

Traffic and delay aware routing using optimization algorithm for wireless sensor networks

P. Priyadharshini* and S. Pavalarajan

Department of Information Technology, PSNA College of Engineering and Technology, Dindigul, India

Abstract. The Internet of Things (IoT) is a system of machines, computing devices, electronic equipment, and different sensors. It forms a network, where the transmission of device-related data can be accomplished. The devices in the IoT are connected to each other through wireless links and form ad-hoc networks. In IoT based applications, the lifetime of the communicating nodes is a greater concern. The network lifetime can be maximized by introducing energy efficient data transmission in the network. Therefore, a traffic and delay-aware energy-efficient routing (TADEER) protocol for IoT-based networks are proposed in this work. The proposed technique assigns delay for transmitting data based on the criticality level of data and traffic rate at the forwarding nodes. Fixing delays for data transmission helps to avoid unnecessary transmissions. The route selection process is implemented using an optimization algorithm. A Fuzzy logic (FL) based biogeography-based optimization (BBO) algorithm is presented in this work. Thus, the number of data transmission and energy consumption can be reduced. The performance of the proposed method is evaluated by analyzing transmission delay, network lifetime, and energy consumption. By comparing the simulation results to the existing methods TEAR and ETASA, the simulation results are validated.

Keywords: Internet of Things, smart home, energy management, demand response, energy consumption, wireless sensor network

1. Introduction

In an IoT system, the devices involved in the applications can communicate with each other and transmit various kinds of data based on the applications. The collection of data at the devices and the transmission of collected data can be accomplished using the sensor nodes. The data transmission can be accomplished through wireless communication. Thus, it forms a wireless ad-hoc network with heterogeneous nodes that observe data and transmit

it through intermediate nodes to a cloud server. In IoT-based applications, the efficiency and reliability of WSN need to be taken into account. The most important problems in IoT are energy consumption of network, bandwidth limitation, throughput, security and network congestion. Hence, the effective realization of sensor networks and better communication scheme in the network is required for IoT systems. The proposed system should overcome the above-mentioned issues. IoT devices should function with better performance without compromising quality of service (QoS) to ensure uninterrupted operations [1]. Therefore, the energy efficiency and lifetime of IoT based ad-hoc networks are key concerns for their development into modern applications. The significant challenges that need to be solved in such

*Corresponding author. P. Priyadharshini, Assistant Professor, Department of Information Technology, PSNA College of Engineering and Technology, Dindigul, India. E-mail: priya@psnacet.edu.in.

networks are data sensing, transmission, and routing of data from source node to destination node [2]. IoT consists of different types of devices over different kinds of networks. Hence, the communication of data over the heterogeneous network is not easier. The sensor nodes of the IoT monitor different discrete events for a long time and produce large amount of data [1]. The manipulation of large amount of data is still a challenge for WSN and IoT systems, since the collection of large amount of data results in heavy traffic load. Also, the sensor nodes need to continuously transmit data to the cloud as well as to other nodes to make proper decisions, according to the varying environmental conditions [3]. Hence, an effective protocol to avoid congestion, data overhead is required. Also, data routing in the network should be energy efficient. Specific to the application of systems with IoT, the WSN has to be designed with effective routing schemes that overcome the following problems.

- i. Transmission of critical data needs to be done without interruption in order to take proper decisions to avoid undesirable events. However, the frequent transmission of critical data results in excessive energy consumption. Therefore, the data must be routed through a less energy-consuming path.
- ii. The next significant perception is delay of critical data transmission. The action must be taken in fraction of time, when the event occurs. So, the data has to be transmitted without delay.
- iii. Data traffic is another significant concern in the network. The nodes with high traffic dissipate more energy. For reducing energy costs, a low traffic path must be chosen.
- iv. In the conventional method of reducing number of transmissions, data compression and data prediction methods were applied. But in those methods, the transmission of critical data is not ensured.
- v. In the FL-BBO algorithm, the immigration and emigration rate are computed based on the fuzzy logic-based migration control parameter. Hence, optimization process takes less time.

Therefore, to overcome the above problems of IoT-based ad-hoc networks, an effective traffic and delay-aware energy efficient routing (TADEER) scheme is presented.

- The energy consumed by the sensors of idle state devices should be avoided by making the nodes

sleep during the normal operating conditions of the device and turning them awake, when the state of the device changes. Thus, the energy consumption of the network can be reduced significantly reduced by identifying energy efficient path.

- In the proposed TADEER method, the critical data are assigned with less delay for data transmission, which ensures that critical data is not missed. The route selection process is implemented using an optimization algorithm.
- A Fuzzy logic (FL) based biogeography-based optimization (BBO) algorithm is presented in this work.

The proposed protocol effectively reduces data traffic, energy consumptions in the network and the delay in transmission of critical data in the network. The performance of the proposed scheme is evaluated through simulations, and the results are verified with the relevant techniques. The upcoming sections include the survey of the previous studies, existing techniques taken for the comparisons, the proposed technique, the result parts obtained from the simulations and finally the conclusion section.

2. Literature survey

Considering the IoT-based networks, the problems were overcome by introducing different protocols in several papers. The works are discussed as follows. In [4], an energy-efficient centroid-based routing protocol (EECRP) was presented for IoT applications. The authors presented cluster based technique, which rotates the cluster head (CH) according to the centroid position. The algorithm allocates the energy load equally to the sensor nodes. It also reduces the energy taken for transmitting data to a larger distance. The centroid position is determined using node's residual energy. The authors demonstrated that the method outperforms the LEACH protocols. In [5], an ontology based work is presented that enables context sharing among network objects in WSNs. This is the first work that presents a context-based model for communication in WSN-based IoT. However, a splendid knowledge of the context of IoT applications should be known.

