Int. J. Nonlinear Anal. Appl. In Press, 1–11
ISSN: 2008-6822 (electronic)
http://dx.doi.org/10.22075/ijnaa.2023.29318.4120

Optimize routing and reduce latency when sending information among Internet of Things (IoT) nodes

Shahram Khoshnavaz[a,*], Mostafa Abbasi Kia[b]

[a]Department of Computer Engineering, Faculty of Technology and Engineering, Lorestan, Iran
[b]Department of Computer Science, University of Lorestan, Khorramabad, Lorestan, Iran

(Communicated by Javad Vahidi)

Abstract

The existing nodes of IoT networks are very small in size, deployed for long periods, and have very limited resources, which means that an IoT network must be very energy efficient to survive for a long time. Therefore, finding optimal routing techniques that lead to better data sharing without wasting energy can lead to more energy savings. This is an optimization problem, which means that we need to use optimization algorithms to find the optimal path in an IoT network. Some of the optimization algorithms are called meta-heuristic algorithms, these algorithms are inspired by nature, such as Artificial Neural Networks (ANN), which are Gradient methods to find the most suitable solution for a given problem. Our next algorithm is Particle Swarm Optimization (PSO). If the search combination of both algorithms is used in parallel, the search power will increase and better answers will be found in less time. For this reason, we suggest using a combination of the above algorithms. This idea is a combination of two optimization algorithms, PSO (Particle Swarm Optimization) and ANN (Neural Network) to optimize routing and reduce latency when sending information between IoT nodes in an IoT system. The proposed protocol is focused on optimizing energy consumption and execution time with the help of the GA-PSO algorithm based on routing-based clustering. Finally, to evaluate the proposed protocol, it was simulated using C++ software and compared with the method presented in the reference article based on the enhanced Ant Colony Algorithm, and the results show the efficiency of the proposed method in terms of energy consumption and execution time. The results show that in the presented algorithm, the execution time has been reduced to almost a quarter of the execution time in the algorithm of the reference article. Also, the results showed that our proposed method consumed 20 kJ less energy.

Keywords: Routing, Latency Reduction, Artificial Neural Network, Hybrid optimization algorithms, IoT
2020 MSC: Primary 68M12, 68W50; Secondary 68T20, 68T07

1 Introduction

The Internet of Things (IoT) is a network of physical devices that work together to provide advanced services. These networks can have different models, depending on the intended user, this network can be a WSN (Wireless Sensor Network), robotic automation, and home and personal automation. The purpose of the Internet of Things is to create devices to share data and achieve desired services, the data to be shared is mostly from the physical

*Corresponding author
Email addresses: khoshnavaz.sh@lu.ac.ir (Shahram Khoshnavaz), abbasikia.m@lu.ac.ir (Mostafa Abbasi Kia)

Received: November 2022 *Accepted:* January 2023

world, not the information world, this is done in the network using sensors. Sensors may have different functions, including recording heat, humidity, light, and weight or measuring wireless signals. These sensor nodes collect data and send them to a sensing (destination) node based on their architecture. A sink can send data to a processing node or a storage node. This protocol depends on how to build and deploy the mesh, in general, IoT network topology is flat, hierarchical, or a hybrid mesh of both, these topologies are not equal in terms of features, which leads to the deployment of a case according to the required application.

The IoT networks existing nodes are very small in size and are deployed for a long time and have very limited resources, which means that an IoT network must have a very efficient energy consumption for long survival. Therefore, finding optimal routing techniques that lead to better data sharing without wasting energy can lead to more energy savings. This is an optimization problem, which means that we need to use optimization algorithms to find the optimal path in an IoT network. Some of the optimization algorithms are called meta-heuristic algorithms, these algorithms are inspired by nature, such as GA (Genetic Algorithm) which uses the theory of biological evolution to find the most suitable solution for a given problem. Another algorithm that has been notably employed in optimization problems in various branches of science is the Particle Swarm Optimization (PSO) algorithm which is inspired by the behavior of birds when they cooperate during flight and food search. But unfortunately, these algorithms are usually stuck in local optima and are not able to find the best solution. However, if the search combination of both algorithms is used in parallel, the search power increases, and better answers are found in less time. For this reason, we suggest using a combination of the above algorithms.

In general, a Wireless Sensor Network (WSN) is a collection of sensor nodes connected by wireless communication channels. Every sensor node is a small device that can collect data from the surrounding area, perform simple calculations, and communicate with other sensors or with the Base Station (BS). Due to this feature, wireless sensor networks have become a very important issue with rapid development, which are vulnerable to a wide range of attacks due to deployment in hostile environments. A WSN is a large network of resource-limited sensor nodes with multiple predefined functions, such as sensing and processing with several low-cost resource-limited sensor nodes, to measure important environmental information and transmit it to a sink node that the agent provides the gateway to another network or the access point for the human interface. These sensor networks are composed of limited energy nodes embedding limited transmission, processing, and measurement capabilities. Therefore, the life cycle of the network is shortened and hence the implementation of an energy-efficient technique becomes an important requirement for WSN [4, 6, 11, 20].

