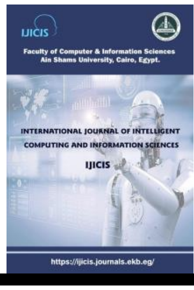
IJICIS, Vol.23, No.1, 53-68
DOI: 10.21608/ijicis.2023.175550.1228

International Journal of Intelligent Computing and Information Sciences

https://ijicis.journals.ekb.eg/



TUNICATE SWARM BASED CLUSTERING AND ROUTING ALGORITHM FOR INTERNET OF THINGS

Aya Saad
Computer Science Department,
Faculty of Computer and Information Sciences, Ain Shams University,
Cairo, Egypt
aya.saad@cis.asu.edu.eg

Islam Hegazy
Computer Science Department,
Faculty of Computer and Information Sciences, Ain Shams University,
Cairo, Egypt
islheg@cis.asu.edu.eg

El Sayed M. El-Horbaty
Computer Science Department,
Faculty of Computer and Information Sciences, Ain Shams University,
Cairo, Egypt
shorbaty@cis.asu.edu.eg

Received 2022-11-18; Revised 2022-12-20; Accepted 2023-02-13

Abstract: *Wireless Sensor Networks (WSNs) are an essential part of the Internet of Things (IoT). Indeed, the usage of efficient routing algorithms makes IoT applications work better. Since sensors are connected to limited sources of energy, some sensor nodes lose energy in a short time. This can affect the network's lifetime. This paper proposes a routing algorithm that works on extending the network lifetime. The proposed algorithm uses Tunicate Swarm Algorithm (TSA), which is a new bio-inspired algorithm. TSA-based clustering is used to select the best cluster heads. Many parameters are considered while selecting the optimal cluster heads such as distance and energy parameters. TSA-based routing is used to create efficient paths from the cluster head to the base station. The path length and the number of hops in the path are considered during creating the paths. The proposed algorithm is compared with three of the most used metaheuristic-based routing algorithms like Particle Swarm Optimization (PSO), Grey Wolf Optimization (GWO), and Ant Colony Optimization (ACO). The comparison evaluates the performance of the TSA-based routing algorithm. TSA-based clustering is used with all the algorithms that are compared. The comparison proves that the proposed algorithm extends the lifetime of the network more than the other algorithms. The time before half of the nodes were dead was extended to be 3.17% more than PSO and GWO, and 1.36% more than ACO.*

Keywords: *Tunicate Swarm Algorithm, Routing, Wireless Sensor Networks, Internet of Things, Network lifetime.*

1. Introduction

IoT is considered one of the hottest topics that attract researchers [1]. IoT applications are used in many fields such as agriculture, military, healthcare, and education [2]. In the IoT paradigm, different wireless sensings objects like humans, animals, buildings, and devices are integrated with the internet. These objects generate and transmit the data via the internet to its destination [3]. This data is used in decisionmaking, which is related to temperature, transportation, healthcare, etc. … [4]. WSN is an indispensable part of IoT. WSN is a set of sensors that are responsible for collecting and transmitting the data to the base station [5,6]. Sensor nodes have many limitations. One of these limitations is that the nodes are powered by a limited source of energy such as batteries [7]. Therefore, transmitting the data to its destination is considered a critical issue. Hence, choosing the wrong path to transmit the data affects the lifetime of the sensor nodes, which could lead to consuming more energy. Moreover, the sensors will run out of energy faster. This will negatively affect the network's lifetime. Furthermore, it will lead to missing important data. Thus, transmitting the data to its destination is a big challenge. Therefore, many researchers are proposing intelligent solutions to solve this issue.

The usage of efficient routing algorithms can solve this issue. That's why researchers are working on proposing efficient routing algorithms to conserve energy and extend the lifetime of the network. The clustering technique is one of the most used techniques to save the energy of nodes [8], where the sensor nodes are split into clusters and every cluster has its cluster head. Each cluster head is responsible for receiving the data from cluster members and sending it to the base station [9]. The clustering technique decreases the number of nodes that need to communicate with the base station. Moreover, it decreases the quantity of data that needs to be transmitted to the base station because the cluster head works on removing the redundant data. Furthermore, multi-hop routing is a popular technique to save the nodes' energy. In this technique, the data is transmitted to the base station through intermediate or relay nodes. Since clustering and multi-hop routing are considered Non-deterministic Polynomial (NP) Hard Problems [10], the researchers start to exploit bio-inspired algorithms to solve them. Bio-inspired algorithms are computational algorithms that mimic the biological and natural behavior of biological organisms such as insects, lions, birds, and bacteria [11,12]. These types of algorithms have many advantages such as speed, adaptation, scalability, and parallelism which help to solve complex and nonlinear problems [12]. TSA is a bio-inspired algorithm that was introduced to optimize complex problems. The algorithm mimics tunicates' behavior while searching for food.

This paper proposes an efficient routing algorithm to extend the network lifetime. In the proposed algorithm, TSA is used twice. First, TSA is used to select the optimal cluster heads where the TSA-based clustering algorithm selects sensors that reduce energy consumption. Furthermore, it selects the sensors which have the shortest distance to the base station and cluster members. Second, TSA-based routing is used to create the optimal paths between cluster heads and the base station, where the path should be short with few hops.

The structure of the paper is as follows: Section 2 discusses the related works. A detailed illustration of TSA is given in section 3. Section 4 explains the network setup and energy model. The proposed algorithm is described in detail in section 5. The experiment and the results are discussed in section 6. Section 7 exposes the conclusion and future works.