In [6], a priority-based and energy-efficient (PriNergy) routing protocol is presented based on the low-power and lossy network (RPL) model for IoT applications. Data routing is established according

to the type of data. Both the content-aware routing and the RPL protocol are jointly used [6]. The roaming data of transitional relay nodes is combined, which helps to do data aggregation at a higher rate. Thus, the network traffic is meritoriously reduced and delays are considerably reduced. An energy-aware routing scheme is proposed in [7] utilizing multi-objective algorithm for IoT applications. The routing protocol is presented based on factors like the node's lifetime, energy, and distance rate. Clustering is employed and the CH node is selected using the C-means method. In [8], a fuzzy clustering method is proposed to improve the energy efficiency of routing methods in IoT applications. The work also employed an immune-inspired routing algorithm for ensuring high reliability and reducing energy consumption for IoT nodes. The authors proved that the packet loss ratio and jitter ratio are lower compared to other optimized algorithms. Wang et al. [9] proposed a partial energy-aware routing algorithm to reduce energy consumption of WSN nodes in the ad hoc on-demand distance vector (AODV) protocol. This method considered the partial energy of each node to identify optimal link information for adjacent nodes.

In smart grid applications, WSNs are used to connect the different environments using IoT, and the operation of WSNs becomes a challenging concern. It is difficult to give reliable communication. [10] addresses the issue and presents an energy-efficient and quality-of-service (QoS)-aware routing protocol (EQRP) based on dynamic clustering. The protocol utilizes a bird mating optimization algorithm (BMO). This work enhances network reliability and redundant data retransmissions are reduced. In work given in [11], the idea of an "equivalent node" is presented to choose a relay node for establishing optimal communication and energy consumption. A probabilistic dissemination algorithm is presented to obtain optimal energy consumption and increase network lifetime. This work reduces energy consumption and ensures the transmission quality. Naji et al., [12] proposed a WSN based energy management scheme and implemented it in a smart building. The proposed method employs the Context-Based Reasoning (CBR) method to denote different types of buildings and offices. The authors controlled the electrical devices using a Finite State Machine (FSM). This work shows effective performance in reducing processing load when implementing multithreading in data acquisition. The lifetime of nodes is optimized through the Energy Aware Context Recognition Algorithm (EACRA), which helps

dynamically arrange the sensors to transmit data under particular conditions and avoid unnecessary data transmissions. EACRA increases the sensors' battery lifetime by optimizing the number of data samples, used components, and data transmissions. In [13], an IoT-based power monitoring method in smart grids is presented that measures the parameters such as voltage, current, power, and energy consumption of loads connected to smart grids. The real-time data of consumers is obtained and the details are taken by the power companies in smart grids, which perform energy management. The WSNs used in IoT are usually Low-Power and Lossy Networks (LLN), and hence, the data routing in these networks should be performed better in order to reduce energy consumption. In [14], fuzzy logic based energy-aware routing protocol is presented that takes network load, node residual energy, and number of transmissions in selecting the optimal route for data transmission. This protocol improves network lifetime and packet delivery ratio. In [15], an energy harvesting-aware routing algorithm (EHARA) is presented for heterogeneous IoT networks with ambient energy sources. This algorithm is improved by adding a parameter known as "energy back-off." This work improves network lifetime and quality-of-service (QoS) in different traffic load and energy conditions. Thangaramya et al., [16] presented a new routing method for IoT-based WSN. It employed Neuro-Fuzzy Rule Based Cluster Formation and routing algorithm. The work gives better performance in energy consumption, delivery ratio, delay, and network lifetime.

In [17], energy conservation in WSN is accomplished using a two-level data reduction technique. The Data and Energy Aware Passive (DEAP) clustering approach is for forming network clusters. A Dual Prediction (DP) is used to reduce number of transmissions between cluster members and cluster heads. In [18], the energy reduction in WSN is attained by using Distributed Similarity-based Clustering and Compressed Forwarding (DSCCF). This method exploits magnitude and trend similarity among the data and performs clustering in the network. Then, the dual prediction method and lossless data compression method are used to reduce the inter-cluster data payload.

With several random levels, both the energy and traffic heterogeneities are taken into account [19]. In the multi-heterogeneity scenario, where taking into account multi-level traffic heterogeneity is a novel idea, an energy model is introduced. Traffic and Energy Aware Routing (TEAR), an innovative rout-

ing algorithm, is introduced. TEAR chooses a CH by taking into account the energy levels and traffic requirements of each node. TEAR outperforms current methods by means of stability period (reliable longevity of the WSN prior to the death of its initial node).

Utilizing duty-cycling in cluster-based routing, as found in the Sleep-awake Energy Efficient Distributed (SEED) clustering algorithm, is one technique to decrease duplicate transmission and boost energy efficiency [19]. A traffic and energy aware routing protocol (TEAR) is recommended to address difficulties with traffic heterogeneity and energy among sensor nodes. To ensure load balancing, TEAR refrains from choosing a cluster head node with a little energy and high traffic rate. In a heterogeneous wireless sensor network situation, a hybrid approach termed the energy and traffic aware sleep-awake (ETASA) mechanism is offered to increase energy efficiency and boost load balancing. This approach is taken into considerations for comparing the newly introduced technique.

The challenge of determining the best path from source to destination sensor nodes to be employed in transmitting data connected to financial technology (FinTech) is expected to be resolved by a novel Non-linear Lévy Brownian Generalized Normal Distribution Optimization (NLBGNDO) algorithm, according to a predicted solution [22]. By implementing a friendship-based trust mechanism throughout each selection phase, this objective function can be used to retain the relay-node candidates that have been chosen. It can determine the sparsity/density ratio of sensor nodes, the node residual energy, experienced reaction time, and inter-node distance. It can increase revenue for organizations and cut costs.

In all the available papers discussed on the IoT-based WSN, for SEM the bandwidth limitation is not considered and network throughput improvement is not validated. Also, the significant of critical data transmission is not discussed. Therefore, the proposed routing protocol should consider the data criticality and delay of data transmission also.

3. Methodology

In this work, an effective routing scheme is presented for ad-hoc networks in IoT applications. A traffic and data routing protocol is presented to overcome the issues mentioned in Section 1. A typical IoT-based network with devices, sensor nodes, and servers is depicted in Fig. 1.

The proposed routing protocol should be less energy-consuming which can be obtained by reducing the communications. For attaining reduced transmissions, the data criticality is outlined as presented above. It must also be considered that the routing protocol should be delay-aware, since the critical data must be sent without much delay. So, the routing is performed by setting a very small amount of delay to the data based on its criticality. The routing algorithm should consider the factors, such as path delay ((D_{path})), path residual energy ($(E_{res_{path}})$), energy consumption of the path ($(E_{cons_{path}})$) and data traffic rate of the path/node ((TR)). The energy consumption of the network increases, when the node frequently transmits data. When increased delay is assigned to the low-critical data, the number of data transmissions is reduced. Also, reduced number of transmissions helps in minimizing path traffic. The notations used in the proposed protocol are listed in Table 1.

