**Clustering Routing Algorithm for Wireless Sensor Network Based on Mixed Strategy Game Theory**

**ABSTRACT**

We propose a clustering routing algorithm for wireless sensor networks (WSNs) based on mixed strategy game theory (CR-MSGT), which simulates the behavior of sensor nodes in a network through the mixed strategy model, so as to determine whether sensor nodes participate in the election of candidate cluster heads (CHs). The sensor nodes are randomly selected as CHs or common nodes according to their residual energy and the average energy of the network. Games are continuously played between nodes until the revenue function is maximized to reach the game equilibrium, thus proving the existence of the Nash equilibrium. Experimental results show that CR-MSGT can effectively extend the survivability of a network and mitigate the energy consumption of nodes.

**Keywords:** wireless sensor network, mixed strategy game theory, cluster.

**CHAPTER 1**

**INTRODUCTION**

A wireless sensor network (WSN) has sensor nodes, which can perceive a certain range of environmental information, as the basic unit. In recent years, with the rapid adoption of the Internet of Things, the range of applications of WSNs has become increasingly extensive and now includes smart medical care,(1) smart transportation,(2) modern agriculture,(3) and warehouse management.(4,5) For a WSN, the survival status of nodes affects the information perception ability of the entire network and determines the operating life of the network. Sensor nodes are usually driven by a limited amount of power, and their ability to calculate, store, and transmit data is also limited. Because of the large number of sensor nodes in most networks, battery replacement is generally unfeasible, so reducing node energy consumption and extending the network life are important research directions. Cluster routing is an effective technology to solve the above problems, where the core idea is to divide the network into multiple clusters with each cluster having a node called the cluster head (CH). The task of communicating with the base station (BS) is completed by the CH node. The nodes in the network take turns acting as the CH. The CH integrates the information collected by other nodes in the cluster, then forwards the information to the BS via a multi-hop or direct communication mode. The clustering mechanism can reduce the amount of forwarding data and shorten the data transmission distance of most nodes. However, the node acting as the CH consumes more energy than the other nodes in the cluster. Our task is to select the most suitable node in the network to act as the CH through game theory, which can balance the node load and energy. Game theory provides a decision-making environment model that is interdependent and may exchange roles. In this paper, a clustering routing algorithm for a WSN based on mixed strategy game theory (CR-MSGT) is proposed.

The game-theory-based distributed clustering approach (GTDCA) algorithm used to maximize the WSN lifetime establishes a CH game equilibrium model.(10) The nodes in the network are randomly declared as the CH with the equilibrium probability. The equilibrium probability is related to the income, cost, and the total number of network nodes when a node is declared to be the CH. However, the algorithm requires all nodes to participate at the same time, making the number of game participants large and the algorithm inefficient. The optimized clustering WSN algorithm based on game theory is a game-theory-based algorithm that partitions the network and employs a partition rotation mechanism to derive the region equilibrium probability according to the total number of nodes in each region.(11) Each region node randomly declares the CH with the equilibrium probability, but the algorithm requires sensor nodes to be evenly distributed in the network. Lin and Wang proposed a non-cooperative game model, in which sensor nodes declare whether they are CHs by computing the highest probability of maximizing revenue in a mixed strategy.(12) Li and Wu proposed a method combining a non-cooperative game with a distributed clustering algorithm to reduce the energy consumption of a network.(13) This method reduces the number of forwarding packets and extends the network life by collecting energy from the network.

**CHAPTER 2**

**LITERATURE SURVEY**

**P. Chanak and I. Banerjee: IEEE Trans. Consum. Electron. 66 (2020) 22.**

Recently, the Internet of Things (IoT) topology has used to collect physical, physiological, vital signs of patients in consumer-centric e-health or consumer' wellness care services. In such healthcare systems, varieties of medical sensors are attached to the patients to collect vital signs from those who are under observation. The data gathering process in IoT-enabled Wireless Sensor Network (WSN) suffers from the congestion problem. The effect of this translates on missing packets, a decrease of reliability and throughput degradation in IoT-enabled WSN. This article proposes a distributed congestion control algorithm for IoT-enabled WSNs to effectively resolve the congestion for healthcare applications. The proposed scheme alleviates congestion by a priority-based data routing strategy. Furthermore, this article presents a priority queue based scheduling scheme for better reliability. We analyze the properties of the proposed congestion control mechanism mathematically and validate its performance through extensive simulation and real-life experiments. The application of this work can be used to an early warning system in detecting abnormal heart rate, blood pressure, ECG, EMG in the hospital/home care environment to the state-of-art diagnosis.