1.1 The need for an energy-efficient schedule

In the practical implementation of a WSN, several levels of sensing coverage and network connectivity are required to provide high-quality data services. In wireless networks with battery-consuming devices, the energy-saving mechanism is extremely important in increasing the lifetime of the network. Energy conservation is important during periods of no activity as well as during incidents. Reducing traffic eavesdropping is critical because the transceiver consumes the same power as the transmission for standby listening. In wireless sensor networks, a sleep-wake duty cycle path is adopted for energy efficiency and energy conservation, because each sensor node is usually equipped with a battery that is power limited. In wireless networks, a cluster is a group of nodes that is generally considered a scalable method for managing large sensor networks, and each cluster contains a single cluster head (CH). Network sensor nodes can be managed locally by the cluster head in a cluster - a node is chosen to coordinate the nodes within the cluster and take responsibility for communication between the cluster and the base station or other cluster heads. Clusters provide a convenient framework for resource management, information synthesis, and local decision-making. Because in a cluster all the nodes will be awake to communicate with the cluster head. This communication occurs regardless of the energy of the nodes in the cluster. During this process, all nodes in the cluster consume energy, regardless of which node can transmit information. Therefore, the sleep/wake scheduling technique is implemented in the network to reduce this energy consumption by nodes that are not able to transmit information or communicate in the cluster with the cluster head. Therefore, energy conservation is possible in the network through this sleep/wake scheduling process. This is possible by dividing the nodes of the cluster into active nodes and inactive nodes. Active nodes are nodes with high energy and able to communicate with the cluster head, and inactive nodes are nodes with low energy unable to communicate with the cluster head. These inactive nodes are sent to sleep mode with low energy. And the remaining nodes remain in the awake state, which can communicate with the cluster head. This method makes it possible to save the energy of the network and also increase the life of the network [2, 3, 9, 12, 14, 18, 19].

1.2 Scheduling advantages and disadvantages

With the sleep/wake schedule, the sensors are densely deployed inside the same cluster, and the connection of the area covered by the active sensors can be guaranteed [12]. If another source has no traffic to send, slot assignment will be interrupted along the flow path from the source to the sink. Therefore, nodes can adjust their sleep schedule according to changing traffic patterns, which is important for energy conservation [9, 10]. For continuous monitoring systems, synchronization-based sleep/wake scheduling schemes are often used because the traffic pattern is periodic [18]. When sleep/wake scheduling is applied to the network node, the energy consumption can be controlled, that is, when a task needs to be done, only some nodes are allocated through the task message [5, 12].

In addition to the mentioned advantages, scheduling also has some issues including the fact that if two neighboring clusters are not densely connected, they cannot merge into a larger cluster. Moreover, a cluster that broadcasts a merge request does not accept a merge request from other clusters, so the cluster must send a message to other clusters, otherwise, it may lead to spending more time [12]. Ideally, in a sensor network, a scheduling protocol must determine a transmission schedule for each packet, otherwise, collisions may occur [9, 10]. Spending time can be more because the transmitted data packet visits all the nodes on the way and waits until the acceptance process is completed [19]. Also, when the sensor nodes are divided into several clusters, only a few sensors from different clusters are selected for active status, if this is repeated and the same nodes are kept inactive as much as possible, then it is possible to lead to problems for the network in the future.

2 Literature review

The paper [16] introduces a method that combines energy efficiency and multiple path selection for data fusion in Wireless Sensor Networks (WSN). The network is divided into clusters, and the cluster heads are chosen based on their remaining energy. The sink calculates multiple paths for each cluster head to transmit data. Data compression is achieved using distributed source programming and lifter scheme wavelet transform in the cluster heads. During each transfer round, the path is changed to conserve energy. However, this method consumes more energy as all the nodes in the network remain active without transmitting anything. This leads to a decrease in the network lifetime. To overcome this problem, the researchers propose a guaranteed distributed sleep/wake schedule scheme, where only the transmission node remains in the active state and the other nodes go to the sleep state.

Nan et al. [12] proposed a distributed sleep/wake schedule scheme with guaranteed coverage. The key objective of this method is to prolong the network lifetime while ensuring comprehensive coverage. To achieve this, the sensor nodes are divided into clusters based on specific sensing coverage criteria. Moreover, a dynamic node selection mechanism is employed, allowing more than one node to remain active within each cluster simultaneously. This approach offers an effective solution for energy conservation in the network.

In their research, Sha Liu et al. [9] contributed to an energy-efficient sleep scheduling protocol called BSMac for sensor networks while maintaining high efficiency and low latency, which is based on a new architecture called BoostNet. in which the base station broadcasts critical planning coordination information using a wide transmission range to reach all sensor nodes in one hop. The main contribution of this paper to the energy-saving planning approach is that it conserves energy during the occurrence of the event and does not require any transmission by the sensors during the period of inactivity, and uses the high in-band transmission power from the base station for optimization. The network parameters are used without the need for a second transmitter and receiver in the sensor nodes.

