round concept and each round includes two stages: a setup stage and a steady-state stage. The setup stage is separated into advertisement aspect and cluster setup aspect, while the steady stage includes the creation of schedule and transferring of data .The LEACH protocol suits WSNs under the following suppositions: & Every sensor node is static, exactly alike, and charged with the identical quantity of initial energy. & Every node consumes energy at the same degree and is capable to identify its remaining energy and controls power transferring and distance. & All nodes can directly connect to every other node, as well as the sink node. & The sink node is determined and in a distance from the wireless network. Thus, the energy consumed by the sink node is ignored. & All nodes have transferred data in each period. The data transmitted by sobering nodes are connected and can be combined.

However, the density of nodes is not considered in the traditional LEACH protocol when selecting the CH. The placement of nodes and the expected number of CHs per round are considered when assigning CHs. Therefore, this protocol cannot ensure the uniform distribution of the CHs Additionally; the LEACH protocol does not consider the residual energy of nodes and the average energy of all nodes when selecting the CH. This will lead to a node with a lower energy being chosen as the CH. Thus, this protocol leads to the quick exhaustion of the node energy. Finally, the CH communicates directly with the BS by adopting a single hop communication mode.

LEACH protocol is threatened by the following types of attacks which degrade the performance of LEACH by dropping, altering, spoofing or replying the packets. A. Sybil Attack Most of the peer to peer networks face security threats due to Sybil attack [8], [9]. This attack is the most difficult attack to detect. In this attack, malicious node uses the identity of many other legitimate nodes to gain the data exchanged between the legitimate nodes. It affects the network by dropping vital packets, increasing traffic, lowering network lifetime etc. Encryption and authentication techniques can be used to prevent wireless sensor network from the Sybil attack. B. Selective Forwarding LEACH protocol is also susceptible to selective forwarding attack. In this kind of attack a malicious node places itself in the path where data is exchanged between the two legitimate nodes. It collects the data and instead of forwarding this node drops all the data. It is the case where the malicious node can easily be detected. The worst scenario of this attack is that when malicious node does not discard the entire data, but selectively forwards some of the non vital information. In this case it is very difficult detect the malicious node. C. HELLO Flooding Attack In many protocols sometimes it is required for node to transmit HELLO packets to advertise itself to its neighboring nodes. The nodes receiving these packets assume that it is within the range of the sender. But in case of malicious node, it continuously keeps on sending the HELLO packets and thus increases the network traffic and causes collisions. It also consumes the energy of the sensor nodes when these nodes receive large amount of HELLO packets continuously and thus lowering the lifetime of the wireless sensor networks. This type of attack is known as HELLO Flood attack .

To develop the LEACH protocol we consider the scenario consisting of the following network model and energy model as proposed by the author:

Network Architecture

The network model for development of the algorithm for clustering and routing consist of the following:

1) In our model we have 100 nodes with equal initial energy The base station is under human observation therefore

has unlimited power and the transmit power can be adjusted in an available range.

2) The nodes are considered to be immobile and their locations have been known with the help of either GPS or

node self-localization algorithms.

3) We have single sink node which can be moved. The distance between node in the network and the sink node is

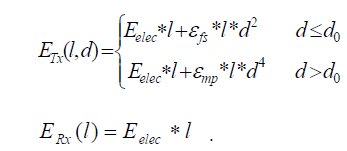
known by exchanging information. We can change the positions of sink node for analyzing the best position so that

minimum distance and low energy communication will take place.

4) CHs can use a single hop to the sink node and need more energy in transmitting the data to the base station and Cluster Member (CM) nodes use single-hop communication with CH as they are closer to the CHs.

5) Sensors periodically sense the environment and send the data to the Base Station

Radio Signal Propagation Model: The first order radio frequency energy consumption model to describe energy feather of the communication channel. The first order radio model can be divided into free-space model and multi-path fading model according to the distance between the sending node and receiving node. The protocol assumes that the communication channel is symmetrical, the energy consumption of l bits message between two nodes with a distance of d can be shown as below equations :



Where E (l,d) Tx is the energy consumption in transmitting l bits data to a node with a distance of d , E (l) Rx is the energy consumption in receiving l bits data. E elec equals the per bit energy consumption for transmitter and receiver circuit . E mp and E fs are the amplifier parameters of transmission corresponding to the multi-path fading model and the free-space model respectively. d0 is the threshold distance between multi-path fading model and the free-space model, If d0 < d , the channel approximates free-space model, the energy dissipation in transmitter amplifier is in direct ratio to d^2 . If d0 > d , the multi-path fading model will be employed and the energy dissipation is in direct ratio to d^4 .

Where, Eelec is the electronics energy; εfs and εmp are the amplifier energy of the free space model and the multipath model. As discussed in the previous section the operation of leach takes place in two phases. Initially all the nodes have equal probability to become the cluster head. Depending upon the random number selected the nodes themselves decide whether to become the cluster head or not. The nodes eligible to become cluster heads then broadcast its decision with larger signal strength so as to reach all the member nodes. ɛmp is the amplification power needed to

transmit the signal. The energy dissipated in setup phase can be calculated as follows.If k numbers of nodes are the CHs then to transmit the l bit message over a distance d each node needs



To receive this message from each CH the member nodes will need



When the nodes hear the cluster head message from the CHs they check for the signal with highest signal strength. The signal from the CH which is closest will have highest signal strength. So the node will join the corresponding head. For joining as member the nodes will send a request to the CH. To transmit the join request the energy dissipated is as follows:

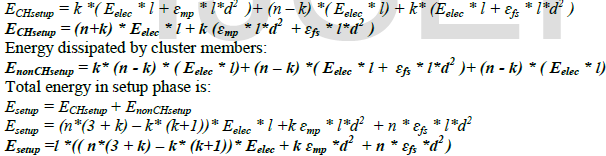


The schedule sent by CHs will be received by all the member nodes consuming the energy as follows:



Thus the total energy needed by the network in setup phase can be calculated as follows:

Energy dissipated by Cluster Head:



**DISADVANTAGES:**

1. One of the biggest disadvantage of LEACH is that when due to any reason Cluster head dies, the cluster will become useless because the data gathered by the cluster nodes would never reach its destination i.e. Base Station.

2. CH selection is the most difficult part of dynamic clustering.

3. LEACH disregards the BS and cluster head geographical positions, energy consumption.

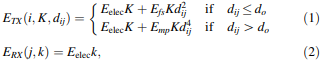
4. We have noticed that the cluster head missions are more than the ordinary nodes, so the cluster head consumes more energy than the others which one of the drawbacks of the LEACH algorithm.