**Summary:** Studied about Distributed congestion Control algorithm For IOT-enabled WSN .

**2 M. N. M. Bhutta and M. Ahmad: IEEE Access 9 (2021) 65660:**

Food supply chain process comprises crops collection, processing of food, shipping & delivery to the whole seller in the market. Harvested foods decompose from the moment they are harvested due to attacks from enzymes, oxidation, and microorganisms. These include bacteria, mold, yeast, moisture, temperature, and chemical reaction. The spoilage of fresh food has increased over time due to the multistage slow food supply chain process. The identification, traceability, and real-time tracking of goods in supply chains have always been a challenge. The advent of the Internet of Things and cloud computing has brought a new approach to the food supply chain process for better cooperation among supply chain partners. The supply chain management (SCM) benefit greatly through automation based on key technologies of IoT, Radio Frequency Identification (RFID), and Wireless Sensor Networks (WSN). These technologies collect the data relevant to the food supply chain system, such as identifying tag-possessed objects or individuals and sensing capabilities of the surrounding environment. However, the collected data can be tempered or modified by attackers to provide false information about environmental conditions. They can destroy or damage the product due to false identification of dynamic environmental conditions. Furthermore, the current automation systems in industry-based retail logistics and SCM do not provide efficient solutions for monitoring the quality of perishable products with integrated solutions. This research aims to develop a secure monitoring and reporting system based on IoT to update the quality of the perishables along with the SCM with a focus on transportation without any human intervention.

**Summary:** Studied about the Secure Identification, Traceability and Real-Time Tracking of Agricultural Food Supply During Transportation Using Internet of Things.

**3 J. B. Valencia, L. C. Londono, D. M. Viloria, and M. R. Garcia: IEEE Internet Things J. 6 (2018) 3024.**

The rapid growth of Internet-of-Things (IoT) in the current decade has led to the development of a multitude of new access technologies targeted at low-power, wide area networks (LP-WANs). However, this has also created another challenge pertaining to technology selection. This paper reviews the performance of LP-WAN technologies for IoT, including design choices and their implications. We consider Sigfox, LoRaWAN, WavIoT, random phase multiple access (RPMA), narrowband IoT (NB-IoT), as well as LTE-M and assess their performance in terms of signal propagation, coverage and energy conservation. The comparative analyses presented in this paper are based on available data sheets and simulation results. A sensitivity analysis is also conducted to evaluate network performance in response to variations in system design parameters. Results show that each of RPMA, NB-IoT, and LTE-M incurs at least 9 dB additional path loss relative to Sigfox and LoRaWAN. This paper further reveals that with a 10% improvement in receiver sensitivity, NB-IoT 882 MHz and LoRaWAN can increase coverage by up to 398% and 142%, respectively, without adverse effects on the energy requirements. Finally, extreme weather conditions can significantly reduce the active network life of LP-WANs. In particular, the results indicate that operating an IoT device in a temperature of -20 °C can shorten its life by about half; 53% (WavIoT, LoRaWAN, Sigfox, NB-IoT, and RPMA) and 48% in LTE-M compared with environmental temperature of 40 °C.