Yan Wu et al. [19] proposed an optimal sleep/wake schedule algorithm that satisfies a certain message recording probability threshold with minimal energy consumption. In this method, there is an inherent substitution relationship between energy consumption and message delivery performance. The advantage of this article is that the authors have developed an optimization problem whose purpose is to determine the threshold of the probability of recording in each hop so that the lifetime of the network is maximized.

Bo Jiang et al. [5] proposed an energy-aware sleep scheduling algorithm, called SSMTT, to support multi-target tracking sensor networks. SSMTT exploits the wake-up effect of interfering targets to save energy in hyperactive communication. Researchers have provided a solution that includes planning the sleep pattern of sensor nodes. The advantage of this method is that compared to handling multiple targets separately through single target tracking algorithms, it is possible to achieve better energy efficiency.

Chih-Min Chao et al. [1] proposed a quorum-based MAC protocol that enables sensor nodes to sleep longer under light loads. As the traffic flows toward the sink node in wireless sensor networks, a new concept, the next hop group, is also proposed to reduce transmission latency. The advantage of this presented method is that it reduces energy loss because the nodes are kept awake only at a specific time. Also, this method prevails when a method fails to adjust the sleep duration of a sensor node based on its traffic load, as a result, it causes lower energy efficiency or higher latency time. Also, sensors may be deployed in inconsistent environments and thus may fail unexpectedly.

3 Problem identification and proposed solution

This paper proposes a distributed sleep/wake scheduling approach that ensures complete coverage. In this approach, the cluster formation is based on previous work, where an initiator node selects the node with the highest remaining energy as the cluster head by considering the energy information of each node. The energy of the entire cluster is compared using the guaranteed distributed sleep/wake schedule scheme, which assigns a connection value to each cluster for communication with active nodes in other clusters. When the selected cluster head node receives a Joint Request (J_{REQ}) message from member nodes, it responds with a Joint Reply (J_{REP}) message. If the remaining energy of the specific cluster head is above a given threshold, only one node with higher energy remains active while the other nodes enter sleep mode. Additionally, clusters with lower remaining energy become inactive. The cluster head then transmits data to the sink node. The proposed approach's advantage lies in determining whether cluster nodes should be in active or sleep mode based on their remaining energy, thereby increasing the network's lifetime.

3.1 Cluster head selection

In selecting the cluster head, the node with the highest remaining energy is first selected as the cluster head to extend the lifetime of the network. initiator nodes are considered to collect network information from the closest sensor nodes. Assume that the sink node here has information about all the sensor nodes and their position in the network. The cluster head (CH) is determined based on the remaining energy of the nodes specified by the initiator nodes (I). In a network, CH uses more energy than other nodes. But the network performance degrades when the cluster head energy drops. To overcome this situation, the network energy consumption must be balanced by ensuring that the CH continues to change in a cluster depending on its remaining energy. The following steps are the cluster head selection process in the network.

- Initiating node I broadcast a request for energy (E_{REQ}) message with its remaining energy level information (RL_{ini}) to its surrounding nodes.
- The sensor node S_i compares its energy level (R_{Li}) with the initiator.
- If $R_{Li} > RL_{ini}$, then S_i sends an energy response message (E_{REP}). Otherwise, S_i waits for the cluster head advertisement message (CH_{AVD}).
- The initiator node I selects the cluster head with the maximum residual energy, and the initiator node is the node with the second maximum residual energy.
- The initiator node changes whenever the energy level of the node decreases.
- After the CH is selected by the initiator node I, clusters are formed in the network.
- Nodes in the cluster send a CH_{AVD} to the CH, and the CH sends it along with the cluster ID to the sink node.
- A Joint J_{REQ} request is transmitted by the member node along with CH_{AVD}.

The transmission range is minimized because the initiator gathers energy information about the nearest sensors. A node with more energy than the initiator's energy level ensures the minimization of E_{REP} message transmission. When the selected cluster head node receives the J_{REQ} message from the member node, it sends a joint response J_{REP} message to the open node. Then the CH transmits the data to the node.

In Figure 1, initiators I_1, I_2, and I_3 send E_{REQ} energy requests to all surrounding nodes. For example, in this figure, when node I_1 sends E_{REQ} to nodes, the energy consumption of nodes is compared with I_1. Since node 2 has a higher energy level than I_1, and therefore node 2 is chosen as the cluster head, and node 4, which has the next highest energy level, is chosen as the next initiator for choosing the cluster head. Likewise, when node I_2 sends E_{REQ} to nodes, the energy consumption of nodes is compared with I_2. Here, node 9 has the highest energy level and is selected as the cluster head, and node I_1, which has the next highest energy level, is selected as the next initiator for the cluster head. Then, when the I_3 node sends E_{REQ} to the surrounding nodes, the energy consumption of the nodes is compared with I_3. Since node 22 has the highest energy level, it is selected as the cluster head and no other surrounding nodes have the next highest energy level than the initiator I_3. Therefore, node I_3 is selected as the next initiating node.