**CHAPTER 4**

**PROPOSED METHOD**

The network model

In this study, a wireless network-based IoT and a BS with unlimited supply connected to a network are considered. In a network of size M 9 M, N nodes are distributed randomly. All nodes and the BS are fixed but they can adjust their transmission power according to the distance. At each round, data are sampled using sensor nodes and transmitted to BS for routing. Moreover, each node can be a CH or non-CH. CHs can collect and integrate data. The radio model of energy consumption is like LEACH protocol (Heinzelman et al. 2002). Energy consumption of transmitter and energy consumption of receiver are defined using Eqs. (1) and (2), respectively



where K is the number of data bits, and d is the distance between two nodes. Eelec(nj/bit) is energy consumption at each bit for transmitting or receiving data. Emp(pj/ bit 9 m-4 ) and Efs(pj/bit 9 m-2 ) are energy consumed at each bit for reinforcing transmitter considering transmission distance. In addition, do is obtained as do ¼ Efs=Emp p Data aggregation model used in simulations assumes that total aggregated information can be compressed to K bits by a set of N nodes in which each node collects K bits of information

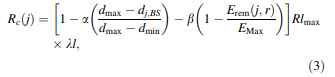
**The proposed algorithm**

In the proposed algorithm, a novel method, called NUWC (new unequal whale optimization algorithm clustering), has been designed for clustering by means of WOA (Mirjalili and Lewis 2016). Execution of NUWC is concentrated on one single node. This algorithm aims to reduce the consumption of energy. The distance between nodes can be determined through RSSI (received signal strength indicator). In order to reduce messaging overhead, a new message called status message has been designed and used instead of traditional controlling messages. Inter-cluster interactions have been managed by means of multi-hop approach which has energy saving considerations. The network operations are split into metarounds to decrease the clustering overhead. In a metaround, NUWC operations include a setup phase and data transmission phase. In the setup phase, BS runs WOA algorithm for clustering the nodes in the network. Data transmission phase is divided into several data transmission rounds. Each round consists of inter-cluster transmission and intracluster transmission. Inter-cluster routing to the BS is formulated in multi-hop form. In this method, the setup phase is only performed at the beginning of each metaround. In other words, when the energy of current CHs is less than, Eth, an energy threshold (Eth = Eavg - Estd); otherwise, current CHs are used for the new round. This threshold is adopted from Sabor et al. (2016). Eavg is the average energy of alive nodes, and Estd is the standard deviation of residual energy of all alive nodes. This reduces overhead and energy consumed for transmitting control packets and reduces computation time in the network. The performance of NUWC is shown in Fig. 1. The flowchart of the proposed algorithm is shown in Fig. 2

**Setup phase**

At the beginning of the first setup phase, the BS broadcasts request state messages in the environment. Each node calculates its distance from the BS considering RSSI. Then they transmit messages including location and energy information to the BS. BS performs 1-layering

2-determination of a node’s radius 3-determination of a node’s degree 4-probability calculation and 5-clustering based on WOA algorithm and then transmits a status message to nodes. Nodes are layered based on their distance from BS. This layering is performed only once for the determination of the node radius coefficient. Layering is independent of clustering. Empirically, and based on papers like reference Malathi et al. (2015), BS defines nodes at 4 levels. In the second stage, the BS obtained the radius of each node from Eq. (3):

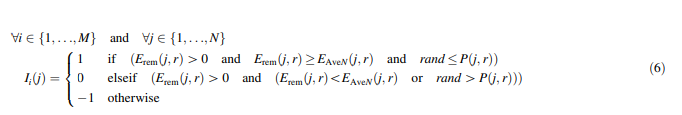


where Rc(j) is the radius of node j. Rlmax is the maximum radius of the node, which is predetermined. Erem(j,r) is the remained energy of node j at round r. EMax is the maximum energy capacity of a node. a and b are weight factors that are between 0 and 1. Moreover, in higher layers, Rlmax is multiplied by a coefficient like kl so that nodes which are closer to the BS have a smaller radius and those that are further have a larger radius. For this purpose, kl in the first layer is 1; and in the second layer, it is 1.25 and in the third and fourth layers, they are equal to 1.75. Equation (3) helps having clusters with unequal sizes. In the third stage, the BS determines the neighbor of each node. Neighbor of node i is the node whose distance from node i is shorter than its radius. The next stage determines the probability of becoming CH. At first, the BS calculates the average energy of neighbor nodes using Eq. (4)

CH. Nodes with suitable energy level have the chance to become CH and this chance is calculated as Eq. (5):

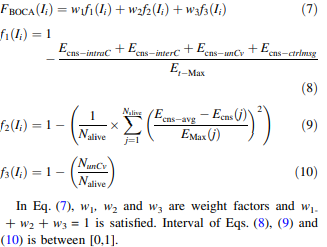


where P(j,r) is the probability of node j to become CH at round r. POpt is the optimal probability of becoming CH in the network which has been analyzed in reference to Heinzelman et al. (2002) and equal to 5% of the total number of nodes in the network. According to Eq. (5), the higher the energy level, the higher the chance to become CH.

The first generation of the population is a set of random solutions for the problem. The competency of each solution can be calculated by the fitness function. In the next stage, random solutions could be chosen and improved through specific functions. This procedure will continue until meeting stop criteria. In the framework of the WOA algorithm, a clustering solution is considered as an individual (I). In a WSN with N sensor nodes, non-CH nodes are considered as 0, CHs are considered as 1 and dead nodes are considered as - 1. A population of M individual solution can be specified as Eq. (6) 

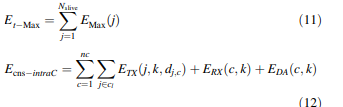
**Fitness function**

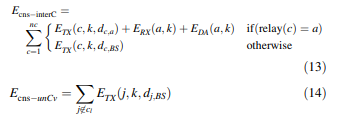
For each individual, the fitness function is calculated using Eq. (7). The proposed objective function is defined as maximization. For NUWC, the proposed fitness function is used to minimize total energy consumption in the network (Eq. (8)) and balance nodes’ consumption (Eq. (9)) and provide suitable coverage of network nodes by the CH (Eq. (10)):



**3 Minimize energy consumption (f1)**

Equation (8) is proposed to reduce energy consumption. In Eq. (8), Et–Max is the sum of the total battery capacity of alive nodes which is obtained using Eq. (11). In Eq. (8), Ecns–intraC is intracluster energy consumption and Ecns–interC is inter-cluster energy consumption based on inter-cluster multi-hop routing which is calculated using Eqs. (12) and (13). In addition, Ecns–unCv is the energy of nodes that have not been covered by the CH and should transmit their data to the BS directly. It is calculated using Eq. (14). Ecns–ctrlmsg is the total energy consumed by control messages