2. Related Works

Many Routing algorithms are proposed to solve energy consumption issues and prolong the network lifetime. Some of those algorithms use bio-inspired algorithms such as Particle Swarm Optimization algorithm (PSO), Ant Colony Optimization (ACO), Genetic Algorithm (GA), and Grey Wolf Optimization algorithm (GWO) because of their optimization abilities.

Wang et al. [13] proposed a hybrid routing algorithm for IoT sensor-based applications. In the proposed algorithm, they used PSO for selecting the optimal cluster heads. They used ACO for creating paths from cluster members to their cluster heads and from the cluster heads to the base station. The fitness function used in the PSO algorithm is affected by nodes' residual energy, communication distance in the intra-cluster, and distribution of cluster heads in the network. Assigning the cluster member to a cluster was due to the residual energy of each cluster head and the distance between the node and its cluster head. ACO-based routing algorithm considered the distance between nodes and the energy to create the shortest and most efficient paths. They compared the proposed hybrid routing algorithm against the PSO algorithm and ACO algorithm. The proposed hybrid routing algorithm shows good results against the other algorithms. However, the proposed algorithm has many drawbacks such as the constant number of created clusters. Furthermore, they selected the cluster heads randomly in the first round. The huge number of sensors will raise the time complexity of the algorithm.

Anand & Pandey [14] proposed the GA-PSO clustering and routing algorithm. The algorithm aims to conserve energy and extend the lifetime of the network in IoT sensor-based applications. To select the optimal cluster heads and to construct the clusters, GA is used. The cluster head is selected based on the distance to the base station, the distance from cluster members to the cluster heads, the distance between all nodes to the base station, and the total energy to send the gathered data to the base station. The path selection process is done using PSO to transmit the collected data from the cluster heads to the base station. The factors considered for calculating the fitness of every particle are the distance between a relay node and the candidate's next hops, and the number of hops in the route. The algorithm performs well in terms of the number of alive nodes per round, consumed energy per round, and the number of packets received by the base station per round. However, the disadvantages of this algorithm are that the PSO algorithm may fall into the problem of local optima. Moreover, the algorithm needs to be tested under a heterogeneous network.

Rezaeipanah et al. [15] proposed an efficient re-clustering-based multipath routing algorithm for WSNs. For the clustering phase, they used the K-mean algorithm and Open-Source Development Algorithm OSDMA. OSDMA is one of the meta-heuristic algorithms. Moreover, they used the “Low Energy Adaptive Clustering Hierarchy” (LEACH) algorithm to elect the cluster head for each cluster. In the routing phase, they used GA for multi-path routing in inter-cluster and inter-cluster. GA is considered one of the most used bio-inspired algorithms to solve WSN issues such as energy consumption issues. The algorithm proved superior in the number of dead nodes, total residual energy, residual energy variance, average residual energy, the number of received packets, and network lifetime. The drawbacks of the algorithm are that they didn’t test their algorithm in a heterogeneous network, the hotspot issue is not solved completely, and the data duplication issue is not solved.

Maheshwari et al. [16] worked on reducing energy consumption during data transmission by proposing an efficient cluster-based routing algorithm. The selection of the best cluster head is done using Butterfly Optimization Algorithm (BOA). The BOA is a meta-heuristic algorithm that mimics the food search and mating behavior of butterflies. The BOA uses five factors to select the cluster heads. These factors are the distance between nodes, the degree of nodes, remaining energy, the centrality of nodes, and the distance from nodes to the base station. The paths between cluster heads and the base station are created by ACO. Three factors affect the creation of the paths. The factors are node degree, the distance between nodes, and remaining energy. Moreover, they use a threshold to reselect the cluster heads and construct the paths. The result of the simulation proved the superiority of the algorithm in terms of stability period, network lifetime, energy conservation, and throughput.

Sefati et al. [17] proposed a routing algorithm using black hole optimization and the ACO algorithms. In the proposed scheme they used the black hole algorithm for node deployment and selecting the optimal cluster heads. The objective function of the black hole algorithm depended on residual energy, the distance between nodes, the free buffer of nodes, and the center of the domain. After selecting the best cluster head, the path selection phase starts. In this phase, they used the ACO algorithm for constructing the paths from the cluster heads to the base station. The data was sent to the closest neighbor. Many scenarios are applied to test the algorithms. The algorithm is compared with ACO, Cuckoo Search (CS) algorithm, and Firefly Algorithm (FA). The results show the superiority of the algorithm in the number of transmitted packets to the base station, network lifetime, and average remaining energy. The drawback of the algorithm is that only the distance parameter was considered while creating the path using the ACO algorithm. Moreover, the black hole algorithm may get trapped in the local optima problem.

Ahmadi et al. [18] presented an improved GWO-based clustering and routing algorithm. The enhanced version of GWO can overcome the problem of the local optima in the traditional GWO. This enhancement aims to balance the exploration and exploitation processes of the traditional GWO. The improved GWO (IGWO) is used for selecting the optimal cluster head. Two factors are considered while selecting the cluster head. These factors are the distance between the cluster member to its cluster head and the remaining energy of the cluster head. The IGWO is used to create paths from the cluster heads to the base station. Three factors are considered while creating the paths. These factors are the length of the path, the remaining energy of the cluster heads in the path, and the total number of cluster members for all the cluster heads in the path. The algorithm was compared with PSO, GWO, and LEACH. The algorithm outperforms the other algorithms in terms of network lifetime and residual energy.