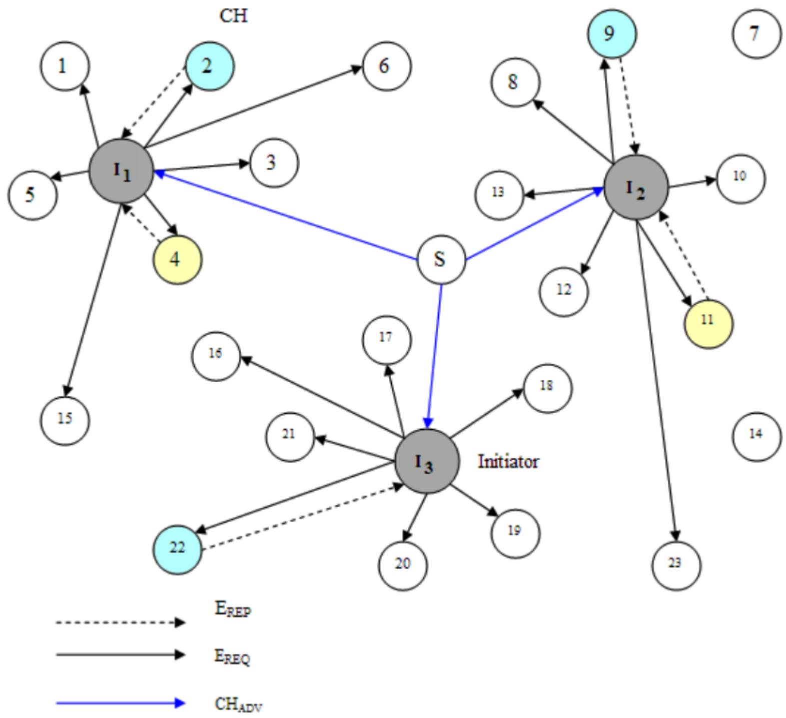

Figure 1: Selecting the cluster head

3.2 Sleep/wake schedule

The sleep/wake schedule is an effective process in which the network energy is saved to the maximum extent. In the network, each cluster starts using the sleep/wake scheduling process as soon as the cluster is formed. only one or two nodes with the highest remaining energy in each cluster are required to be kept active to save energy, while the rest are kept in sleep mode. At the beginning of this planning, all the nodes in the cluster will be active to analyze the remaining energies. This analysis is done to select an active node with the highest remaining energy in a cluster. And this active node will do the measurement work in a cluster. The node that will perform the measurement task is selected by the cluster head. The cluster head sends a task message to instruct the selected node to perform its task as an active node. And also the cluster head sends a sleep message to all the remaining nodes. When a certain task is completed, all nodes in sleep mode send a WORK_REQ to the cluster head to participate in node selection when the next round starts. While WORK_REQs are received from all sleeping nodes in the cluster, the head will execute the active node selection process. Here, an initiator node sends an energy request (E_{REQ}) to all clusters, and the clusters send an energy response (E_{REP}) to the initiator node. Through this process, the initiator node collects the details of the remaining energy of all the clusters in the network. The energy of each cluster is compared with a standard threshold value which is fixed. If the initiator node finds any other cluster with residual energy less than a fixed standard threshold value, that particular cluster is sent to sleep mode until the particular transfer in the network is completed. Through this method, network energy is consumed in a limited way.

Our goal is to optimize the routing of an IoT health network using an optimization algorithm, which is a combination of PSO and GA [17]. Various solutions and methods were presented to reduce energy consumption in these wireless sensor networks. The first and most important thing that can be done to reduce energy consumption and improve other parameters of this network is the clustering of sensors. In fact, instead of all the sensors being directly connected to the server, a representative is chosen from among themselves, and all the information collected by the same node is sent to the server. This method helps the sensors to use less energy to send and receive data according to their geographical distance. The method proposed in this research even increases the reliability of sending and receiving the correct data in the fastest time. In this research, the Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) algorithm were used for clustering.

Using the combination of these two algorithms, we were able to select the cluster heads and calculate the sensors of their subsets in the clusters. In the reference article, the ant colony routing algorithm is used for clustering-based routing, but we used the GA-PSO algorithm. The GA-PSO algorithm will converge to more optimal solutions because it can use the capabilities of both powerful GA and PSO in parallel. In this section, the link model between sensors and main nodes is presented and then based on the link model, the probability of disconnection on the underlying cognitive collaborative communication is analyzed using normal and critical traffic analysis.