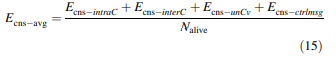




In Eq. (11), EMax(j) is the maximum capacity of node j. Nalive is the number of alive nodes. In Eq. (12), nc is the number of clusters. ETX(j,k,dj,c) is the energy consumed by the node j which is CM and transmit data (k bit) to their own CH (dj,c is the distance between node j and CHc). ERX(c,k) is the energy consumed at CH c for receiving data from CM or other CH in inter-cluster multi-hop routing. EDA(c,k) is the energy consumed for aggregating k bit data by CH c. In Eq. (13), ETX(c,k,dc,BS) is energy consumed by CH c which transmits data to the BS directly. ETX(c,k,dc.a) is the energy consumed for transmitting data in multi-hop format from CH c to CH a. In Eq. (14), ETX(j,k,dj, BS) is the energy consumed for node j which transmit data to BS directly (nodes which are not covered by the CH).

**Balance energy consumption (f2)**

In Eq. (9), f2 has been proposed to balance energy consumption and load balancing. This measure helps to select a state of the network in which nodes consume energy simultaneously in a balanced manner which increases stability. In this equation, Ecns–avg is the average energy consumed in the network which is calculated using Eq. (15). In Eq. (9), Ecns(j) is the energy consumed by node j and EMax(j) is the maximum capacity of node j.



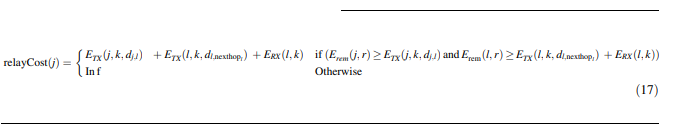
Coverage of network (f3) As discussed before, a normal node joins the closest CH; where the distance between a node and CH is less than the radius of the CH calculated using Eq. (4). Therefore, there might be a node in the network which has not been covered by the CH. Improper coverage of network nodes is adverse because the node should transmit its data to the BS directly. Therefore, the fitness function should be designed such that all network nodes are covered by the CHs. Equation (10) is proposed to cover all nodes of the network. NunCv is the number of nodes that are not in the radius of any CH. In other words, it is the number of nodes that have not been covered. Nalive is the number of alive nodes.

**Inter-cluster multi-hop routing**

In Eq. (13), selecting the next hop in routing for inter-cluster 5670 S. M. Bozorgi et al. 123 transmission at each individual is performed such that minimum energy is required for transmission and transmission quality is also guaranteed. BS at each individual calculates the cost of transmission to BS according to Eq. (16)



In this equation, ETX(j,k,dj, BS) is the energy required for transmitting k data bits from CH j to BS which its distance is d. Then, BS calculates the cost of transmitting data to the middle CH and next hop. The next hop is selected with minimum cost for multi-hop transmission such that conditions of Eq. (17) are satisfied. Evaluation parameter for transmission to BS or middle CH is calculated using Eq. (17):

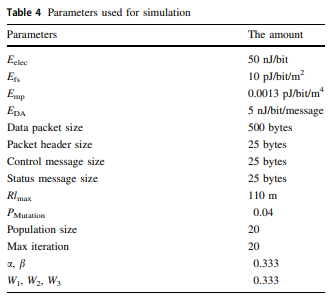


After clustering, the BS transmits information regarding the status of each node of the network considering its role. In this section, a new control message called status message is used. This message is divided into CH status message and CM status message. Considering the role of each node in the cluster, the BS transmits a control message associated with its role for that node. CM status message includes information regarding the ID of the node, status of the node (whether it is a CH or not), ID and distance from CH, and its turn to transmit node based on TDMA in intra-cluster transmission phase. Figure 3a shows the structure of this control message. Moreover, CH status message includes information regarding node’s ID, node status, ID and distance of next hop of this CH for inter-cluster transmission, number of nodes which are CMs. where ETX(j,k,dj,l) is the energy required for transmitting data from CH j to CH l and ETX(l,k,dl,nexthopl) is the energy required for transmitting data from CH l to the next hop. Thus, for transmitting data of CH j, BS considers two subsequent hops. ERX(l,k) is the energy required for receiving data by CH l. The BS selects CH of next hop such that it has enough energy for supporting reception of relay packet considering the number of CMs, data receiving cost and distance. After selecting CH with minimum relay cost, the next hop for CH j is selected. In addition, closer CHs will also transmit directly to the BS.

**Data transmission phase**

Data transmission includes some rounds and each round has two intra-cluster and inter-cluster transmission phases.

Round At the first phase of each round, according to time division multiple accesses (TDMA) scheduling performed by the BS, nodes transmit their data and their amount of energy to the CH. When a CH receives a packet from a node which is a CM, integrates and aggregates it. After transmitting information of nodes to CH, the CHs transmit their packets to the BS in multi-hop format. Routing is formulated such that energy consumed for transmitting packets is minimized. At this phase, the carrier sense multiple access (CSMA) method is used for transmitting data. Each CH transmits data to the BS according to the route constructed by the BS.