**Summary:** Studied about Low-Power Wide Area Network Technologies for Internet-of-Things.

**4 S. A. Alavi, K. Mehran, Y. Hao, A. Rahimian, H. Mirsaeedi, and V. Vahidinasab: IEEE Trans. Smart Grid 10(2019)4323:**

This paper presents a complete design, analysis, and performance evaluation of a novel distributed event-triggered control and estimation strategy for dc microgrids. The primary objective of this work is to efficiently stabilize the grid voltage, and to further balance the energy level of the energy storage systems. The locally-installed distributed controllers are utilized to reduce the number of transmitted packets and battery usage of the installed sensors, based on a proposed event-triggered communication scheme. Also, to reduce the network traffic, an optimal observer is employed which utilizes a modified Kalman consensus filter to estimate the state of the dc microgrid via the distributed sensors. Furthermore, in order to effectively provide an intelligent data exchange mechanism for the proposed event-triggered controller, the publish-subscribe communication model is employed to setup a distributed control infrastructure in industrial wireless sensor networks. The performance of the proposed control and estimation strategy is validated via the simulations of a dc microgrid composed of renewable energy sources. The results confirm the appropriateness of the implemented strategy for the optimal utilization of the advanced industrial network architectures in the smart grid.

**Summary:** Studied about a Distributed Event-Triggered Control Strategy for DC Microgrids Based on Publish-Subscribe Model Over Industrial Wireless Sensor Networks.

**5 Q. Li, N. Zhang, M. Cheffena, and X. Shen: IEEE Trans. Wireless Commun. 19 (2020) 696.**

Recent developments in industrial wireless sensor networks (IWSNs) have revolutionized industrial automation systems. However, harsh industrial environment poses great challenges to a time-critical and reliable wireless communication. For instance, effects of multipath fading, noise and co-channel interference can have unpredictable and time-varying impacts on the propagation channel, leading to the failure of on-time packet delivery. To address this problem, in this paper we propose a channel-based Optimal Back-off Delay Control (OBDC) scheme which can minimize the total time a packet spends in the sensor node (TSN) by assessing the features of a generic wireless channel. Specifically, we first explore the channel impairments by investigating the probability density function (PDF) of the level crossing rate (LCR) of the received signal in the industrial wireless environment. Then, with the obtained channel assessment results, we develop a phase-type semi-Markov model to investigate the probability distribution of the back-off delay of a packet in the sensor node (SN). The probability distribution of the back-off delay can be further substituted with TSN according to the queuing theory. The proposed OBDC scheme examines the Kullback-Leibler (KL) divergence between the obtained distribution of TSN and the packet arrival rate, and reduces the TSN according to an objective function which is constantly renewed in every transmission round with regard to a delay constraint. The simulation results show that the OBDC scheme can reduce TSN and guarantee to keep the TSN in an acceptable range even though the wireless channel is impaired by interference effects. It also shows that the OBDC scheme can reduce the proportion of packets meeting their deadline to the total packets in transmission when the number of SN and LCR changes.

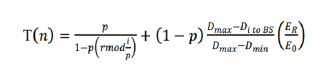
**SUMMARY:** Studied about Channel-Based Optimal Back-Off Delay Control in Delay-Constrained Industrial WSNs.

**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.

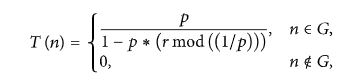
Naveen explored fifteen different types of clustering wireless sensor protocols which considered more in energyefficient and lifetime of the network system.. Prasad simulated LEACH using TDMA routing protocol. Also, they surveyed the previous approaches for selecting CH and improving the WSN performance 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

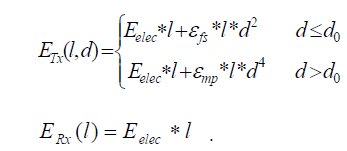
known by exchanging information. We can change the positions of sink node for analyzing the best position so that

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:

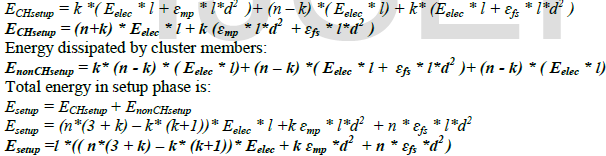


The schedule sent by CHs will be received by all the member nodes consuming the energy as follows:



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.