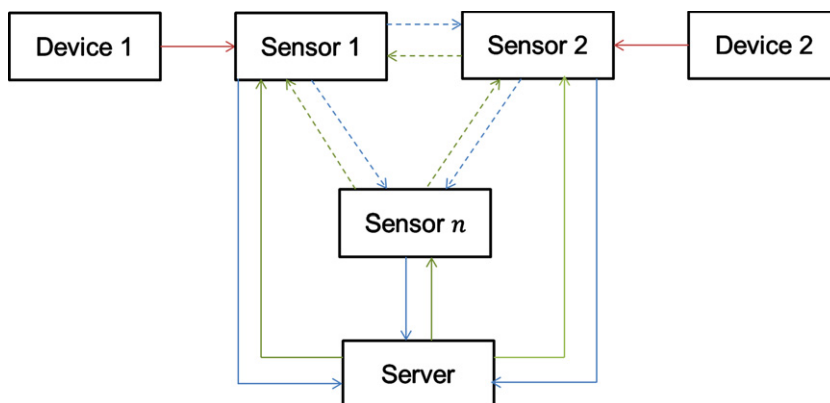


Fig. 1. Block diagram of system operation.

Table 1
Notations of the parameters used in the system

Notation	Parameter
D_{path}	Path delay
$E_{res_{path}}$	Path residual energy
$E_{cons_{path}}$	Energy consumption of the path
$TR(i)$	Data traffic rate of the path/node i
C_f	Criticality level of data
$D_{node}(init)$	Data transmission delay at the initial node
D_{path}	End-to-end delay between source node and the SMU
D_{node}	Node-to-node delay
D_{path}	Total end-to-end delay
$\alpha(i)$	Is a delay factor which is set based on TR value
$\beta(i)$	Is a delay factor that is set based on the residual energy of the node
N	Number of sensor nodes
E_{trans}	Energy consumed for transmitting data
E_{rece}	Energy consumed for receiving data
m_i	Packet size of a message
d	Distance of transmission
E_{ckt}	Energy dissipated by the circuit
ε_{fs}	Fixed radio parameter in free space
ε_{mp}	Fixed radio parameter in multipath
E_0	Energy lower bound
H_e	Energy heterogeneity factor
E_T	Total initial energy
m_0	Is the lower bound
H_{ii}	Traffic heterogeneity factor
m_{i_c}	Number of message transmitted by node i in previous rounds
$E_T(i)$	Initial energy of node i
$E_{res}(i)$	Residual energy of node i

3.1. Energy model and Traffic rate

In the IoT based WSN, the energy consumed for transmitting a message with m_i packets for a distance (d) can be computed using Equation (1) [19].

$$E_{trans}(m_i, d) = \begin{cases} m_i E_{ckt} + m_i \varepsilon_{fs} d^2 & \text{if } d < d_0 \\ m_i E_{ckt} + m_i \varepsilon_{mp} d^4 & \text{if } d \geq d_0 \end{cases} \quad (1)$$

E_{ckt} , energy dissipated by the circuit to transmit one bit. ε_{fs} and ε_{mp} are fixed radio parameters related to energy dissipation of the transmitter amplifier in free space and multipath, respectively, $d_0 = \sqrt{\frac{\varepsilon_{fs}}{\varepsilon_{mp}}}$. Energy consumed for receiving a message with m_i packets is given by,

$$E_{rece}(m_i) = m_i E_{ckt} \quad (2)$$

The initial energy of the nodes in the heterogeneous network is randomly distributed over $[E_0, E_0(1 + H_e)]$. E_0 is the energy lower bound and H_e is energy heterogeneity factor. Thus, the total ini-

tial energy of the network is given by,

$$E_T = \sum_{i=1}^N E_0(1 + H_{ei}) \quad (3)$$

where H_{ei} is energy heterogeneity factor of node i . Message length of a data packet of node i can be computed as,

$$m_i = m_0(1 + H_{ii}) \quad (4)$$

where m_0 is the lower bound and H_{ii} traffic heterogeneity factor of node i .

The number of message transmitted by node i in the previous transmission rounds is computed as,

$$m_{i_c} = \sum_{j=1}^r m_i(j) \quad (5)$$

where r is current round number. The average traffic rate (TR) of node i in current round r is given by,

$$TR(i) = \frac{m_{i_c}}{r} \quad (6)$$

The nodes with large data traffic rates generally have a large number of transmitted data packets. For sending such large number of messages, more energy is dissipated from the node. The energy consumption rate of the node is higher. Hence, in such conditions, the numbers of transmissions need to be reduced. For reducing transmissions, the delay is increased in the proposed method. Thus, the traffic rate of a node i and the delay at the node i is proportional to each other. If the traffic rate in a transmission path is higher, larger delay is assigned at the nodes present in that path.

3.2. Computation of delay based on node's traffic rate

The end-to-end delay between the source node and the SMU (D_{path}) can be obtained by the addition of node-to-node delay (D_{node}) along the path. The total end-to-end delay (D_{path}) can be computed using expression (1).

$$D_{path} = \sum_{i=m}^l D_{node}(i, i+1) \quad (7)$$

where i signifies the i^{th} node in the network. $D_{node}(i, i+1)$ represents the propagation delay between the i^{th} node and its subsequent node ($i+1$); m and l denote the indices of path's start and end node in the network. The propagation delay depends

Table 2
Critical factor value based on data type

Message	Data type	Critical factor
Control message for changing device state	Most critical data	0.9
Energy consumption of device which is in ON	Most critical data	0.9
Energy consumption of device which is in OFF	Moderately critical data	0.6
Load of the device	Non-critical data	0.3
Device running time	Non-critical data	0.3

on the channel bit rate and message length. The propagation between nodes in the network depends on the transmitted packet rate. In addition, in the proposed work, an extra delay is added, while computing the end-to-end delay (D_{path}). The additional delay is computed at the initial node where the data transmission is started. Additional delay at the initial node m is represented as $D_{add}(m)$ and this delay is computed based on the criticality level of data (Cf).