**CHAPTER 5**

**ADVANTAGES AND APPLICATIONS**

**Advantages:**

1. Cluster Heads are calculated based on the fitness function.

2. Calculation of fitness function at every stage of the process makes the CH selection even more robust.

3. Clustering is done through fitness function which results in a better clustering than the previous existing methods.

**Applications:**

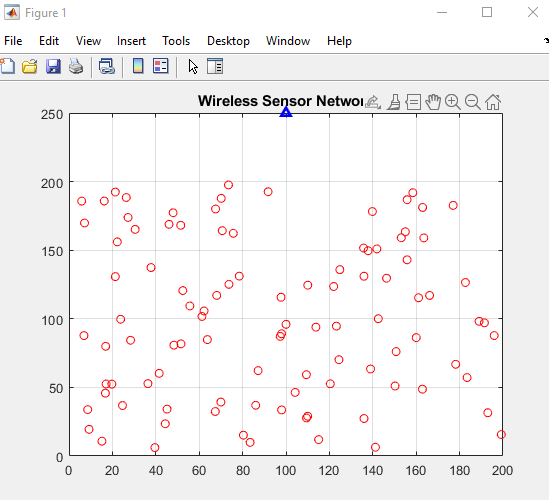
1. IOT environment

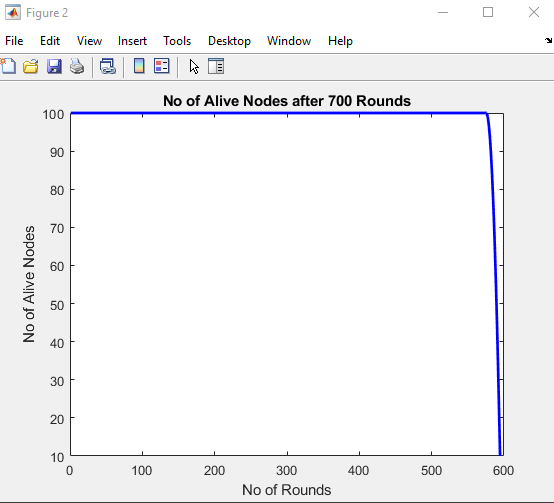
2. Satellite communication

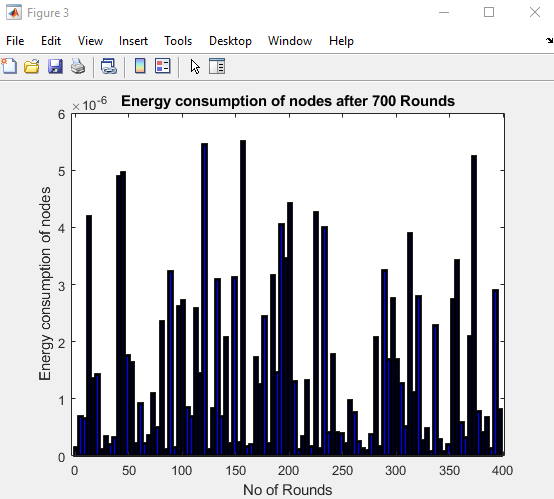
3. Weather and temperature forecasting

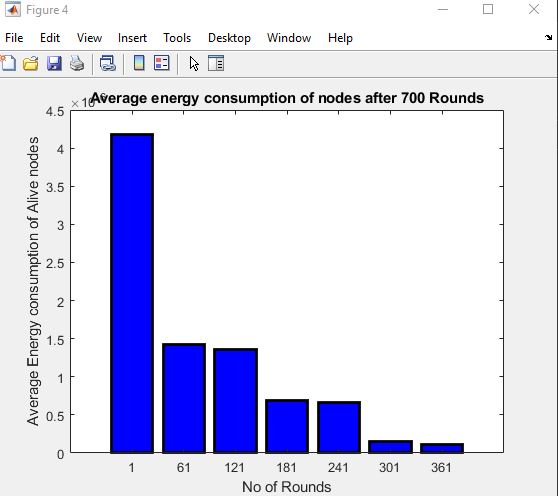
**CHAPTER 6**

**RESULTS**









**CHAPTER 7**

**CONCLUSION**

Using IoT technology which is implemented through wireless nodes is a new trend in the world. Limited energy is the most important challenge in these networks. Sending and receiving messages is the main reason for consuming energy. Hence, designing an efficient method for routing and managing messages could have a significant impact on the network’s lifespan. In the proposed method, a concentrated clustering method, which has been implemented by WOA, is designed to tackle the problem. Inter-cluster interactions have been managed by means of multi-hop approach which has energy saving considerations. Saving more energy, load balancing and CH coverage at different points of the network are the main objectives in fitness function control messages are replaced by status messages in order to reduce messaging overhead. The performance of the algorithm has been analyzed in terms of energy consumption. Results suggest that NUWC managed to improve the level of stability and lifespan. Moreover, it can balance energy consumption and throughput of the network. In future works, it is intended to develop the proposed method for IoT over energy harvesting wireless nodes

**REFERENCES**

Abdul-Qawy ASH, Srinivasulu T (2018) SEES: a scalable and energy-efficient scheme for green IoT-based heterogeneous wireless nodes. J Ambient Intell Humaniz Comput 10:1571–1596

Afsar MM, Tayarani-N MH (2014) Clustering in sensor networks: a literature survey. J Netw Comput Appl 46:198–226

Azharuddin M, Jana PK (2016) Particle swarm optimization for maximizing lifetime of wireless sensor networks. Comput Electr Eng 51:26–42

Bozorgi SM, Amiri MG, Rostami AS, Mohanna F (2016) A novel dynamic multi-hop clustering protocol based on renewable energy for energy harvesting wireless sensor networks. In: Conference proceedings of 2015 2nd international conference on knowledge-based engineering and innovation, KBEI 2015

Bozorgi SM, Shokouhi Rostami A, Hosseinabadi AAR, Balas VE (2017) A new clustering protocol for energy harvesting-wireless sensor networks. Comput Electr Eng 64:233–247

Brest J, Zumer V, Maucec MS (2006) Self-adaptive differential evolution algorithm in constrained real-parameter optimization. IEEE Int Conf Evol Comput 2006:215–222

Chaurasiya SK, Pal T, Bit SD (2011) An enhanced energy-efficient protocol with static clustering for WSN. In: International conference on information networking 2011, ICOIN 2011, pp 58–63

Clerc M, Kennedy J (2002) The particle swarm—explosion, stability, and convergence in a\nmultidimensional complex space. IEEE Trans Evol Comput 6(1):58–73

Deb K (2000) An efficient constraint handling method for genetic algorithms. Comput Methods Appl Mech Eng 186:311–33

Eberhart R, Shi Y (2004) Guest editorial special issue on particle swarm optimization. IEEE Trans Evol Comput 8(3):201–203

Gupta V, Pandey R (2016) An improved energy aware distributed unequal clustering protocol for heterogeneous wireless sensor networks. Eng Sci Technol Int J 19(2):1050–1058

**BIBLIOGRAPHY**

**Introduction To Matlab**

What Is MATLAB?

The name MATLAB stands for Matrix Laboratory. The software is built up around vectors and matrices. This makes the software particularly useful for linear algebra but MATLAB is also a great tool for solving algebraic and differential equations and for numerical integration. MATLAB has powerful graphic tools and can produce nice pictures in both 2D and 3D. It is also a programming language, and is one of the easiest programming languages for writing mathematical programs. These factors make MATLAB an excellent tool for teaching and research.

MATLAB was written originally to provide easy access to matrix software developed by the LINPACK (linear system package) and EISPACK (Eigen system package) projects. It integrates computation, visualization, and programming environment. Furthermore, MATLAB is a modern programming language environment: it has sophisticated data structures, contains built-in editing and debugging tools, and supports object-oriented programming. MATLAB has many advantages compared to conventional computer languages (e.g., C, FORTRAN) for solving technical problems.