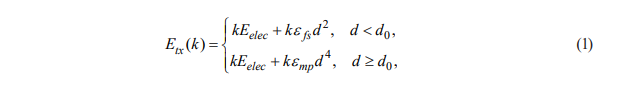
4. We have noticed that the cluster head missions are more than the ordinary nodes, so the cluster head consumes more energy than the others which one of the drawbacks of the LEACH algorithm.

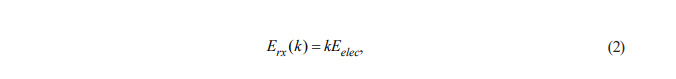
**CHAPTER 4**

**PROPOSED METHOD**

**Network model :**

The topology of the WSN in this study is shown in Fig. 1. The sensor nodes are randomly distributed in the monitoring area and can be divided into CH and common nodes, all of which have unique numbers. The nodes have the same function, do not have mobility, and can calculate the communication distance on the basis of the signal strength. To reduce the amount of data forwarding, the CH node adopts data fusion technology. The BS is usually located outside the monitoring area and is responsible for sending the information collected by the WSN to the enduser of the information. Its energy is not limited and it usually has an unlimited communication capability. A sensor node usually includes four functional modules: sensor, data processing, communication, and energy supply modules.(14) The sensor module is mainly responsible for the perception and collection of data and converts analog signals into digital signals. The data processing module is mainly responsible for data processing, such as data fusion. The communication module oversees information transmission between nodes. The energy supply module is responsible for the energy management of the node. 3.2 Energy consumption model The traditional energy consumption model is used for the sensor node.(15) The free space model is used to calculate the energy consumed by a node in forwarding information when the distance that the node transmits information is less than the distance threshold.(16) Using a multipath fading model,(17) the node calculates the energy consumed by a node in forwarding information when the distance is greater than or equal to the distance threshold. Specifically, when the nodes send and receive one bit of data, the energy consumption is as follows:





where Eelec is the energy consumption of sending or receiving one bit of data, Etx is the energy consumption of transmitting k bits of data, Erx is the energy consumption of transmitting k bits of data, εfs is the power amplification energy consumption coefficient under the free space model, εmp is the power amplification energy consumption coefficient under the multipath fading model, and d0 is the critical distance for selecting the two transmission models, calculated as

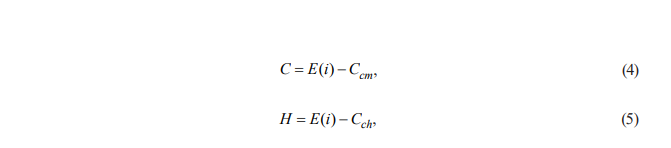


**Mixed Strategy Game Model:**

In a game model, a pure strategy refers to the case that participants can choose only one strategy, whereas in a mixed strategy, participants can choose different strategies with given probabilities. A mixed strategy is the spatial probability distribution of a pure strategy, which is a special case of a mixed strategy. In regions with a communication radius of R and N sensor nodes, the game is represented by G(N, S, U). The actions of nodes are organized in rounds, and in each round, sensor nodes can select policies from the strategy set S = {Ych, Nch}, where Ych represents participation in the candidate CH election and Nch represents nonparticipation in the candidate CH election. 1 2 1 2 ( , ) (0 1, 1) i i i ik i i p pp p p p = ≤≤ += indicates that sensor node i participates in the candidate CH election with probability pi1 and does not participate with probability pi2. The N sensor nodes participate or do not participate in the candidate CH election with probability p = (p1, p2, …, pN) as the mixed strategy of this paper. U represents the network utility, which is formulated as a revenue cost model, and different selection strategies for sensor nodes yield different gains. To maximize the network utility, the sensor node selects strategy Ych as a candidate CH or strategy Nch as a common node. In the clustering algorithm, different nodes choose to become candidate CHs or common nodes with different probabilities, resulting in different revenue functions. To maximize the network utility, a mixed strategy game is formed between all sensor nodes

**Revenue function:**

In the cluster game model, when at least one sensor node j selects strategy Ych in the network, the revenue function of sensor node j is H and revenue functions of the other nodes are C. If no sensor node selects strategy Ych, then the gain of the revenue function is 0. C and H are given by