3.3 Analysis of the probability of connection and disconnection

In CRBSN, the sensor (SS) and the master node (MN1) must adapt their transmission power below an interference threshold so as not to cause any interference to the master receiver (PR). At the same time, since PS, SS and MN1 share the same channel, and the transmission power of PS also provides the interface to the WBSN network. Therefore, the minimum transmission power of secondary SS and MN1 can be expressed as follows:

$$P_{ss} = \frac{I_{s,x}}{h_{s,x}} \tag{3.1}$$

$$P_{mn1,x} = \frac{I_{mn1,x}}{H_{mn1,x}} \tag{3.2}$$

where the maximum transmission power of P_{ss} and P_{mn1}, is where and, respectively. And the interference threshold is from ss to pr and the interference threshold is from ps to mn1, respectively. As a result, the signal-to-noise ratio in mn2 can be expressed as follows:

$$\gamma_{s,mn2} = \frac{P_{ss}h_{s,mn2}}{P_N + (P_{ps}h_{p,mn2})} \tag{3.3}$$

where the ps transmission power and noise power are vwhere and, respectively. Similarly, the signal-to-noise ratio at mn1 due to transmission from SS and the signal-to-noise ratio at mn2 due to transmission from mn1 are expressed as:

$$\gamma_{s,mn2} = \frac{P_{ss}h_{s,mn1}}{P_N + (P_{ps}h_{p,mn1})} \tag{3.4}$$

$$\gamma_{mn1,mn2} = \frac{P_{mn1}h_{mn1,mn2}}{P_N + (P_{ps}h_{p,mn2})} \tag{3.5}$$

The probability of disconnection from a node to another node can be expressed as follows:

$$P^{out} = P_r(R_{i,j} \leq R_o) = P_r(\log_2(1+\gamma_{i,j}) < R_o) = P_r(h_{i,j} < \frac{2^{R_o}-1}{\widehat{\gamma_{i,j}}}) = 1 - \exp\left(-\frac{2^{R_o}-1}{\widehat{\gamma_{i,j}}}\right) \tag{3.6}$$

Therefore, the channel gain is an exponentially distributed random variable with mean value, where the distance between nodes and, and is the power of the path loss. The probability of incremental cooperative disconnection of the two main node-board architectures can be expressed as follows:

$$P^{out}_{coop} = P^{out}_{s,mn2}P^{out}_{s,mn1} + (1 - P^{out}_{s,mn2})P^{out}_{s,mn1}P^{out}_{mn1,mn2} \tag{3.7}$$

$$\begin{aligned} P^{out}_{coop} = &\left(1 - \exp\left(-\frac{2^{R_o}-1}{\widehat{\gamma_{s,mn2}}}\right)\right)\left(1 - \exp\left(-\frac{2^{R_o}-1}{\widehat{\gamma_{s,mn1}}}\right)\right) + \\ &\exp\left(-\frac{2^{R_o}-1}{\widehat{\gamma_{s,mn2}}}\right)\left(1 - \exp\left(-\frac{2^{R_o}-1}{\widehat{\gamma_{s,mn1}}}\right)\right)\left(1 - \exp\left(-\frac{2^{R_o}-1}{\widehat{\gamma_{mn1,mn2}}}\right)\right) \end{aligned} \tag{3.8}$$

Inserting (3.4), (3.5), and (3.6) into (3.8), we obtain the cutoff probability under the interference constraint.

3.4 Proposed method

In this research, the Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) algorithm were used for clustering. Using the combination of these two algorithms, we were able to select clusters and calculate the sensors in the subset of those clusters.

3.4.1 Basic settings of network and nodes

In the first step, the settings related to the network are made, which are related to how the nodes are wirelessly connected, the total number of rounds that the wireless sensor network is supposed to travel, and other things. In the next step, the initial value of the settings related to the nodes, includes the initial energy, the coordinates of the nodes, and other items. As seen in the diagram, if the condition of the problem is met, the network enters the circuit and starts working.

3.4.2 Determine cluster head and clustering of nodes

The first step is to select the cluster head and then clustering, which is done using the GA in combination with the PSO algorithm. Then, we will explain more about the combination of these two algorithms. In this research, the operation related to the determination of cluster heads is done by GA and the best nodes with the ability to be cluster heads are done by this algorithm. In the next step, it is the turn of clustering, in this step, we use the PSO algorithm for the clustering of nodes.

3.4.3 GA-PSO algorithm for clustering

In this paper, a clustering algorithm based on routing is used to optimize the IoT network based on the GA-PSO algorithm. The objective function in this algorithm is defined according to the position of the nodes and their energy so that the nodes that are in the center of other nodes are introduced as the cluster head. In this way, according to the reduction of the total distances of the nodes from the cluster centers, the network energy consumption is reduced followed by increased speed of routing and information transmission in the network. Below is the algorithm steps description.