MATLAB abilities a family of add-on software program utility software application software program software utility software-unique solutions called toolboxes. Very essential to maximum customers of MATLAB, toolboxes assist you to studies and observe specialized technology. Toolboxes are entire collections of MATLAB abilities (M-files) that increase the MATLAB surroundings to remedy precise schooling of problems. Areas in which toolboxes are to be had embody signal processing, manipulate systems, neural networks, fuzzy correct judgment, wavelets, simulation, and hundreds of others.

It has powerful built-in routines that enable a very wide variety of computations. It also has easy to use graphics commands that make the visualization of results immediately available. Specific applications are collected in packages referred to as toolbox. There are toolboxes for signal processing, symbolic computation, control theory, simulation, optimization, and several other fields of applied science and engineering. MATLAB is an interactive system whose basic data element is an array that does not require dimensioning. The software package has been commercially available since 1984 and is now considered as a standard tool at most universities and industries worldwide.

**Brief History of MATLAB:**

Cleve Moler, the chairman of the computer science department at the University of New Mexico, started developing MATLAB in the late 1970s. The first MATLAB® was not a programming language; it was a simple interactive matrix calculator. There were no programs, no toolboxes, no graphics and no ODEs or FFTs. He designed it to give his student’s access to LINPACK and EISPACK without them having to learn FORTRAN. It soon spread to other universities and found a strong audience within the applied mathematics community. The mathematical basis for the first version of MATLAB was a series of research papers by J. H. Wilkinson and 18 of his colleagues, published between 1965 and 1970 and later collected in Handbook for Automatic Computation, Volume II, Linear Algebra*,* edited by Wilkinson and C. Reinsch. These papers present algorithms, implemented in Algol 60, for solving matrix linear equation and Eigen value problems.

In the 1970s and early 1980s, I was teaching Linear Algebra and Numerical Analysis at the University of New Mexico and wanted my students to have easy access to LINPACK and EISPACK without writing FORTRAN programs. By “easy access,” I meant not going through the remote batch processing and the repeated edit-compile-link-load-execute process that was ordinarily required on the campus central mainframe computer. Jack little, an engineer, was exposed to it during a visit Moler made to Stanford University in 1983. Recognizing its commercial potential, he joined with Moler and Steve Bangert. They rewrote MATLAB in C and founded Math Works in 1984 to continue its development. These rewritten libraries were known as JACKPAC. In 2000, MATLAB was rewritten to use a newer set of libraries for matrix manipulation, LAPACK. MATLAB was first adopted by researchers and practitioners in control engineering, Little's specialty, but quickly spread to many other domains. It is now also used in education, in particular the teaching of linear algebra and numerical analysis, and is popular amongst scientists involved in video processing**.**

## **EISPACK and LINPACK**:

In 1970, a group of researchers at Argonne National Laboratory proposed to the U.S. National Science Foundation (NSF) to “explore the methodology, costs, and resources required to produce, test, and disseminate high-quality mathematical software and to test, certify, disseminate, and support packages of mathematical software in certain problem areas.” The group developed EISPACK (Matrix Eigen system Package) by translating the Algol procedures for Eigen value problems in the handbook into FORTRAN and working extensively on testing and portability. The first version of EISPACK was released in 1971 and the second in 1976.

In 1975, four of us Jack Dongarra, Pete Stewart, Jim Bunch, and myself proposed to the NSF another research project that would investigate methods for the development of mathematical software. A byproduct would be the software itself, dubbed LINPACK, for Linear Equation Package. This project was also centered at Argonne. LINPACK originated in FORTRAN; it did not involve translation from Algol. The package contained 44 subroutines in each of four numeric precisions. In a sense, the LINPACK and EISPACK projects were failures. We had proposed research projects to the NSF to “explore the methodology, costs, and resources required to produce, test, and disseminate high-quality mathematical software.” We never wrote a report or paper addressing those objectives. We only produced software.

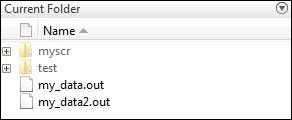
So, I studied Niklaus Wirth’s book Algorithms + Data Structures *=* Programs and learned how to parse programming languages. I wrote the first MATLAB an acronym for Matrix Laboratory in FORTRAN, with matrix as the only data type. The project was a kind of hobby, a new aspect of programming for me to learn and something for my students to use. There was never any formal outside support, and certainly no business plan. This first MATLAB was just an interactive matrix calculator. This snapshot of the start-up screen shows all the reserved words and functions. There are only 71. To add another function, you had to get the source code from me, write a FORTRAN subroutine, add your function name to the parse table, and recompile MATLAB.

**Starting MATLAB:**

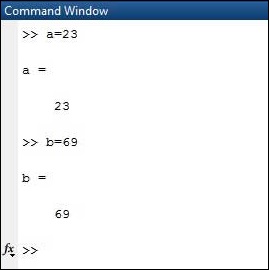
After logging into your account, you can enter MATLAB by double-clicking on the MATLAB shortcut icon (MATLAB 7.0.4) on your Windows desktop. When you start MATLAB, a special window called the MATLAB desktop appears. The desktop is a window that contains other windows. The major tools within or accessible from the desktop are:

* The Command Window
* The Command History
* The Workspace
* The Current Directory
* The Help Browser

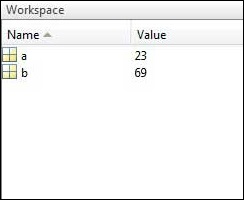
**Current Folder:** This panel allows you to access the project folders and files.



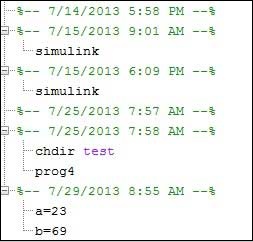
**Command Window:** This is the main area where commands can be entered at the command line. It is indicated by the command prompt (>>).



**Workspace:**  The workspace shows all the variables created and/or imported from files.



**Command History:** This panel shows or return commands that are entered at the command line.



**Help Browser:**

The critical way to get assist online is to use the MATLAB help browser, opened as a separate window every through clicking at the question mark photograph (?) on the computing tool toolbar, or through manner of typing assist browser on the spark off in the command window. The assist Browser is an internet browser blanketed into the MATLAB computing tool that shows a Hypertext Markup Language (HTML) files. The Help Browser consists of panes, the help navigator pane, used to find out information, and the show pane, used to view the information. Self-explanatory tabs apart from navigator pane are used to performs are searching out.