R. Dogan et.al [19] addressed the problem of energy efficiency of heterogeneous sensor nodes by proposing Tunicate Swarm Algorithm-based Optimized Routing Mechanism (TORM) for IoT. Sensor nodes are divided into three levels. Level 1 contains nodes with lower energy values, while level 2 contains nodes with a medium amount of energy. Level 3 contains nodes with higher energy values. TSA is used to select the optimal cluster head by considering five objectives. These objectives are the distances between cluster heads to the base station, the distances between cluster members to their cluster head, the initial energy level of nodes, the residual energy of the cluster head, and the number of neighbors of the cluster heads. The selected cluster heads receive data from their cluster members, and then transmit the data to the base station directly. The result of the simulation proved the superiority of the algorithm in terms of stability period, network lifetime, throughput, count of dead nodes versus rounds, and energy conservation.

A. D. Gupta & R. K. Rout [20] proposed energy-efficient cluster-based routing algorithms. The proposed algorithm is a hybrid of two meta-heuristic algorithms, namely, the Remora Optimization Algorithm (ROA) and TSA. The hybrid algorithm is used to select the optimal cluster heads. Moreover, they used four energy harvesting (EH) - enabled nodes to receive the collected data from cluster heads and send it to the base station in a multi-hop fashion. The cluster heads collect the data from cluster members and forward it to an EH-enabled sensor node (relay node), which in turn forwards it to another relay node located on the base station route. Five objectives are considered while selecting the optimal cluster head using the hybrid algorithm. These objectives are the distance between the cluster head nodes and the base station, node density, remaining energy of the cluster head nodes, energy consumption rate, and average delay in transmission. The algorithm has proven superior in terms of stability period, remaining energy per round, number of cluster heads per round, and throughput.

3. Tunicate Swarm Algorithm TSA

Tunicate Swarm Algorithm (TSA) is a new bio-inspired algorithm proposed recently by [21]. The algorithm mimics swarm behaviors and jet propulsion of tunicates during the exploration and exploitation operations to find the location of food sources in the sea. The shape of the tunicate is cylindered. Each tunicate pulls water and makes jet propulsion for moving in water [21]. This process helps tunicate while searching for food.

In the mathematical model of TSA, $\vec{A}$ vector is used to avoid conflicts between tunicates by using it while updating the position of tunicates. It is calculated by Eq. (1). The gravity force $\vec{G}$ is calculated by Eq. (2). $\vec{F}$ is the water flow vector which is calculated by Eq. (3), where c_1, c_2, and c_3 are random values between 0 and 1. $\vec{M}$ represents the interaction and behavior between search agents which is calculated by Eq. (4), where P_{max} and P_{min} are velocities used to represent the interaction between search agents.

$$\vec{A} = \frac{\vec{G}}{\vec{M}} \tag{1}$$

$$\vec{G} = c_2 + c_3 - \vec{F} \tag{2}$$

$$\vec{F} = 2 \,.\, c_1 \tag{3}$$

$$\vec{M} = \lfloor P_{min} + c_1 . P_{max} - P_{min} \rfloor \tag{4}$$

The movement of each search agent toward the best neighbor is calculated by Eq. (5), $\overrightarrow{PD}$ represents the distance between the search agent and the source food, $\overrightarrow{FS}$ is the food source position, $rand()$ is a function that creates a random number in the range of 0 and 1, $\vec{P}_{P(x)}$ is the position of the tunicate. Eq. (6) is used to update the tunicate's position concerning the best position. The search agent's new position is updated using the first two best search agents as shown in Eq. (7). Figure 1 shows the main steps of TSA.

$$\overrightarrow{PD} = \overrightarrow{FS} - rand().\vec{P}_P(x) \tag{5}$$

$$\vec{P}_P(x) = \begin{cases} \overrightarrow{FS} + \vec{A}.\overrightarrow{PD}, & if \quad rand \geq 0.5 \\ \overrightarrow{FS} - \vec{A}.\overrightarrow{PD}, & if \quad rand < 0.5 \end{cases} \tag{6}$$

$$\overrightarrow{P_p}(x+1) = \frac{\vec{P}_P(x)+\vec{P}_P(x+1)}{2+c_1} \tag{7}$$

Initialize the N tunicate population.
Set the initial parameters like $Pmin, Pmax, \vec{A}, \vec{G}, \vec{F}, \vec{M}$
Calculate the fitness value of each search agent.
Explore the best search agent in the given search space.
While the stopping Condition is not satisfied
    Update the position of each search agent using Eq. (7).
    Adjust the updated search agent if it goes beyond the search space boundaries.
    Calculate the updated search agent fitness value. If there is a better solution than the previous optimal solution, then update previous optimal solution with the newest one.
End While
Return the best optimal solution.

Figure 1: The Pseudo Code of TSA

4. System Setup

4.1. Network Model

In the proposed work, we consider the following characteristics in setting the environment:

- The network area is M x M m^2.
- N sensor nodes are deployed in random positions within the environment.
- At the beginning of the network, all the sensor nodes have the same energy.
- The position of the Base Station (BS) is at the center of the network. Moreover, it doesn't have any energy constraints, unlike the sensor nodes.
- We assume a stationary scenario for all sensor nodes and BS.
- There are no physical medium constraints.
- The network is assumed to be secured.