A) Initial values

At first, we determine all the initial values required for the GA-PSO method:
nVar: The number of variables we have considered here is 10.
VarSize: The size matrix of the variables.
VarMini - VarMax: Specifies the allowed range of variables.
Maxit: It is the number of repetitions of the algorithm operation loop.
nPop: It is the initial population number.
Wdamp: to reduce the speed in the next steps, we multiply ....by this value after each iteration of the loop.
C1: Personal training coefficient of PSO
C2: PSO general training coefficient
Pc: Crossover percentage in GA
Nc: Number of parents in GA
Gamma: Additional factor in GA Crossover operation
Pm: Mutation percentage in GA
Nm: Number of the population that mutates in GA
Mu: Mutation coefficient in GA
Beta: Additional factor in GA mutation operation

B) Determine the structure of the chromosomes

Each chromosome contains the location and energy of each node. Clustering is done by the GA-PSO algorithm according to the position and energy of nodes. By choosing the correct chromosomes, a cluster with a smaller distance from other nodes and also more energy is selected.

C) GA initial population

The initial population consists of sets of chromosomes, that is, a set of high-powered solutions. Each line represents an individual in GA or a particle in PSO, and the initial velocity of the particles is considered zero. In the GA-PSO algorithm, the cost function is determined for each individual and particle, and finally, the best ones are selected.

D) perform operations on the initial population

Selection: At this stage, some pairs are selected as parents from the initial population by the roulette wheel method. This algorithm is a selection method in which the element that has a lower cost is selected. We attribute a cumulative probability to the cost ratio for each element, and it is with this probability that the chance of selecting each element is determined.

Crossover: In this step, a percentage of the chromosome is separated from each parent and two chromosomes or a new individual is created by the act of transplantation.

Mutation: At this stage, several primary individuals are selected and a mutant individual is created by changing the chromosome of that individual according to the formula.

E) Creation of new PSO particles

Now, by comparing the individuals of the GA population produced in the previous steps, individuals with lower costs replace the particles with higher costs in PS0 and new particles of PSO are created.

F) Performing operations on new particles

In this step, the new velocity is determined for all particles. Then, by applying the velocity limit, the velocities that are outside the range are brought into the allowed range so that convergence takes place, then the new location of the particles is determined by the following relation: New location = previous location + velocity

G) Evaluating and calculating fitness

Finally, the objective function, which is based on the time and energy of the nodes, is calculated for all people in the population. People with the best fitness are selected as optimal answers.

3.4.4 Data transmission and energy calculation

After the clustering operation, it is time to send data and calculate the energy used to send and receive data by nodes. In the end, if the time of using the network or the distance used from the network is more than the predetermined value, and also if the energy of all the nodes ends, the wireless sensor network ends its work. In this section, the aim is to compare the original article and the proposed solution for the Internet of Things system. Various parameters help us to make an accurate comparison between different solutions. For example, energy is one of the investigated parameters. We expect that the system designed with the solution presented in this research can consume less energy and naturally have a longer life. The second parameter that is examined is the information transit time. This criterion evaluates the clustering of sensors. In this section, different strategies were compared based on the parameters mentioned. The algorithm presented in this research is a combination of two meta-heuristic algorithms widely used in different problems. The use of these algorithms in this research, according to the evaluations, showed that it is much more useful and optimal than the solution presented in the original article. One of the reasons for reducing energy consumption in LoT systems is the spatial distance in sending data from the client side to the server. In this proposed algorithm, by evaluating the best spatial distances of the clusters concerning the cluster heads and sending the data of a set by a node which is the cluster head, it causes a drop in energy consumption.

4 Evaluation

4.1 First assessment

In this section, we will evaluate the data transfer time. This measure determines how fast the data can move from the sender to the receiver and at what time it reaches the destination. One of the most important things in the evaluation of this parameter is that shorter distances increase the speed of receiving data, so using the cluster method, in addition to reducing the energy consumption of sensors, increases the speed of information transmission. The type of clustering method also has a significant effect on increasing the speed of information transfer. In addition to the fact that shorter distances increase information transmission, it also causes the sensor to consume less energy to send to the sensors, and actually, the energy consumption of the system also decreases with the improvement of this parameter. There is a very big difference between the solution proposed in this research and the solution of the original article. In this diagram, the vertical axis represents the amount of energy per unit of time. The horizontal axis shows the algorithm used in [14]. In figure 2, the proposed method, GAPSO, has been compared with the method of the reference article, the Ant Colony Algorithm (ACO and enhanced ACO). Practically, this difference between the solution of the original article and the proposed solution in the long-term use of the system causes a great difference in the stability and reliability of the solution between these two.