**MATLAB language:**

This is a high-level matrix/array language with control flow statements, functions, data structures, input/output, and object-oriented programming features. It allows both "programming in the small" to rapidly create quick and dirty throw-away programs, and "programming in the large" to create complete large and complex application programs.

**MATLAB working environment:**

This is the set of tools and facilities that you work with as the MATLAB user or programmer. It includes facilities for managing the variables in your workspace and importing and exporting data. It also includes tools for developing, managing, debugging, and profiling M-files, MATLAB's applications.

**MATLAB mathematical function library:**

This is a vast collection of computational algorithms ranging from elementary functions like sum, sine, cosine, and complex arithmetic, to more sophisticated functions like matrix inverse, matrix eigenvalues, Bessel functions, and fast Fourier transforms.

**MATLAB Application Program Interface (API):**

This is a library that allows you to write C and FORTRAN programs that interact with MATLAB. It includes facilities for calling routines from MATLAB (dynamic linking), calling MATLAB as a computational engine, and for reading and writing MAT-files.

**MATLAB DESKTOP:**

MATLAB Desktop is the precept MATLAB utility window. The computing tool includes five sub home windows, the command window, the workspace browser, the modern-day-day list window, the command records window, and one or greater decide domestic windows, which is probably confirmed high-quality on the identical time due to the truth the client suggests a photo. The command window is in which the character types MATLAB instructions and expressions at the spark off (>>) and in which the output of these commands is displayed. MATLAB defines the workspace because the set of variables that the client creates in a bit consultation. The workspace browser suggests those variables and some facts about them. Double clicking on a variable within the workspace browser launches the Array Editor, which may be used to gain statistics and profits instances edit exceptional homes of the variable.

The modern-day-day-day Directory tab above the workspace tab suggests the contents of the cutting-edge list, whose path is shown inside the modern-day list window. For example, in the home windows on foot machine the path is probably as follows: C: MATLAB Work, indicating that listing “artwork” is a subdirectory of the number one list “MATLAB”; WHICH IS INSTALLED IN DRIVE C. Clicking on the arrow within the modern list window suggests a listing of these days used paths. Clicking at the button to the right of the window permits the individual to trade the present day listing. MATLAB uses a seeking out path to find out M-documents and one-of-a-type MATLAB associated documents, which can be put together in directories within the computer document tool. Any report run in MATLAB need to be dwelling in the modern-day-day listing or in a list that is on is looking for course. By default, the documents supplied with MATLAB and math works toolboxes are included inside the searching out direction. The first-rate manner to look which directories are on the searching out route. The satisfactory manner to appearance which directories are speedy the quest route, or to characteristic or regulate a searching for course, is to pick out outset path from the File menu the computing device, and then use the set course talk discipline. It is proper exercise to feature any generally used directories to the hunt route to avoid again and again having the exchange the cutting-edge-day listing.

The Command History Window contains a file of the instructions a person has entered in the command window, together with every contemporary-day and former MATLAB periods. Previously entered MATLAB instructions can be determined on and re-completed from the command statistics window thru proper clicking on a command or series of commands. This movement launches a menu from which to select numerous options similarly to executing the commands. This is useful to select out abilities options in addition to executing the instructions. This is a beneficial feature at the equal time as experimenting with numerous commands in a piece session.

**Using the MATLAB Editor to create M-Files:**

The MATLAB editorial manager is a literary substance proofreader particular for growing M-facts and a graphical MATLAB debugger. The supervisor can seem in a window through command facts technique for itself, or it is probably a right-clicking inside the PC. M-information this gadget signified through the use of the expansion .M, as in pixel up.M. The MATLAB editorial supervisor window has a few draws down menus for obligations collectively with sparing, seeing, and troubleshooting facts. Since it plays more than one easy test and furthermore affects utilization of shade to separate among exclusive variables of code, this article editorial supervisor is often supported due to reality the system of a need for composing and altering M-talents. To open the manager, type at enact opens the M-document filename. M in a supervisor window, sorted out for enhancing. As stated earlier than, the file should be inside the cutting-edge posting, or in a posting in the seeking out direction.

## **Features of MATLAB:**

Following are the basic features of MATLAB.

* It is a high-level language for numerical computation, visualization and application development.
* It also provides an interactive environment for iterative exploration, design and problem solving.
* It provides vast library of mathematical functions for linear algebra, statistics, Fourier analysis, filtering, optimization, numerical integration and solving ordinary differential equations.
* It provides built-in graphics for visualizing data and tools for creating custom plots.
* MATLAB's programming interface gives development tools for improving code quality maintainability and maximizing performance.
* It provides tools for building applications with custom graphical interfaces.
* It provides functions for integrating MATLAB based algorithms with external applications and languages such as C, Java, .NET and Microsoft Excel.

## **Uses of MATLAB:**

MATLAB is widely used as a computational tool in science and engineering encompassing the fields of physics, chemistry, math and all engineering streams. It is used in a range of applications including

* Signal Processing and Communications
* Video and Video Processing
* Control Systems
* Test and Measurement
* Computational Finance
* Computational Biology

**Applications of MATLAB:**

MATLAB can be used as a tool for simulating various electrical networks but the recent developments in MATLAB make it a very competitive tool for Artificial Intelligence, Robotics, Video processing, Wireless communication, Machine learning, Data analytics and whatnot. Though it’s mostly used by circuit branches and mechanical in the engineering domain to solve a basic set of problems its application is vast. It is a tool that enables computation, programming and graphically visualizing the results. The basic data element of MATLAB as the name suggests is the Matrix or an array. MATLAB toolboxes are professionally built and enable you to turn your imaginations into reality. MATLAB programming is quite similar to C programming and just requires a little brush up of your basic programming skills to start working with.

Below are a few applications of MATLAB –

* **Statistics and machine learning (ML)**

This toolbox in MATLAB can be very handy for the programmers. Statistical methods such as descriptive or inferential can be easily implemented. So is the case with machine learning. Various models can be employed to solve modern-day problems. The algorithms used can also be used for big data applications.

* **Curve fitting**

The curve fitting toolbox helps to analyze the pattern of occurrence of data. After a particular trend which can be a curve or surface is obtained, its future trends can be predicted. Further plotting, calculating integrals, derivatives, interpolation, etc. can be done.

* **Control systems**

Systems nature can be obtained. Factors such as closed-loop, open-loop, its controllability and observability, Bode plot, NY Quist plot, etc. can be obtained. Various controlling techniques such as PD, PI and PID can be visualized. Analysis can be done in the time domain or frequency domain.