Thus, Equation (7) can be modified as below.

$$D_{path}(m, l) = \sum_{i=m}^l D_{node}(i, i+1) + D_{add}(m) \quad (8)$$

$$D_{add}(m) = \frac{D_{node}(m, m+1)}{Cf} \forall_{available\ m + 1\ nodes} \quad (8.1)$$

The Cf parameter is computed using Table 2.

The nodes with a large traffic rate consume more energy. If a node with a large traffic rate forwards all of its data, it dissipates more energy. Such nodes should be avoided in the transmission. Nodes with less traffic have very little data to forward. Hence, such nodes can be selected for transmission. This is attained by setting a large delay for high traffic nodes and a lower delay for low traffic rate nodes. Hence, another delay is added at each forwarding node, which is denoted as D_{TR} . At each node, along with the node-to-node delay, an extra delay is added according to the data traffic rate at the node.

$$D_{path}(m, l) = \sum_{i=m}^l (D_{node}(i, i+1) + D_{TR}(i, i+1)) + D_{add}(m) \quad (9)$$

$D_{TR}(i, i+1)$ represents delay between node i and $i+1$ based on traffic rate at node $i+1$. Its computation is derived as follows.

Table 3
Delay factor assigned for different traffic rates

Traffic rate (%) (%TR(i))	Type	Delay factor $\alpha(i+1)$
>80%	Large	0.7
50% to 80%	Moderate	0.5
20% to 50%	Low	0.3
<20%	Very low	0.1

Table 4
Delay factor assigned according to residual energy

Residual energy (%)	Delay factor $\beta(i+1)$
>80%	0.2
50% to 80%	0.5
20% to 50%	0.7
<20%	0.9

$$D_{TR}(i, i+1) = D_{node}(i, i+1) \times \alpha(i+1) \quad (9.1)$$

where $\alpha(i+1)$ is a delay factor which is set based on TR value at the node computed using (6). The traffic rate at a node is proportional to the energy consumption rate at that node. The TR value is computed in percentage and the delay factor is assigned corresponding to the TR as given in Table 3.

The residual energy ($E_{res}(i)$) at a node i is computed from the difference between the initial energy of the node ($E_T(i)$) and the total energy consumed by the node in previous transmission rounds. Based on the residual energy, a delay factor can be set as $\beta(i+1)$. The value, $\beta(i+1)$ is selected in the range of 0.2 – 0.9. The selection of $\beta(i+1)$ is performed as given in Table 6. A high delay factor is assigned to the nodes having less residual energy and vice versa.

Consequently, final equation for computing $D_{path}(m, l)$ can be obtained as,

$$D_{path}(m, l) = \sum_{i=m}^l (D_{node}(i, i+1) + D_{TR}(i, i+1) + D_{Eres}(i, i+1)) + D_{add}(m) \quad (10)$$

$$D_{Eres}(i, i+1) = D_{node}(i, i+1) \times \beta(i+1) \quad (10.1)$$

$D_{Eres}(i, i+1)$ represents delay between node i and $i+1$ based on residual energy of node $i+1$.

From the above delay computation strategy, the delay is directly proportional to traffic rate and indirectly proportional to Cf and residual energy. Therefore, the route with minimum space $D_{path}(m, l)$ is selected from the available routes in the network.

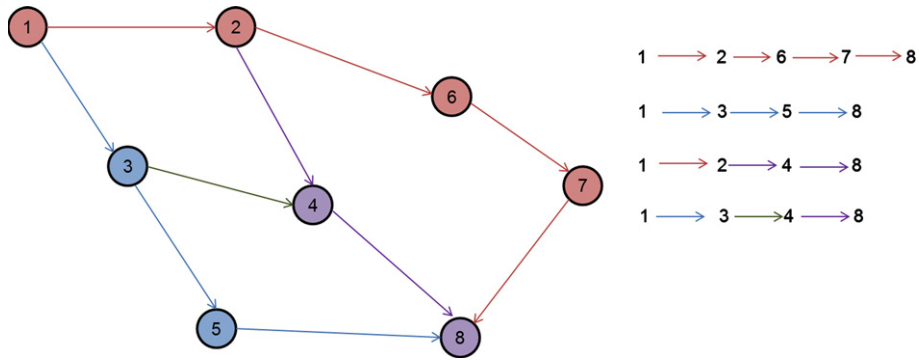


Fig. 2. Typical network diagram for representing proposed routing protocol.

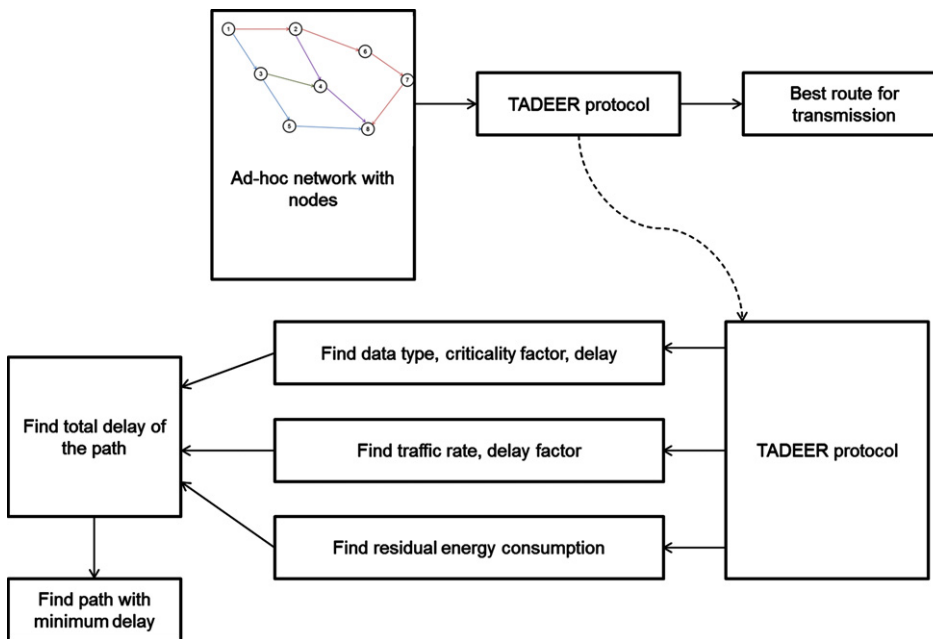


Fig. 3. Overall diagram for the proposed work.