4.2. Energy Model

The energy model introduced in [22] is used in our proposed work. Eq. (8) shows the energy consumed to transmit b bits over distance d.

$$E_{TX}(b,d) = \begin{cases} b * E_{elec} + b * \varepsilon_{fs} * d^2, & d < d_0 \\ b * E_{elec} + b * \varepsilon_{mp} * d^4, & d \geq d_0 \end{cases} \quad (8)$$

$$d_0 = \sqrt{\frac{\varepsilon_{fs}}{\varepsilon_{mp}}} \quad (9)$$

E_{elec} is the energy required to transmit a bit. ε_{fs} and ε_{mp} are the energy required to transmit b bits in free space and multipath, respectively. d_0 is the threshold distance that is calculated using Eq. (9). The required energy to receive b bits is calculated by Eq. (10). Eq. (11) shows the energy consumption by every cluster head to aggregate b bits from m cluster members, where E_{DA} is the energy required by the cluster head to combine the received bits.

$$E_{RX} = b * E_{elec} \quad (10)$$
$$E_{PR} = E_{DA} * b_m * m \quad (11)$$

5. Proposed Algorithm

5.1. TSA-Based Clustering

In TSA-based clustering, each search agent considers a solution, where the number of tunicates in the search agent equals the total number of cluster heads (CHs). Let $SA_i = \{ (x_1, y_1), (x_2, y_2), \ldots, (x_{d-1}, y_{d-1}), (x_d, y_d) \}$ is the i^{th} search agent, (x_1, y_1) is the position of the first node selected as a CH, and d is the total number of the CHs. Figure 2 gives a simple example of a population of five search agents with 5 CHs.

	(x_1, y_1)	(x_2, y_2)	(x_3, y_3)	(x_4, y_4)	(x_5, y_5)
SA_1	(84.07, 23.47)	(67.7, 23.47)	(33.71, 79.48)	(9.71, 14.55)	(93.4, 44.67)
SA_2	(56.78, 8.11)	(75.77, 86.86)	(24.35, 1.54)	(91.71, 68.67)	(70.93, 18.39)
SA_3	(54.68, 65.4)	(9.75, 60.19)	(95.71, 22.89)	(25.1, 73.17)	(70.93, 18.39)
SA_4	(75.77, 86.86)	(27.84, 26.29)	(81.42, 82.11)	(91.71, 68.67)	(28.58, 18.35)
SA_5	(12.69, 31.12)	(95.71, 22.89)	(58.52, 36.92)	(38.04, 78.02)	(34.99, 16.89)

Figure 2: Example of Search Agents Initialization in TSA-Based Clustering

Four objectives are considered while selecting the search agent that contains the best cluster heads:

- $F1$ is the cumulative distance between sensor nodes and their base station. $F1$ is calculated by Eq. (12), where N is the number of alive nodes, C_{CH} is a set of cluster heads.
- $F2$ is the cumulative distance between cluster heads and the base station. The value of $F2$ is calculated by Eq. (13), where M is the count of Cluster heads, C_j is the j^{th} cluster in C_{CH} set.

- $F3$ is the cumulative distance between each cluster member and the cluster head, which is calculated by Eq. (14), where SM_j is the count of members in the j^{th} cluster.
- $F4$ is the total energy of the system. $F4$ is given by Eq. (15), where e_{mj} is the consumed energy during sending the member m to cluster head j, while e_r is the consumed energy by the j^{th} cluster to receive a packet, and e_i is the consumed energy by a cluster head to send the data to the BS.

$$F_1 = \sum_{i=1}^{N} D\left(S_i, S_j\right) \ , S_j \in C_{CH} \mid BS \ \wedge \ i \neq j \tag{12}$$

$$F_2 = \sum_{j=1}^{M} D\left(C_j, BS\right) \ , \ C_j \in C_{CH} \tag{13}$$

$$F_3 = \sum_{j=1}^{M} \sum_{m=1}^{SM_j} D\left(S_m, C_j\right) \tag{14}$$

$$F_4 = \sum_{j=1}^{M} \sum_{m=1}^{SM_j} e_{mj} + SM_j * e_r + e_i \tag{15}$$

The final value of the objective function is given by Eq. (16), where a, b, c, and d are weight factors, and their values should satisfy Eq. (17).

$$Fitness(SA_i) = a * F1 + b * F2 + c * F3 + d * F4 \tag{16}$$

$$a + b + c + d = 1 \tag{17}$$

Figure 3 shows the pseudo-code of the full steps of TSA-based Clustering. The cluster formation phase starts after selecting the optimal cluster head. The remaining sensor nodes are considered normal nodes. The normal nodes choose to join a cluster if both of the following conditions are met:
- The CH is within the communication range of the normal node.
- The distance between the normal node and the selected CH is less than the distance to the other CHs.

Initialize the $\boldsymbol{N}$ search agents with $\boldsymbol{D}$ tunicates. Whereas $\boldsymbol{D}$ = 10% of alive sensor nodes.
Adjust the search agents that go beyond the search space boundaries.
Map the initialized search agents to the nearest nodes based on lower and upper boundaries.
Set the initial parameters like $\boldsymbol{Pmin}, \boldsymbol{Pmax}, \vec{\boldsymbol{A}}, \vec{\boldsymbol{G}}, \vec{\boldsymbol{F}}, \vec{\boldsymbol{M}}$
Calculate the fitness value of each search agent using Eq. (16).
Maintain the search agent with the best fitness value as the optimal cluster heads
While the stopping Condition is not satisfied
    Update the position of tunicates each search agent using Eq. (7).
    Adjust the updated search agents that go beyond the search space boundaries.
    Map the updated search agents to the nearest nodes.
    Calculate the updated search agent fitness value using Eq. (16).
    Update the best solution If there is a better solution than the previous optimal solution.
End While
Return the optimal cluster heads.