****

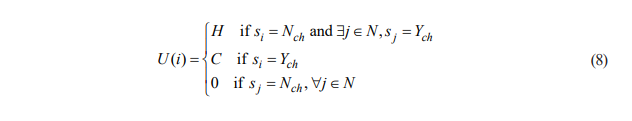
where E(i) is the residual energy of the sensor node, Cch is the cost of the sensor node becoming a CH, and Ccm is the cost of the sensor node becoming a common node. The cost of the sensor node becoming a CH consists of three parts: the energy required to receive the packets from the cluster member, the energy required to integrate the data of the cluster members, and the energy required to transfer the packet to the BS, where Eqs. (1) and (2) are used for the energy calculation. The expression for Cch is

****

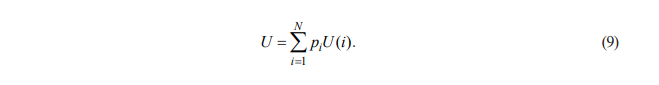
where Eaggr is the energy required to fuse the data of the cluster members. The cost of becoming a sensor node is expressed as

****

According to the above expressions, the revenue function expression U(i) of node i is as follows.

****

Therefore, in the case of two sensor nodes involved in the CH game, the main function is shown in Table 1. Under the definition of a mixed strategy, the revenue function of each sensor node is also random because of the randomness of the strategy. In models having multiple sensor nodes with mixed strategies known to be p = (p1, p2, …, pN), the revenue function of the network is

****

**Cluster Routing Algorithm Based on Mixed Strategy Game :**

In this paper, the algorithm refers to the LEACH protocol,(18) which is performed periodically and consists of three stages: network initialization, cluster establishment, and stable communication. In the cluster establishment stage, the main task is the selection and determination of candidate CHs, and cluster formation.

**Network initialization:**

In the implementation of this algorithm, the network should first be initialized. Assuming that all sensor nodes are randomly distributed in a region, each sensor node can adjust its own transmit power to adjust the communication radius. Upon receiving the broadcast message from the BS, all nodes record the distance from the BS and adjust the optimal transmission power to communicate with the BS. In the first deployment, all sensor nodes broadcast messages within the same communication radius R. The nodes receiving the message determine the neighbor nodes within the communication radius R and are stored in the list of neighbor nodes.

**Candidate CH election**

After the network initialization, the sensor nodes can choose to be the CH or common node of the cluster, and their strategy can be changed in each round. To improve the quality of CHs, in the start key phase, the average energy in the network is calculated from the number of surviving nodes and the residual energy. The amount of residual energy and the average energy for each node are then compared, and only the nodes with residual energies greater than the average energy are included in the candidate CH node set. Each CH node in the set produces a random number between 0 and 1.

**Identification of CH**

If there is only one sensor node selection strategy Ych in the candidate CH stage within a cluster group, the node automatically becomes the true CH. If there are two or more node selection strategies Ych, the present game model is used to reduce the energy consumption difference between CHs. At the same time, the game model balances the node energy consumption so that only one node within a cluster group is elected as the CH while maximizing the utility of all nodes in this game.

**Cluster group formation**

After all the CHs are determined, each CH adjusts the transmit power to send data to the BS and broadcasts the message within the communication radius. When the common sensor nodes receive the message, they select the nearest CH to join the cluster group

**Stable communication**

After the completion of the clusters, the WSN enters the stable data communication stage, in which the main task is to send the collected data to the BS through the CH of each cluster. By scheduling the CH, the members of the cluster send data to the CH in a coordinated manner by time-division multiplexing. After receiving data from all nodes in the cluster, the CH preprocesses and fuses the data and sends it to the BS.

**CHAPTER 5**

**ADVANTAGES AND APPLICATIONS**

**Advantages:**

* This method reduces the number of forwarding packets and extends the network life by collecting energy from the network.
* Easy to implement and easy to understand

**Applications:**

1.industrial control

2.environmental monitoring,

3. military surveillance,

4.intelligent transportation systems and medical field.