3.3. TADEER protocol

The transmission of data in the network and route selection process is illustrated here. A simplified representation of the proposed network is depicted in Fig. 4. Node 1 has a message about device state to deliver to other nodes. This message comes under the critical message class; hence, the delay assigned at the node D_{add} is computed using Equation (8.1). Then, all possible routes from the starting node 1 to the end node 8 are checked and $D_{path}(1, 8)$ is determined for all possible routes using Equation (9). The route with fewer turns D_{path} is chosen for transmission.

If an appliance in the network, communicates data about its state, the data is considered as critical data

and C_f is assigned a value greater than 0.7. Similarly, the traffic rate at that node is also checked and α is selected according to the TR value. Thus $D_{node}(init)$ is identified. Consequently, the next forwarding node is selected based on $D_{node}(i)$. For all possible forwarding paths, D_{path} is computed, and the path with the least amount of delay is chosen for the most critical data. For the least critical data, the path with more delay is chosen. Thus the number of data transmissions in the work is reduced by setting different delays, which considerably reduce energy consumption in data transmissions.

The proposed traffic and delay-aware routing technique helps in reducing data transmission by choosing the path with less traffic. The routing

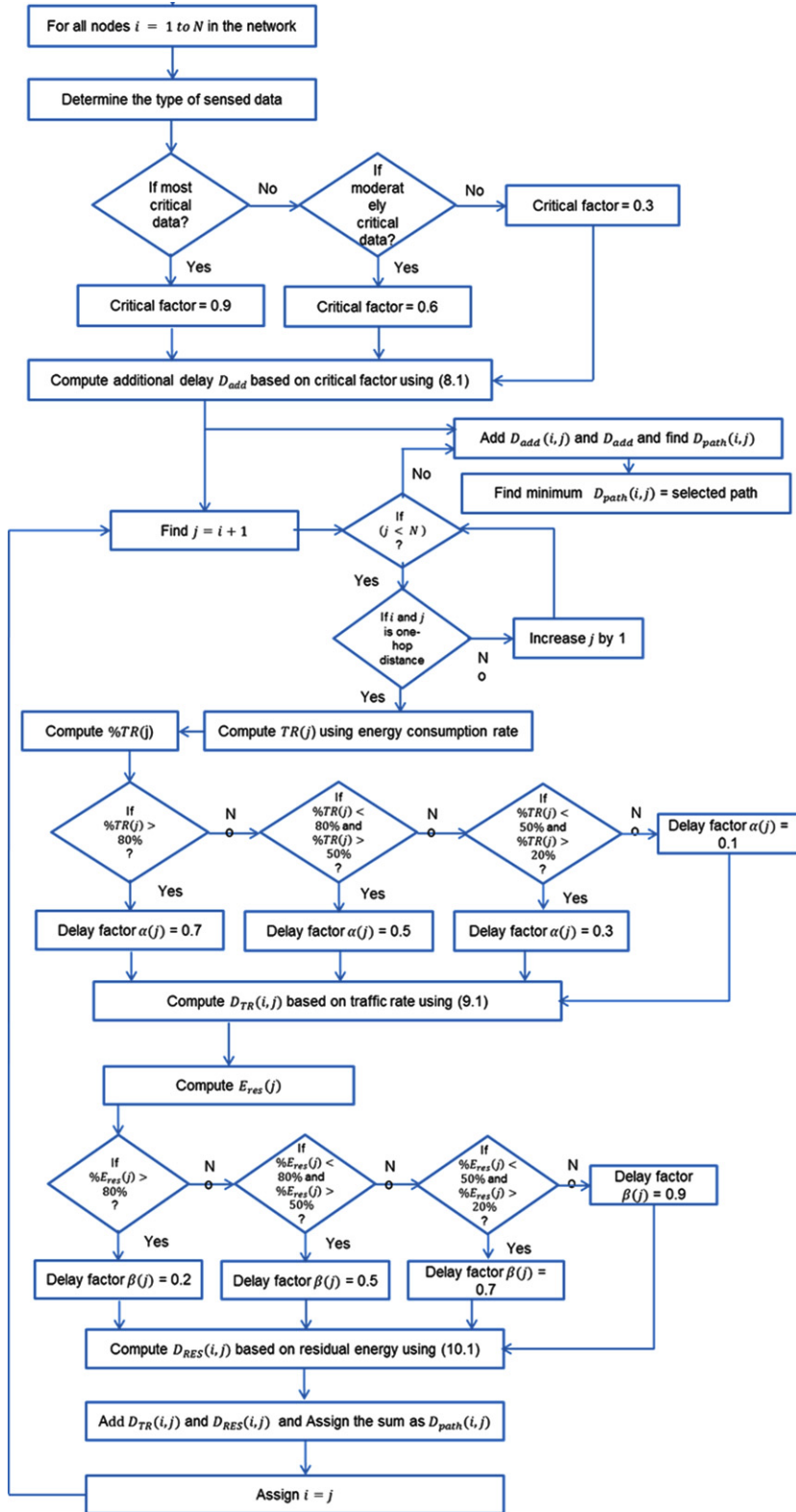


Fig. 4. Flowchart of TADEER protocol.

method allocates delay to the path according to the data criticality factor. By reducing the data transmission based on their criticality level, the energy consumption of the nodes is reduced and the network lifetime increases. This proposed work ensures that all the critical data is sent without interruption. In the conventional method of reducing number of transmissions, data compression and data prediction methods were applied. But in those methods, there is no assurance that the most important data will be transmitted. Also the accuracy of data transmission is affected because of prediction and compression. In the proposed TADEER method, the critical data is assigned with less delay for data transmission, which ensures that critical data is not missed.

The overall block diagram of the newly proposed work is shown in Fig. 3. A flow chart for the proposed TADEER protocol is given in Fig. 4. The steps involved in implementing the TADEER protocol are explained as follows.