Figure 3: The Pseudo code of TSA-Based Clustering

5.2. TSA-Based Routing

TSA-based routing is used to create paths from every CH to the BS. Every CH can use relay nodes to transmit the data if the BS is not in the communication range of the CH. The objective here is to select the shortest path to the BS and reduce the number of relay nodes in the path.

In TSA-based routing, each search agent considered a solution. The number of tunicates in the search agents equals the number of the CHs. In the beginning, we initialize each tunicate in the search agents with a random number between 0 and 1. Figure 4 is an example of the initialization of 5 search agents and 5 CHs. Let $SA_i = \{r_1, r_2, \dots, r_{d-1}, r_d\}$ be the i^{th} Search agent where each search agent has d tunicates. Each tunicate in the search agent represents a CH.

	r_1	r_2	r_3	r_4	r_5
SA_1	0.559623	0.205513	0.817395	0.002305	0.685859
SA_2	0.648776	0.636287	0.773266	0.897573	0.119135
SA_3	0.687275	0.400554	0.117371	0.310278	0.123296
SA_4	0.618029	0.184302	0.995186	0.025489	0.702208
SA_5	0.183694	0.678463	0.650315	0.622811	0.655443

Figure 4: Example of Search Agents Initialization in TSA-Based Routing

Each random number in the search agent is mapped to one of the relay nodes. This mapping happens by calculating the next hop (relay node) using Eq. (18). $NextHop_s$ is the next hop that the node s will transmit data to it. PNH_s is the list of relay nodes in the communication range of relay node s. If the BS is in the communication range of s, it will be added to PNH_s. r_s is the random value that is assigned to the relay node, and Num_PNH_s is the count of prospective candidates as a next hop.

$$NextHop_s = PNH_s(\lceil Num\_PNH_s * r_s \rceil) \quad (18)$$

Table 1 shows an example of the next-hop selection process if we have 5 cluster heads.

Table 1: Next Hop Selection Process

Cluster Head	PNH_s	Num_PNH_s	r_s	$\lceil Num\_PNH_s * r_s \rceil$	$NextHop_s$	$Path_s$
CH_1	CH_3, CH_5	2	0.559623	2	CH_5	$CH_1 \rightarrow CH_5 \rightarrow BS$
CH_2	CH_1	1	0.205513	1	CH_1	$CH_2 \rightarrow CH_1 \rightarrow CH_5 \rightarrow BS$
CH_3	CH_1, CH_2, BS	3	0.817395	3	BS	$CH_3 \rightarrow BS$
CH_4	CH_3, CH_5	2	0.002305	1	CH_3	$CH_4 \rightarrow CH_3 \rightarrow BS$
CH_5	CH_2, BS	2	0.685859	2	BS	$CH_5 \rightarrow BS$

For calculating the fitness of the created solution by search agents, two objectives are considered:

- Minimizing the maximum distance from relay node s to the BS. The maximum distance is calculated using Eq. (19), where $HopCount(s)$ is the count of hops in $Path_s$.
- Minimizing the maximum number of hops. The maximum number of hops is given by Eq. (20).

$$Min(Max\_D) = Max\left\{\sum_{h=1}^{HopCount(s)-1} D(Path_s(h), Path_s(h+1)) \mid \forall\, s, 1 < s \leq M\right\} \quad (19)$$

$$Min(Max\_Hop) = Max\{HopCount(s) \mid \forall\, s, 1 < s \leq M\} \quad (20)$$

The fitness value of the search agent is calculated by Eq. (21), where W_1 and W_2 are constant factors. The search agent with a minimum fitness value is the best solution. Figure 5 presents the pseudo-code of the TSA-based routing algorithm.

$$Fitness(SA_i) = W_1 * Min(Max\_D) + W_2 * Min(Max\_Hop) \quad (21)$$

Initialize the $\boldsymbol{N}$ search agents with $\boldsymbol{D}$ tunicates. Whereas $\boldsymbol{D}$ = number of relay nodes.
Adjust the search agents that go beyond the search space boundaries.
Set the initial parameters like $\boldsymbol{Pmin}, \boldsymbol{Pmax}, \vec{\boldsymbol{A}}, \vec{\boldsymbol{G}}, \vec{\boldsymbol{F}}, \vec{\boldsymbol{M}}$
Calculate the fitness value of each search agent using Eq. (21).
Maintain the search agent with the best fitness value as the optimal solution.
While the stopping Condition is not satisfied
    Update the position of tunicates each search agent using Eq. (7).
    Adjust the updated search agents that go beyond the search space boundaries.
    Get the next hops of updated search agents using Eq. (18).
    Calculate the updated search agent fitness value using Eq. (21).
    Update the best solution If there is a better solution than the previous optimal solution.
End While
Return the optimal created routes.

Figure 5: The Pseudo Code of TSA-Based Routing

6. Experiment and Evaluation

MATLAB R2016b is used to simulate the proposed algorithm. As mentioned in the related work section, there are many routing algorithms proposed recently to enhance the network lifetime and conserve energy. Researchers try to find the best algorithms to enhance clustering and routing processes. Because the wrong choice of routing algorithm can affect the performance of the network lifetime despite the usage of a good clustering algorithm. Therefore, one of the main objectives of the experiments is to prove that the wrong choice of routing algorithm negatively affects the network lifetime even if the same clustering algorithm is used. Moreover, our experiment is applied to prove the ability of the proposed algorithm to extend the network lifetime more than the other algorithms. The proposed algorithm is compared with three of the most used metaheuristic-based routing algorithms in many research papers such as for transmitting data from cluster heads to the base station. These algorithms are GWO, PSO, and ACO. We chose these algorithms because they are some of the most commonly used states of art algorithms [13], [14], [16], [17], and [18]. To show how the selected routing algorithm could affect the network lifetime, our proposed TSA-based clustering is used in the clustering phase with the previously mentioned metaheuristic-based routing algorithms. Moreover, the proposed objective function is used with all the compared routing algorithms.