5.Furthermore, it can function independently in harsh or high-risk places where human presence is not possible

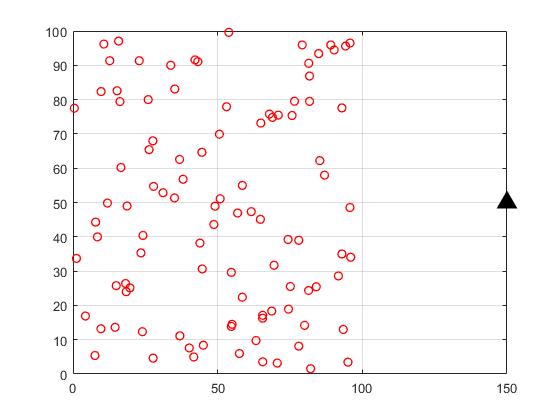
6.Disaster relief operations.

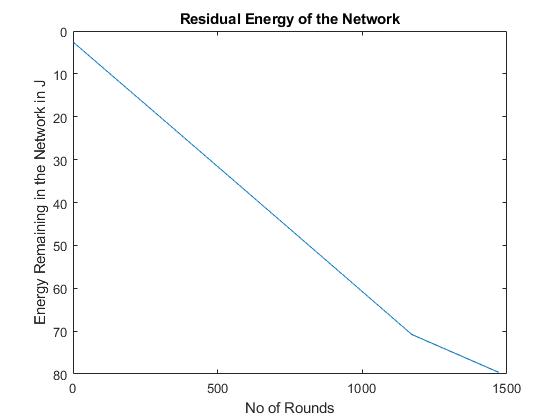
7.Biodiversity mapping

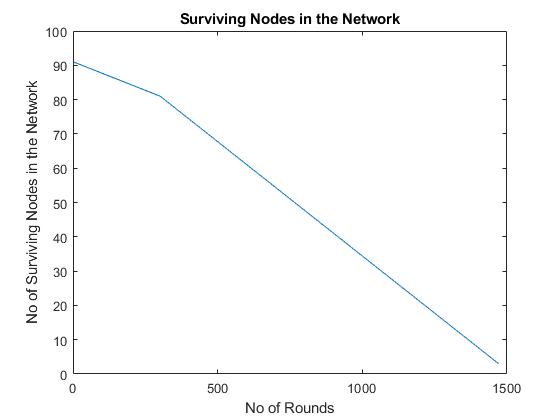
8.monitoring of temperature, pressure, and humidity

**CHAPTER 6**

**RESULTS**







**CHAPTER 7**

**CONCLUSION**

Toward solving the clustering routing problem in WSNs, we propose an algorithm based on CR-MSGT in this paper. All sensor nodes choose whether to become the CH with a random probability, resulting in a mixed strategy game model. In accordance with the mixed strategy game model, the CH node set in the network is determined in order to form clusters and carry out stable communication. Experimental results show that the proposed algorithm can effectively balance the energy consumption of nodes, thereby prolonging the life of the network.

**REFERENCES**

1 P. Chanak and I. Banerjee: IEEE Trans. Consum. Electron. 66 (2020) 223. https://doi.org/10.1109/ TCE.2020.2987433

2 M. N. M. Bhutta and M. Ahmad: IEEE Access 9 (2021) 65660. <https://doi.org/10.1109/ACCESS.2021.3076373>

3 J. B. Valencia, L. C. Londono, D. M. Viloria, and M. R. Garcia: IEEE Internet Things J. 6 (2018) 3024. https:// doi.org/10.1109/JIOT.2018.2878528

4 S. A. Alavi, K. Mehran, Y. Hao, A. Rahimian, H. Mirsaeedi, and V. Vahidinasab: IEEE Trans. Smart Grid 10(2019)4323. <https://doi.org/10.1109/TSG.2018.2856893>

5 Q. Li, N. Zhang, M. Cheffena, and X. Shen: IEEE Trans. Wireless Commun. 19 (2020) 696. https://doi. org/10.1109/TWC.2019.2948156