* **Signal Processing**

Signals and systems and digital signal processing are taught in various engineering streams. But MATLAB provides the opportunity for proper visualization of this. Various transforms such as Laplace, Z, etc. can be done on any given signal. Theorems can be validated. Analysis can be done in the time domain or frequency domain. There are multiple built-in functions that can be used.

* **Mapping**  
  Mapping has multiple applications in various domains. For example, in Big Data, the Map Reduce tool is quite important which has multiple applications in the real world. Theft analysis or financial fraud detection, regression models, contingency analysis, predicting techniques in social media, data monitoring, etc. can be done by data mapping.
* **Deep learning**

It’s a subclass of machine learning which can be used for speech recognition, financial fraud detection, and medical video analysis. Tools such as time-series, Artificial neural network (ANN), Fuzzy logic or combination of such tools can be employed.

* **Financial analysis**

An entrepreneur before starting any endeavor needs to do a proper survey and the financial analysis in order to plan the course of action. The tools needed for this are all available in MATLAB. Elements such as profitability, solvency, liquidity, and stability can be identified. Business valuation, capital budgeting, cost of capital, etc. can be evaluated.

* **Video processing**

The most common application that we observe almost every day are bar code scanners, selfie (face beauty, blurring the background, face detection), video enhancement, etc. The digital video processing also plays quite an important role in transmitting data from far off satellites and receiving and decoding it in the same way. Algorithms to support all such applications are available.

* **Text analysis**

Based on the text, sentiment analysis can be done. Google gives millions of search results for any text entered within a few milliseconds. All this is possible because of text analysis. Handwriting comparison in forensics can be done. No limit to the application and just one software which can do this all.

* **Electric vehicles designing**

Used for modeling electric vehicles and analyze their performance with a change in system inputs. Speed torque comparison, designing and simulating of a vehicle, whatnot.

* **Aerospace**

This toolbox in MATLAB is used for analyzing the navigation and to visualize flight simulator.

* **Audio toolbox**

Provides tools for audio processing, speech analysis, and acoustic measurement. It also provides algorithms for audio and speech feature extraction and audio signal transformation.

**COMMUNICATION:**

Communications System Toolbox™ offers algorithms and gear for the layout, simulation, and analysis of communications systems. These capabilities are furnished as MATLAB ® features, MATLAB System gadgets™, and Simulink ® blocks. The machine toolbox includes algorithms for source coding, channel coding, interleaving, modulation, equalization, synchronization, and channel modeling. Tools are supplied for bit blunders charge evaluation, producing eye and constellation diagrams, and visualizing channel characteristics. The machine toolbox additionally provides adaptive algorithms that allow you to version dynamic communications structures that use OFDM, OFDMA, and MIMO techniques. Algorithms support fixed-point facts arithmetic and C or HDL code era.

**Key Features**

▪ Algorithms for designing the physical layer of communications systems, which includes supply coding, channel coding, interleaving, modulation, channel fashions, MIMO, equalization, and synchronization

▪ GPU-enabled System objects for computationally intensive algorithms together with Turbo, LDPC, and Viterbi decoders

▪ Interactive visualization equipment, consisting of eye diagrams, constellations, and channel scattering capabilities

▪ Graphical tool for evaluating the simulated bit mistakes rate of a machine with analytical outcomes

▪ Channel models, consisting of AWGN, Multipath Rayleigh Fading, Rician Fading, MIMO Multipath Fading, and

LTE MIMO Multipath Fading

▪ Basic RF impairments, along with nonlinearity, section noise, thermal noise, and section and frequency offsets

▪ Algorithms available as MATLAB features, MATLAB System objects, and Simulink blocks

▪ Support for fixed-point modeling and C and HDL code technology

**System Design, Characterization, and Visualization:**

The layout and simulation of a communications gadget requires analyzing its reaction to the noise and interference inherent in real-world environments, reading its behavior the usage of graphical and quantitative manner, and determining whether the resulting overall performance meets requirements of acceptability. Communications System Toolbox implements a selection of obligations for communications machine layout and simulation. Many of the functions, System objects™, and blocks inside the device toolbox perform computations associated with a specific thing of a communications gadget, consisting of a demodulator or equalizer. Other talents are designed for visualization or evaluation.

**System Characterization**

The system toolbox offers several standard methods for quantitatively characterizing system performance:

▪ Bit error rate (BER) computations

▪ Adjacent channel power ratio (ACPR) measurements

▪ Error vector magnitude (EVM) measurements

▪ Modulation error ratio (MER) measurements

Because BER computations are fundamental to the characterization of any communications system, the system toolbox provides the following tools and capabilities for configuring BER test scenarios and accelerating BER simulations:

**BER tool**— A graphical user interface that enables you to analyze BER performance of communications systems. You can analyze performance via a simulation-based, semi analytic, or theoretical approach.

**Error Rate Test Console** — A MATLAB object that runs simulations for communications systems to measure error rate performance. It supports user-specified test points and generation of parametric performance plots and surfaces. Accelerated performance can be realized when running on a multi core computing platform.

**Multi core and GPU acceleration** — A capability provided by Parallel Computing Toolbox™ that enables you to accelerate simulation performance using multi core and GPU hardware within your computer.

**Distributed computing and cloud computing support** — Capabilities provided by Parallel Computing Toolbox and MATLAB Distributed Computing Server™ that enable you to leverage the computing power of your server farms and the Amazon EC2 Web service. Performance Visualization. The system toolbox provides the following capabilities for visualizing system performance:

**Channel visualization tool** — For visualizing the characteristics of a fading channel

**Eye diagrams and signal constellation scatter plots** — for a qualitative, visual understanding of system behavior that enables you to make initial design decisions

**Signal trajectory plots** — for a continuous picture of the signal’s trajectory between decision points

**BER plots** — for visualizing quantitative BER performance of a design candidate, parameterized by metrics such as SNR and fixed-point word size

**Analog and Digital Modulation**

Analog and digital modulation strategies encode the facts circulation into a sign this is appropriate for transmission. Communications System Toolbox presents some of modulation and corresponding demodulation abilities. These talents are available as MATLAB features and gadgets, MATLAB System Modulation sorts provided by the toolbox are:

**Source and Channel Coding**

Communications System Toolbox affords source and channel coding talents that can help you develop and compare communications architectures fast, enabling you to discover what-if eventualities and avoid the need to create coding competencies from scratch.