Tables 2, 3, and 4 show the network simulation parameters, TSA-based clustering parameters, and TSA-based routing parameters, respectively. To make the comparison more impartial, all the compared algorithms have the same number of iterations and search agents. The number of search agents equals 20 and the number of iterations equals 50 as mentioned in Table 4. The remaining parameters of PSO, ACO, and GWO are like [13], [14], and [18], respectively.

Table 2: Network Simulation Parameters

Parameters	Values	Parameters	Values
Network Area	100 x 100 m^2	E_{elec}	50 nJ/bit
Number of Nodes	100	ε_{fs}	10 $pJ/bit/m^2$
Base Station Position	(50 , 50)	ε_{mp}	0.0013 $pJ/bit/m^4$
Packet Size	4000 bits	E_{DA}	5 $nJ/bit.$
Initial Energy of Nodes	0.5 J		

Table 3: TSA-Based Clustering Parameters

Parameters	Values
Search agents	20
Number of Iterations	50
P_{min}	1
P_{max}	4
$a = b = c = d$	0.25

Table 4: TSA-Based Routing Parameters

Parameters	Values
Search agents	20
Number of Iterations	50
P_{min}	1
P_{max}	4
W_1	0.3
W_2	0.7

The comparison is carried out in terms of network lifetime. The network lifetime is calculated by the number of rounds, where each round consists of two phases. The steady-state phase is the first phase. In this phase, the clustering algorithm selects the optimal cluster head, and the routing algorithm creates the optimal paths from cluster heads to the base station. The data transmission phase is the second phase. In this phase, all the cluster members transmit the data to their cluster head, and the cluster head transmits the collected data to the base station. Moreover, the comparison is carried out in terms of the first dead node (FDN), half-dead node (HDN), and last dead node (LDN). FDN is the time before the death of the first node, HDN is the time interval before 50% of nodes became dead nodes, and LDN here is defined as the time until 80% of the nodes became dead nodes. The reason for choosing 80% is that the creation of the clusters is not stable after 80% of the nodes became dead nodes. Moreover, the total number of packets sent from cluster members to the cluster heads is used to compare the performance of algorithms. Furthermore, the number of packets sent from cluster heads to the base station is compared.

Figure 6 shows the number of rounds that every algorithm achieved until 80% of sensor nodes became dead nodes. It shows that the proposed algorithm achieves higher than the other algorithms, where it algorithm takes 1429 rounds. While GWO, PSO, and ACO achieved 1408, 1416, and 1425, respectively. This means that the proposed algorithm can extend the lifetime of the network. Moreover, it can conserve nodes' energy more than the other algorithms.

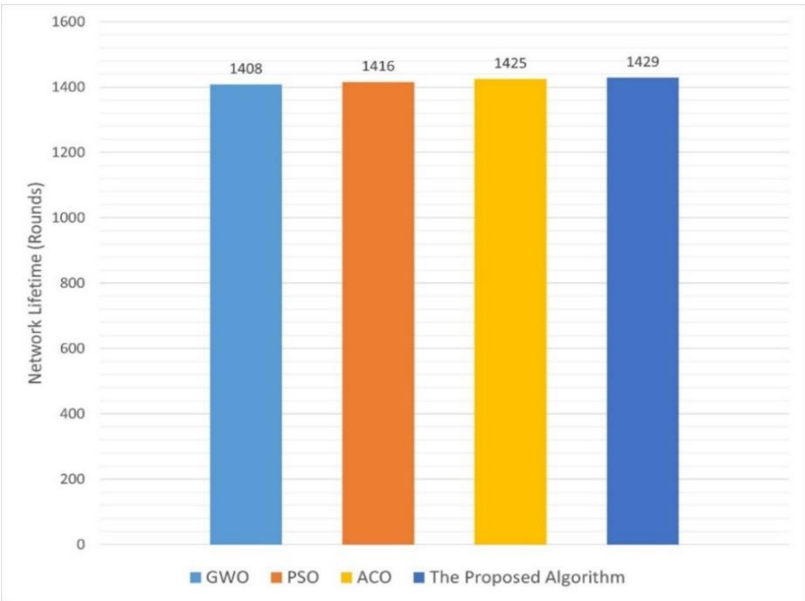


Figure 6: The Lifetime of The Network

Figure 7 shows three main metrics to evaluate the network lifetime. These metrics are FDN, HDN, and LDN. Despite the proposed algorithm cannot overcome the other algorithms in the FDN, it outperforms the others in HDN and LDN, where the FDN in the proposed algorithm is at round 559. While FDN happens in ACO, PSO, and GWO at rounds 634, 593, and 560, respectively. The HDN of the proposed algorithm is at round 1104. While the HDN in GWO, PSO, and ACO happens at rounds 1069, 1069, and 1089, respectively. Moreover, the LDN in the proposed algorithm happens in round 1429 which is higher than the other algorithms. Therefore, the proposed algorithm outperforms the other algorithms during the network lifetime despite FDN happening first. Furthermore, it proves that the TSA-based routing algorithm works efficiently with the TSA-based clustering algorithm more than the other algorithms in terms of the network lifetime.