Algorithm 1: TADEER protocol

- Step – 1: Read the measurements from the devices.
 - Step – 2: Get the data measured by the nodes from all nodes (from node 1 to N).
 - Step – 3: Determine the type of sensed data at the starting node.
 - Step – 4: Find C_f using Table 4.
 - Step – 5: Compute $D_{add}(m)$ using Equation (8)
 - Step – 6: Find out next hop nodes for node i
 - Step – 7: Compute TR rate for all next hop nodes ($i + 1$) using energy consumption rate
 - Step – 8: Determine %TR
 - Step – 9: Find $\alpha(i + 1)$ using Table 5.
 - Step – 10: Compute $D_{TR}(i, i + 1)$ using Equation (9.1)
 - Step – 11: Find $\beta(i + 1)$ using Table 6.
 - Step – 12: Compute $D_{Eres}(i, i + 1)$ using Equation (10.1)
 - Step – 13: Compute $D_{path}(i, i + 1)$ using Equation (10)
 - Step – 14: Find out minimum D_{path} from all available path delays and select the route for data transmission
-

3.4. Optimal route selection using TADEER protocol

In data transmission using the TADEER protocol, minimum delay, minimum energy consumption, and less traffic are considered for selecting routes. Furthermore, optimal path selection with traffic-aware delay and a shorter distance path is required. Therefore, optimal path with less delay based on TADEER and the shortest path should be selected for better performance. For the optimal route selection process, the Fuzzy logic biogeography based optimization (FL-BBO) algorithm [20] is employed. The

optimization process in FL-BBO is explained as follows.

The geographical dispersal of biological species is taken into account and an evolutionary algorithm is developed, which is the conventional BBO algorithm [21]. In BBO algorithm, species migrate from one habitat to another habitat, according to the Habitat Suitability Index (HSI) factor. This factor determines the amount of suitability for a species to survive in the habitat. A high value of HSI should be possessed by the habitat to be suitable for the species. Suitability Index Variables (SIV) is the factors that measure the habitability of an environment. Different environmental factors like rainfall, temperature, crop diversity, etc. fall under SIVs. This metric is variable for different habitats, which is based on the suitability factors. In large HSI habitats, many species are present and they have a higher emigration rate. The immigration rate (λ) and emigration rate (μ) are the main points in the BBO algorithm. These rates are computed for every habitat (H_i) using the equations (11) and (12) respectively.

$$\lambda_i = I \left(1 - \frac{K(i)}{N} \right) \tag{11}$$

$$\mu_i = E \left(\frac{K(i)}{N} \right) \tag{12}$$

where E and I are the maximum values of emigration and immigration rates, respectively. N and K signifies the total number of habitats and the number of species in i^{th} habitat. The habitats are taken as the population of optimization problems. The SIVs are equivalent to the components of the solutions and HSI is similar to the fitness function for the problem. The habitats with highest μ rates and lowest λ rates are considered better for dwelling, and the species intend to come out of the corresponding habitat. Moreover, those habitats are ready to provide their SIV factors to obtain better quality in other habitats. The objective is to improve the SIV of the habitats.

In the developing BBO algorithm, the maximum number of iterations and SIVs are initialized for the problem. The habitat set is initialized as a vector having a set of SIVs, which is considered as the solution vector. A habitat set is depicted as,

$$H = \begin{bmatrix} x_{11} & x_{11} & \dots & x_{1m} \\ x_{21} & x_{22} & \dots & x_{2m} \\ \dots & \dots & \dots & \dots \\ x_{n1} & x_{n2} & \dots & x_{nm} \end{bmatrix} \tag{13}$$

For each habitat, HSI is computed for given λ and μ rates. For the fitness function $f(x)$, HSI can be obtained as,

$$HSI_1 = f(x_{11}, x_{12}, \dots, x_{1m}) \quad (14a)$$

$$HSI_2 = f(x_{21}, x_{22}, \dots, x_{2m}) \quad (14b)$$

⋮

$$HSI_n = f(x_{n1}, x_{n2}, \dots, x_{nm}) \quad (14c)$$

Elite habitats are identified and kept untouched. The non-elite habitats undergo a migration process based on immigration rate λ and emigration rate μ . The probability for a habitat to be altered is computed according to the immigration rate, while the probability of the habitat that acts as a source for alteration is computed, according to the emigration rate. A habitat H_i is taken, and the migration process is performed as follows. A habitat H_i with probability proportional to λ_i , and a habitat H_j with a probability proportional to μ_j are selected. Then, a random SIV is taken from the habitat H_j , to replace the randomly selected SIV H_i . Then, HSI is computed for non-elite habitats. Then, the species count probability is updated for every habitat. The probability update equation for the time instant t to t is expressed in (15).

$$P_s(t + t) = P_s(t) (1 - \lambda_s \Delta t - \mu_s \Delta t) + P_s - \lambda_s - \Delta t + P_s + \mu_s + \Delta t \quad (15)$$

Subsequently, the mutation process is accomplished on each non-elite habitat and HSI is computed again. For all habitats, P_i is computed using (15) and SIV, having a probability proportional to P_i is selected. These processes are repeated several times, until the obtained solution is fit.

In the traditional BBO algorithm [21], the process of selecting the immigration habitat does not consider the amount of deviation existing between the selected habitat for immigration and the original habitat. This deviation must be significant in order to achieve the necessary improvement in solution at each migration process. If there is no imperative deviation between habitats involved in migration operations, the optimization process takes more time. In the FL-BBO algorithm, the immigration rate (λ_i) and emigration rate (μ_i) are computed based on the fuzzy logic-based migration control parameter. This control parameter is computed according to the difference between SIVs selected for the migration process. Two sets of SIVs

SIV_{ij} (in i^{th} row) and SIV_{kj} (in k^{th} row) are taken. Then, the difference between each element of the rows SIV_{ij} and SIV_{kj} is computed using Equation (16).

$$Diff_k = SIV_{ij} - SIV_{kj} \neq i \quad (16)$$

After computing the difference between SIVs, the change in difference is also computed. For computing the change in difference, $Diff_k$ parameters ($Diff_k$) computed in current iteration $iter$ and $Diff_k$ parameters computed in the previous iteration $iter - 1$ are considered. Therefore, $Diff_k$ can be computed using the Equation (17).

$$Diff_k = Diff_k(iter) - Diff_k(iter - 1) \quad (17)$$

Then, the difference parameter and change in difference parameter are applied to the fuzzy logic controller. The fuzzy controller analyses these parameters for high, low, and medium values. Based on these values, the control parameter for migration is obtained. The input values of difference and change in difference parameters are mapped into fuzzy groups. For the computed fuzzy sets, membership functions are formed, which fuzzify the input values into the relevant linguistic variables. Then, the migration control parameter (MC_k), which is the output of the fuzzy controller, is obtained using IF-THEN rules as follows.