**Source Coding**

Source coding, also referred to as quantization or signal formatting, is a manner of processing facts a good way to lessen redundancy or prepare it for later processing. The system toolbox offers a diffusion of styles of algorithms for imposing source coding and interpreting, inclusive of:

▪ Quantizing

▪ Companding (*µ*-law and A-law)

▪ Differential pulse code modulation (DPCM)

▪ Huffman coding

▪ Arithmetic coding

**Channel Coding**

▪ orthogonal area-time block code (OSTBC) (encoder and decoder for MIMO channels)

▪ Turbo encoder and decoder examples

The gadget toolbox offers application functions for developing your personal channel coding. You can create generator polynomials and coefficients and syndrome deciphering tables, in addition to product parity-take a look at and generator matrices.

The system toolbox additionally presents block and convolutional interleaving and deinters leaving functions to reduce facts errors as a result of burst mistakes in a conversation machine:

**Block,** including General block interleaver, algebraic interleaver, helical scan interleaver, matrix interleaver, and random interleaver.

**Convolutional,** including General multiplexed interleaver, convolutional interleaver, and helical interleaver

**Channel Modeling and RF Impairments**

Channel Modeling

Communications System Toolbox provides algorithms and tools for modeling noise, fading, interference, and different distortions which might be commonly found in communications channels. The system toolbox supports the subsequent styles of channels:

▪ Additive white Gaussian noise (AWGN)

▪ Multiple-enter multiple-output (MIMO) fading

▪ Single-enter single-output (SISO), Rayleigh, and Rician fading

▪ Binary symmetric

A MATLAB channel object provides a concise, configurable implementation of channel models, enabling you to

specify parameters such as:

▪ Path delays

▪ Average path gains

▪ Maximum Doppler shifts

▪ K-Factor for Rician fading channels

▪ Doppler spectrum parameters

For MIMO systems, the MATLAB MIMO channel object expands these parameters to also include:

▪ Number of transmit antennas (up to 8)

▪ Number of receive antennas (up to 8)

▪ Transmit correlation matrix

▪ Receive correlation matrix

To combat the effects noise and channel corruption, the system toolbox provides block and convolutional coding and decoding techniques to implement error detection and correction. For simple error detection with no inherent correction, a cyclic redundancy check capability is also available. Channel coding capabilities provided by the system toolbox include:

▪ BCH encoder and decoder

▪ Reed-Solomon encoder and decoder

▪ LDPC encoder and decoder

▪ Convolutional encoder and Viterbi decoder

****

**RF Impairments**

To model the effects of a non-ideal RF front end, you can introduce the following impairments into your communications system, enabling you to explore and characterize performance with real-world effects:

▪ Memory less nonlinearity

▪ Phase and frequency offset

▪ Phase noise

▪ Thermal noise

You can include more complex RF impairments and RF circuit models in your design using SimRF™.

****

**Equalization and Synchronization**

Communications System Toolbox lets you discover equalization and synchronization strategies. These techniques are usually adaptive in nature and tough to design and symbolize. The machine toolbox affords algorithms and tools that will let you swiftly select the proper approach on your communications machine. Equalization To compare one-of-a-kind techniques to equalization, the device toolbox offers you with adaptive algorithms which include:

▪ LMS

▪ Normalized LMS

▪ Variable step LMS

▪ Signed LMS

▪ MLSE (Viterbi)

▪ RLS

▪ CMA

These adaptive equalizers are available as nonlinear decision feedback equalizer (DFE) implementations and as

Linear (symbol or fractionally spaced) equalizer implementations.

**Synchronization**

The device toolbox provides algorithms for each service segment synchronization and timing phase synchronization. For timing section synchronization, the machine toolbox presents a MATLAB Timing Phase Synchronizer object that offers the following implementation techniques:

▪ Early-late gate timing method

▪ Gardner’s method

▪ Fourth-order nonlinearity method

**Stream Processing in MATLAB and Simulink**

Most verbal exchange structures cope with streaming and frame-primarily based statistics using a aggregate of temporal processing and simultaneous multi frequency and multichannel processing. This form of streaming multidimensional processing can be visible in superior communication architectures consisting of OFDM and MIMO. Communications System Toolbox enables the simulation of advanced communications structures via helping move processing and frame-based simulation in MATLAB and Simulink. In MATLAB, circulate processing is enabled by way of System items™, which use MATLAB objects to symbolize time-based and facts-driven algorithms, sources, and sinks. System objects implicitly manipulate many information of flow processing, including information indexing, buffering, and management of set of rules state. You can mix System gadgets with fashionable MATLAB functions and operators. Most System items have a corresponding Simulink block with the identical abilities. Simulink handles circulation processing implicitly with the aid of coping with the float of information thru the blocks that make up a Simulink model. Simulink is an interactive graphical environment for modeling and simulating dynamic systems that uses hierarchical diagrams to symbolize a machine version. It includes a library of widespread-reason, predefined blocks to represent algorithms, resources, sinks, and device hierarchy.

**Implementing a Communications System**

Fixed-Point Modeling Many communications systems use hardware that requires a fixed-point representation of your design.

Communications System Toolbox supports fixed-point modeling in all relevant blocks and System objects™ with tools that help you configure fixed-point attributes.

Fixed-point support in the system toolbox includes:

▪ Word sizes from 1 to 128 bits

▪ Arbitrary binary-point placement

▪ Overflow handling methods (wrap or saturation)

▪ Rounding methods: ceiling, convergent, floor, nearest, round, simplest, and zero

Fixed-Point Tool in Simulink Fixed Point™ facilitates the conversion of floating-point data types to fixed point. For configuration of fixed-point properties, the tool tracks overflows and maxima and minima.

**Code Generation**

Once you've got advanced your set of rules or communications device, you can robotically generate C code from it for verification, rapid prototyping, and implementation. Most System gadgets, functions, and blocks in Communications System Toolbox can generate ANSI/ISO C code the use of MATLAB Coder™, Simulink Coder™, or Embedded Coder™. A subset of System gadgets and Simulink blocks also can generate HDL code. To leverage present highbrow belongings, you can choose optimizations for specific processor architectures and integrate legacy C code with the generated code.

You can also generate C code for both floating-point and fixed-point data types.

DSP Proto typing DSPs are used in communication system implementation for verification, rapid prototyping, or final hardware implementation. Using the processor-in-the-loop (PIL) simulation capability found in Embedded Coder, you can verify generated source code and compiled code by running your algorithm’s implementation code on a target processor. FPGA Prototyping

FPGAs are used in communication systems for implementing high-speed signal processing algorithms. Using the FPGA-in-the-loop (FIL) capability found in HDL Verifier™, you can test RTL code in real hardware for any existing HDL code, either manually written or automatically generated HDL code.