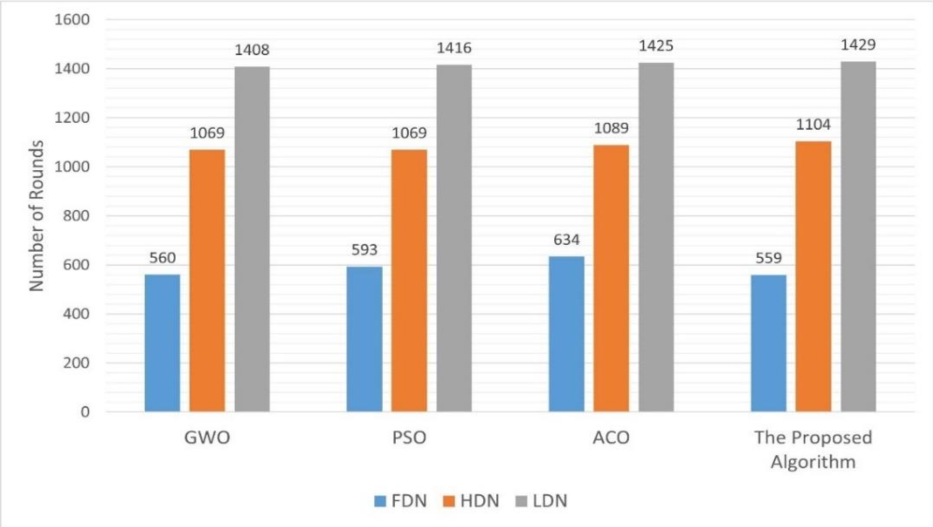


Figure 7: Lifetime metrics (FDN, HDN, LDN)

Figure 8 shows the total number of packets sent from cluster members (CMs to cluster heads (CHs) during the network lifetime for every algorithm. The figure shows that the ACO outperforms the proposed TSA algorithm, where the total number of packets received by the cluster heads using ACO is 100159 packets. The total number of packets received by the cluster heads using the proposed algorithm is 99643 packets. However, the proposed algorithm outperforms GWO and PSO which scored 99439 packets and 99431 packets, respectively.

Figure 8: Number of packets sent from CMs to CHs

Figure 9 shows the total number of packets sent from cluster heads (CHs) to the base station (BS) during the network lifetime. The figure shows the convergence of the performance of both the proposed algorithm and the ACO, where ACO was higher than the proposed algorithm by only one packet. The total number of packets sent from CHs to BS using the proposed algorithm and ACO are 11468 packets and 11469 packets, respectively. Despite that, the proposed algorithm outperforms GWO and PSO algorithms which score 11465 packets and 11366 packets, respectively.

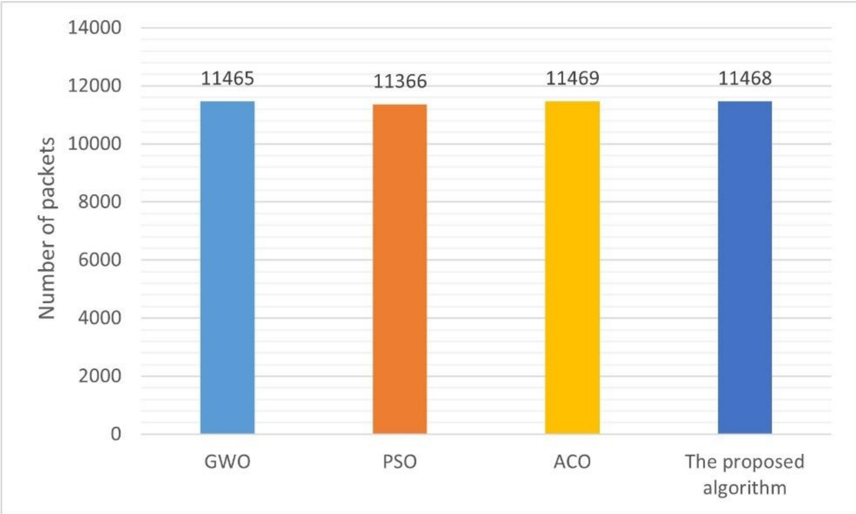


Figure 9: Number of packets sent from CHs to BS

7. Conclusion and Future Work

The routing algorithm is one of the critical issues for IoT networks. Many researchers work on proposing efficient routing algorithms to extend the network lifetime. This paper proposes Tunicate Swarm Algorithm (TSA) for solving clustering and routing issues for IoT networks. The TSA-based clustering algorithm is used to select the optimal cluster heads. Four factors are considered while selecting the best cluster heads. The factors are the distance between cluster members and cluster heads, the distance between the sensor node and its base station, the total energy of the system, and the distance between cluster heads and the base station. After selecting the optimal cluster heads, normal nodes start to join a cluster if the normal node is close to the cluster head. After creating clusters, the TSA-based routing algorithm starts to create routes from the cluster heads to the base station. Two factors are considered while creating the routes. The first one is minimizing the longest path to the base station. The second factor is minimizing the maximum number of hops in the paths.

The proposed algorithm is compared with three of the most used metaheuristic-based routing algorithms. These algorithms are used recently in many research papers to solve routing issues. These algorithms are GWO, PSO, and ACO. The same clustering methods are used in all the algorithms. The result proves that the proposed algorithm outperforms the other algorithms in the network lifetime and HDN metrics. The proposed algorithm extends the network lifetime to 1429 rounds. This is higher than the other algorithms. Furthermore, the HDN of the proposed algorithm is extended to be 3.17% more than GWO and PSO, and

1.36% more than ACO. Moreover, the proposed algorithm outperforms GWO and PSO algorithms in terms of the number of packets sent from CMs to CHs and from CHs to BS.