4.2 Second assessment

The second parameter that is evaluated is energy consumption. Energy is one of the most important parts of the Internet of Things systems. Because many of these tools have temporary and limited energy sources for their situation of use, therefore, it is very useful to use a solution capable of reducing energy consumption to some extent. In this section, the solution proposed in this research has been compared with the solution presented in the reference article along with the basic solution proposed in the reference article. This comparison shows that the solution proposed in this article has been able to save energy consumption to a great extent and provide a system with higher reliability. According to the figure, the energy consumption by the solution proposed in this research is very different from

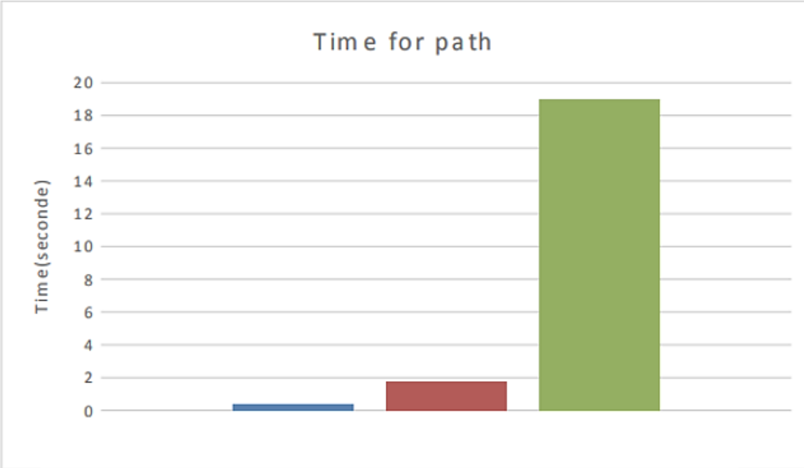

Figure 2: Comparison between the proposed method and the solution of the original article

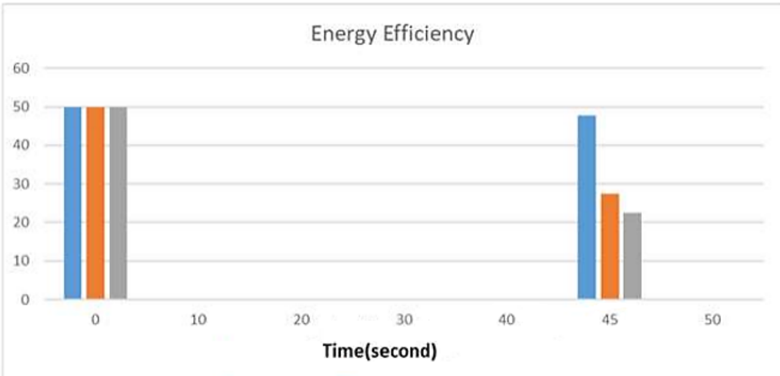

Figure 3: Energy consumption comparison diagram

the energy consumption by the solution proposed in the reference article, and practically after the same period, the energy consumption in the solution presented in the reference article is more. The comparison diagram between the algorithms has been displayed, in this diagram, three algorithms which include the proposed algorithm of this research, the proposed algorithm of the reference article, and another algorithm that was proposed in the reference article are compared in two times zero seconds and 45 seconds in terms of consumption energy. In this diagram, the vertical axis shows the amount of energy in joules. The horizontal axis represents time. In figure 3, the proposed method i.e. GAPSO is compared with the method of the reference article i.e. the Ant Colony Algorithm (ACO and enhanced ACO). As can be seen in this diagram, the energy of the network at the beginning of the simulation is considered the same for all methods but over time for 45 seconds, the energy of the network has decreased to 28 kJ in the method of the reference article, while our proposed method has reached 48 kJ with less energy consumption. As a result, our proposed method has 20 kJ of energy consumed less.

4.3 Third assessment

The third parameter that is evaluated is latency. Latency is a parameter that calculates the time it takes for data to travel back and forth in the network. One of the advantages of the proposed algorithm of this research is the very short latency. The lower the latency, the closer the nodes are to the cluster heads, and this shortening of the path causes less energy consumption, and overall our system will be more stable. A comparison has been made between the amount of latency in the three algorithms of the reference article, the second algorithm proposed in the reference article, and the proposed algorithm of this research. The horizontal axis is in terms of time in milliseconds units, and the vertical axis is in kilobytes per second.

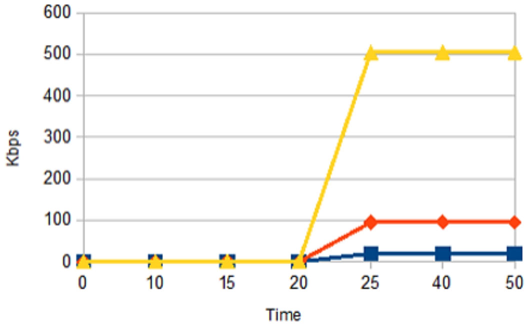

Figure 4: Comparison of latency among nodes

5 Conclusion

The Internet of Things (IoT) and its impact on information sharing have been a big advancement for the Internet. The development and increasing use of the Internet in health sectors requires more optimal performance in the speed of information sharing and dissemination and energy consumption. routing optimization is an important aspect of the Internet of Things. We introduce a new idea that is a combination of two optimization algorithms (PSO (Particle Swarm Optimization) and GA (Genetic Algorithm) and is used to optimize routing and reduce latency when sending information between IoT nodes in an IoT system. Results indicate that the proposed protocol consumes less energy with less execution time. Finally, to the proposed protocol was simulated using MATLAB software to evaluate and compare it with the method presented in the reference article based on the enhanced Ant Colony Algorithm, and the results show the efficiency of the proposed method in terms of energy consumption and time. The results show that in the presented algorithm, the execution time has been reduced to almost a quarter of the execution time in the algorithm of the reference article. Also, the results showed that our proposed method consumed 20 kJ less energy.