If $Diff_k$ and $Diff_k$ are minimum, then MC_k is minimum; which is almost equal to 0.

If $Diff_k$ and $Diff_k$ are maximum, then MC_k is maximum; which is almost equal to 0.9.

If $Diff_k$ is minimum and $Diff_k(iter - 1)$ are maximum, then MC_k is moderately low 0.3.

If $Diff_k$ is maximum and $Diff_k(iter - 1)$ are minimum, then MC_k is moderately high 0.6.

Using the migration control parameters is given with the new immigration rate and emigration rate values are computed as provided in Equations (18a) and (18b).

$$\lambda_{kp} = MC_k I \left(1 - \frac{K(k)}{N} \right) \quad (18a)$$

$$\mu_{kp} = MC_k E \left(\frac{K(k)}{N} \right) \quad (18b)$$

The optimal route selection process using FL-BBO is illustrated in the algorithm 2.

The FL-BBO algorithm is used for selecting optimal routes. The links in the routes are taken as the habitats in the habitat matrix. It is considered that

Algorithm-2: FL-BBO for optimal route selection process

1. Take the links of network and give as habitat sets H_i input
2. Assign the initial parameters of BBO
3. for $i = 1$ to n
4. Compute HSI for each link set (H_i) using the fitness function given in Equation (10).
5. Select the H_i with worst HSI value
6. End
7. for $i = 1$ to n
8. for $j = 1$ to m
9. for $k = 1$ to n
10. While ($k \neq i$)
11. Compute difference between link parameters of SIVs
12. End
13. End
14. End
15. End
16. For $k = 1$ to n
17. Compute $Diff_k$ and $\Delta Diff_k$
18. Set min and max; mod_{low} and mod_{high}
19. If $Diff_k = \min$ and $\Delta Diff_k = \min$
20. Set $MC_k = \min$
21. Else if $Diff_k = \max$ and $\Delta Diff_k = \max$
22. Set $MC_k = \max$
23. Else if $Diff_k = \min$ and $\Delta Diff_k = \max$
24. Set $MC_k = mod_{low}$
25. Else if $Diff_k = \max$ and $\Delta Diff_k = \min$
26. Set $MC_k = mod_{high}$
27. Compute migration controlled immigration rate and emigration rate using (21) and (22)
28. end
29. for each habitat from $i = 1$ to n
30. for $k = 1$ to m
31. Select H_{ik} whose probability is proportional to λ_{kp}
32. end
33. For each habitat from $j = 1$ to m
34. for $k = 1$ to n
35. Select H_{kj} whose probability is proportional to μ_{kp}
36. Replace H_{ik} link with H_{ik} link
37. end
38. end
39. end

there are n number of nodes, $X = \{X_1, X_2, \dots, X_n\}$ and m number of links, $Y = \{Y_1, Y_2, \dots, Y_m\}$. The number of routes R and the set of routes are denoted as $r = \{r_1, r_2, \dots, r_R\}$. Based on the parameters of the link, the delay of each habitat set is computed using Equation (10). Each habitat set is considered as a route, and the habitat set with the minimum D_{path} is selected. If required fitness is not obtained, a migration process is performed, where the links of one habitat set are taken and moved to another habitat set. The migration process is done using a probabilistic immigration rate and a probabilistic emigration rate.

Table 5
Simulation settings

Parameters	Value
Number of nodes	20
Communication range	50m × 50m
Nodes initial energy	3J
Packet size	512 bits

4. Results

The performance of the newly proposed traffic-aware and delay-aware routing technique for IoT-based WSN is observed through simulation results. A simulation setup with different home appliances connected with sensor nodes is deployed. The performance is evaluated using MATLAB simulation software. An IoT system with 20 sensor nodes is arranged where the nodes are unevenly distributed. For the effective purpose of communicating data in the system, wireless links are used. The simulation settings used for evaluating the proposed work are provided in Table 2. To validate the proposed work, the comparative analysis is performed with the TEAR and ETASA methods. The network lifetime is observed for the proposed work and is depicted in Fig. 2. From the figure, it is noted that the nodes die after 3000 rounds and 4500 rounds in the TEAR technique and ETASA technique, respectively. In the TADEER method, traffic rate and data critical level are considered, and delay is set accordingly. Hence, less energy is consumed in the proposed work. The employment of the proposed technique increases the network lifetime where the nodes are alive even after 5000 rounds.

If criticality factor (CF) is high, the delay is set to be low. Hence, the chance for transmitting data having high CF is high. When CF of data increases, energy consumption in the network also increases. In other methods like TEAR and ETASA, only traffic rate is considered and the critical level of data is ignored. In the proposed TADEER method, the delay for data transmission is set based on its critical level. Therefore, the energy consumption is low, when data is less critical, as illustrated in the result graph in Fig. 4. When the critical factor increases, the energy consumption is high.

In the TEAR and ETASA methods, the delay is not assigned, according to the data critical level. In the proposed work, when the data criticality level is high, less delay is set. Delay fixed according to CF of data for different methods is shown in Fig. 5. Similarly,

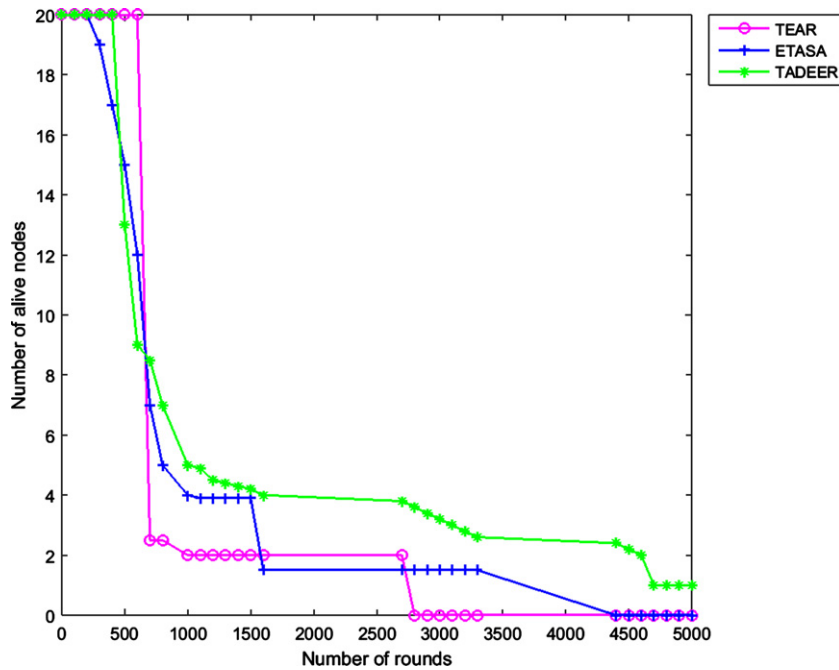


Fig. 5. Network lifetime in different methods.