For future work, more experiments with different parameters needed to be done to prove the superiority of the proposed algorithm in different conditions. These parameters could be the scalability and the position of the base station Moreover, the proposed algorithm can be enhanced to perform better in terms of FDN metric, which can lead to conserving more energy and extending the network lifetime. Furthermore, the proposed algorithm can be enhanced by allowing intra-cluster routing. It can be tested in a heterogeneous network.

References

1. R. ElGawish, M. Hashim, R ElGohary, R. Abu-Rizka, A threshold-based technique to cluster ransomware infected medical records on the internet of medical things, International Journal of Intelligent Computing and Information Sciences. 22(1) (2021) 16-31.
2. D. K. A. A. Rizk, H. M. Hossny, E. M. El-Horbaty, A. M. Salem, Smart hospital management systems based on internet of things: challenges, intelligent solutions and functional requirements, International Journal of Intelligent Computing and Information Sciences. 22(1) (2021) 32-43.
3. M. Aboubakar, M. Kellil, P. Roux, A review of IoT network management: current status and perspectives, Journal of King Saud University – Computer & Information Sciences. 34(7) (2022) 4163-4176.
4. M. Chernyshev, Z. Baig, O. Bello, S. Zeadally, Internet of Things (IoT): research, simulators, and testbeds, IEEE Internet of Things Journal. 5(3) (2018) 1637-1647.
5. E. Selem, A. Khedr, H. Nassar, HM-LEACH: Energy efficient heterogeneous mobile leach protocol for heterogeneous wireless sensor networks, International Journal of Intelligent Computing and Information Sciences. 14(1) (2014) 51-61.
6. P. Nayak, C. P. Reddy, Bio-inspired routing protocol for wireless sensor network to minimize the energy consumption, IET Wireless Sensor Systems Journal. 10(5) (2020) 229-235.
7. M. Sadrishojaei, N. J. Navimipour, M. Reshadi, M. Hosseinzadeh, A new clustering-based routing method in the mobile internet of things using a krill herd algorithm, Cluster Computing Journal. 25(1) (2022) 351-361
8. S. Rani, S. H. Ahmed, R. Rastogi, Dynamic clustering approach based on wireless sensor networks genetic algorithm for IoT applications, Wireless Network Journal. 26(4) (2020) 2307-2316
9. A. Shahraki, A. Taherkordi, Ø. Haugen, F. Eliassen, Clustering objectives in wireless sensor networks: a survey and research direction analysis, Computer Networks Journal. 180 (2020) 107376.
10. D. W. Sambo, B. O. Yenke, A. Forster, P. Dayang, Optimized clustering algorithms for large wireless sensor networks: a review, Sensors (Basekl, Switzerland). 19(2) (2019) 322.
11. S. M. Almufti, R. B. Marqas, V. A. Saeed, Taxonomy of bio-inspired optimization algorithms, Journal of advanced computer science & technology. 8(2) (2019) 23-31.
12. A. Darwish, Bio-inspired computing: Algorithms review, deep analysis, and the scope of applications, Future Computing and Information Journal. 3(2) (2018) 321-246.
13. Q. Wang, W. Liu, H. Yu, S. Zheng, S. Gao, F. Granelli, CPAC: Energy-efficient algorithm for IoT sensor networks based on enhanced hybrid intelligent swarm, Computer Modeling in Engineering & Sciences. 121(1) (2019) 83-103.

14. V. Anand, S. Pandey, New approach of GA-PSO -based clustering and routing in wireless sensor networks, International Journal of Communication Systems. 33(16) (2020) e4571.
15. A. Rezaeipanah, P. Amiri, H. Nazari, M. Mojarad, H. Parvin, An energy-aware hybrid approach for wireless sensor networks using re-clustering based multi-hop routing, Wireless Personal Communication Journal. 120(4) (2021) 3293-3314.
16. P. Maheshwari, A. K. Sharma, K. Verma, Energy efficient cluster based routing protocol for WSN using butterfly optimization algorithm and ant colony optimization, Ad Hoc Networks Journal. 110 (2021) 102317.
17. S. Sefati, M. Abdi, A. Ghaffari, Cluster-based data transmission scheme in wireless sensor networks using black hole and ant colony algorithms, International Journal of Communication Syststems. 34(4) (2021) e4768.
18. R. Ahmadi, G. Ekbatanifard, P. Bayat, BioWSN: A bio-inspired method for optimization of routing in wireless sensor networks, Mathematical Problems in Engineering. 2022 (2022) 4826388.
19. R. Dogra, S. Rani, S. Verma, S. Grag, M. M. Hassan, TORM: Tunicate swarm algorithm-based optimized routing mechanism in IoT-based framework, Mobile Networks and Applications, 26(6) (2021) 2365-2373.
20. A. D. Gupta, R. K. Rout, ROTEE: Remora Optimization and Tunicate swarm algorithm-based Energy-Efficient cluster-based routing for EH-enabled heterogeneous WSNs. (2022) e5372.
21. S. Kaur, L. K. Awasthi, A. L. Sangal, G. Dhiman, Tunicate swarm algorithm: A new bio-inspired based metaheuristic paradigm for global optimization, Engineering Applications of Artificial Intelligence. 90 (2020) 103541.
22. W. R. Heinzelman, A. Chandrakasan, H. Balakrishnan, Energy-efficient communication protocol for wireless microsensor networks, In: 33rd Annual Hawaii International Conference on System Science. (2000) 1-10.