References

[1] Ch. Chao and Y. Lee, *Quorum-based energy saving MAC protocol design for wireless sensor networks*, IEEE/IPIP Int. Conf. Embedded and Ubiquitous Comput., (Shanghai, China), vol. 1, IEEE, 2008, pp. 316–322.

[2] S. Chauhan and L.K. Awasthi, *Cluster based task scheduling in wireless sensor network*, Int. J. Comput. Appl. **33** (2011), no. 4, 38–43.

[3] F. De Pellegrini, K. Gomez, D. Miorandi, and I. Chlamtac, *Distributed wake-up scheduling for energy saving in wireless networks*, CoRR **1** (2011).

[4] O. Gaddour, A. Koubˆaa, and M. Abid, *SeGCom: A secure group communication mechanism in cluster-tree wireless sensor networks*, First Int. Conf. Commun. Network. (Hammamet, Tunisia), 2009, pp. 1–7.

[5] B. Jiang, B. Ravindran, and H. Cho, *Energy efficient sleep scheduling in sensor networks for multiple target tracking*, S.E. Nikoletseas, B.S. Chlebus, D.B. Johnson, B. Krishnamachari, (eds) Distributed Computing in Sensor Systems, DCOSS 2008, Lecture Notes in Computer Science, vol 5067, Springer, Berlin, Heidelberg, 2008, pp. 498–509.

[6] T. Kavitha and D. Sridharan, *Security vulnerabilities in wireless sensor networks: A survey*, J. Inf. Assur. Secur. **5** (2009), 31–44.

[7] M.N. Khan, H. Ur Rahman, M.Z. Khan, *An energy efficient adaptive scheduling scheme (EASS) for mesh grid wireless sensor networks*, J. Parall. Distrib. Comput. **146** (2020), 139—157.

[8] S. Kumar and S Chauhan, *A survey on scheduling algorithms for wireless sensor networks*, Int. J. Comput. Appl. **20** (2011), no. 5, 7–13.

[9] Sh. Liu, K. Fan, and P. Sinha, *Dynamic sleep scheduling using online experimentation for wireless sensor networks*, Proc. SenMetrics, San Diego, 2005.

[10] S.V. Manisekaran and R. Venkatesan, *Power efficient scheduling technique for multiple sinks in wireless sensor networks*, Eur. J. Sci. Res. **89** (2012), no. 2, 191–202.

[11] T. Nagamalar and T.R. Rangaswamy, *Sleeping cluster based medium access control layer routing protocol for wireless sensor networks*, J. Comput. Sci. **8** (2012), no. 8, 1294–1303.

[12] G. Nan, G. Shi, Z. Mao, and M. Li, *CDSWS: Coverage-guaranteed distributed sleep/wake scheduling for wireless sensor networks*, EURASIP J. Wireless Commun. Network. **2012** (2012).

[13] Ns2 Network Simulator, `http://www.isi.edu/nsnam/ns`.

[14] P. Priyanka and J.P. Kaur, *Ant colony optimization based routing in IoT for healthcare services*, Second Int. Conf. Intell. Comput. Control Syst. (Madurai, India), 2018, pp. 1155–1159.

[15] M.R. Ramli, J. Lee, and D. Kim, *Hybrid MAC protocol for UAV-assisted data gathering in a wireless sensor network*, Internet Things **14** (2021), 100088.

[16] V. Sasikala, and C. Chandrasekar, *Energy efficient multipath data fusion technique for wireless sensor networks*, ACEEE Int. J. Network Secur. **3** (2012), no. 2, 34–41.

[17] A. Utkarsh, A.S. Kantha, J. Praveen, and J.R. Kumar, *Hybrid GA-PSO trained functional link artificial neural network based channel equalizer*, 2nd Int. Conf. Signal Process. Integrated Networks (SPIN) (Noida, India), 2015, pp. 285–290.

[18] Y. Wu, S. Fahmy, and N.B. Shroff, *Optimal sleep/wake scheduling for time-synchronized sensor networks with QoS guarantees*, IEEE/ACM Trans. Network. **17** (2006), 1508–1521.

[19] Y. Wu, S. Fahmy and N. B. Shroff, *Sleep/wake scheduling for multi-hop sensor networks: Non-convexity and approximation algorithm*, Ad Hoc Networks **8** (2010), no. 7, 681–693.

[20] A.S. Zahmati, B. Abolhassani, A. Beheshti Shirazi, and A.Sh. Bakhtiari, *An energy efficient protocol with static clustering for wireless sensor networks*, Int. J. Electron. Commun. Engin. **1** (2007), no. 4, 69–72.