6 M. A. Rahman, L.-M. Ang, and K. P. Seng: IEEE/ACM Trans. Comput. Biol. Bioinf. 18 (2021) 765. https://doi. org/10.1109/TCBB.2020.2978188

7 A. Verma, S. Kumar, P. R. Gautam, T. Rashid, and A. Kumar: IEEE Sens. J. 20 (2020) 5615. https://doi. org/10.1109/JSEN.2020.2969697

8 J. He, H. Li, H. Tong, Z. Yuan, X. Yang, and W. Huang: IEEE Access 8 (2020) 93380. https://doi.org/10.1109/ ACCESS.2020.2994133

9 T. N. Tran, T. V. Nguyen, K. Shim, and B. An: IEEE Access 8 (2020) 141310. https://doi.org/10.1109/ ACCESS.2020.3013644

10 S. Kassan, J. Gaber, and P. Lorenz: J. Network Comput. Appl. 123 (2018) 80. [https://doi.org/10.1016/j. jnca.2018.09.004](https://doi.org/10.1016/j.%20jnca.2018.09.004)

11 A. C. Voulkidis and P. G. Cottis: IEEE Trans. Wireless Commun. 15 (2016) 4196. https://doi.org/10.1109/ TWC.2016. 2536727

12 D. Lin and Q. Wang: IEEE Access 7 (2019) 49894. <https://doi.org/10.1109/ACCESS.2019.2911190>

13 Y. Li and Y. Wu: IEEE Access 8 (2020) 138382. <https://doi.org/10.1109/ACCESS.2020.3012409>

14 A. A. H. Hassan, W. M. Shah, A. H. H. Habeb, M. F. I. Othman, and M. N. Al-Mhiqani: IEEE Access 8 (2020) 200500. <https://doi.org/10.1109/ACCESS.2020.3035624>

15 J. Kang, J. Kim, M. Kim, and M. Sohn: IEEE Access 8 (2020) 69359. https://doi.org/10.1109/ ACCESS.2020.2986507

16 A. Jaiswal and M. R. Bhatnagar: IEEE Trans. Inf. Theory 65 (2019) 1113. https://doi.org/10.1109/ TIT.2018.2856116

17 Y. Chen, M. Ding, and D. L. Pérez: IEEE Wireless Commun. Lett. 8 (2019) 1419. https: //doi.org/10.1109/ LWC.2019.2919834

18 M. Adnan, L. Yang, T. Ahmad, and Y. Tao: IEEE Access 9 (2021) 38531. https://doi.org/10.1109/ ACCESS.2021.306309

**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.

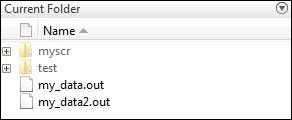
So, I studied Niklaus Wirth’s book Algorithms + Data Structures *=* Programs and learned how to parse programming languages. I wrote the first MATLAB an acronym for Matrix Laboratory in FORTRAN, with matrix as the only data type. The project was a kind of hobby, a new aspect of programming for me to 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:**

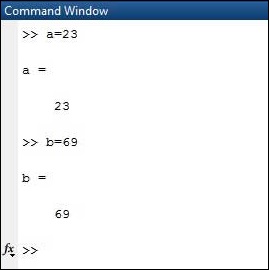
After logging into your account, you can enter MATLAB by double-clicking on the MATLAB shortcut icon (MATLAB 7.0.4) on your Windows desktop. When you start MATLAB, a special window called the MATLAB desktop appears. The desktop is a window that contains other windows. The major tools within or accessible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

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

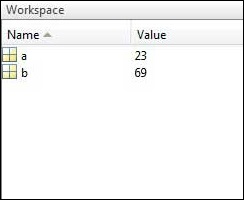
**Current Folder:** This panel allows you to access the project folders and files.



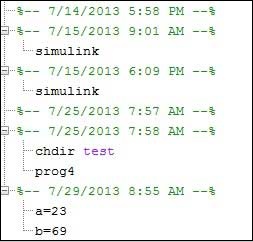
**Command Window:** This is the main area where commands can be entered at the command line. It is indicated by the command prompt (>>).



**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.