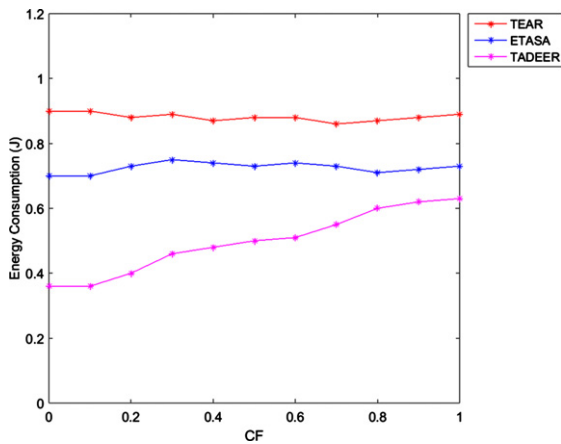


Fig. 6. Energy consumption vs Criticality factor.

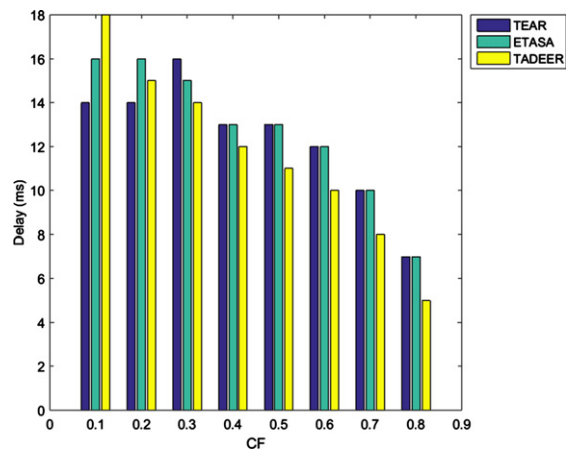


Fig. 7. Delay vs criticality factor.

delays are high, when traffic rate at the node is large. The delay for different traffic rates is given in Fig. 6.

Figure 7 shows energy consumption for different traffic rates. When the traffic rate is high, more energy is consumed in data transmission. In the TEAR and ETASA methods, energy consumption is decreased when the traffic rate increases. In the proposed work, along with the traffic rate, the critical level of data is considered. Based on these two factors, a delay is set for data transmission. For high traffic rate, more

delays are set. Thus, the energy consumed for data transmission is reduced.

5. Conclusion

In this work, TADEER scheme is presented for lifetime maximization in IoT system applications. The proposed routing algorithm enables to send the critical data in the network without any delay. In the

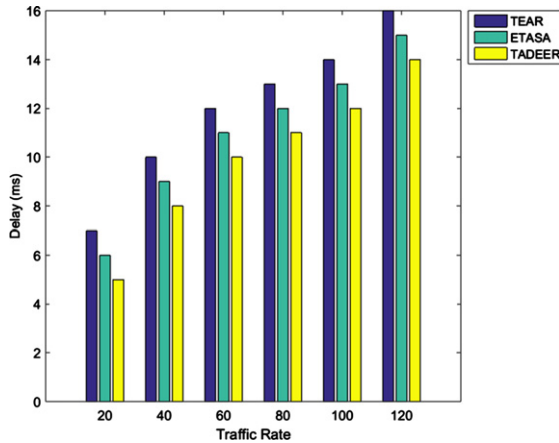


Fig. 8. Delay vs Traffic rate.

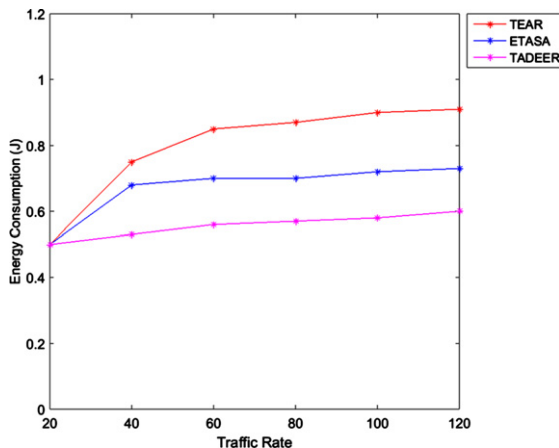


Fig. 9. Energy consumption vs Traffic rate.

proposed TADEER method, a delay is set for transmitting data, based on its critical level and traffic rate at the node. The routing algorithm helps in reducing the number of transmissions within the network. Moreover, the delay for critical data transmission is set to be low. Hence, the critical data are sent, without a miss. This ensures data transmission accuracy in the network. The proposed work is evaluated by simulating the network and observing metrics like energy consumption, delay, and network lifetime. The results show that for nodes with less traffic, less energy is consumed. It is also observed that when the data critical level is low, less energy is consumed. The proposed work gives better performance, when compared with the TEAR and ETASA methods. In the future, security concerns and methods for enhancing security in the proposed work can be presented.

Declaration

The author has no Conflict of interest to declare.

References

- [1] S. Pavalarajan and P. Thirumurugan, Prediction Based Location Estimation in Wireless Sensor Network, *Asian Journal of Information Technology* **15**(2) (2016), 191–198.
- [2] S. Pavalarajan and A. Prasanth, Implementation of Efficient Intra- and Inter-Zone Routing for Extending Network Consistency in Wireless Sensor Networks, *Journal of Circuits, Systems and Computers* **39**(6) (2019), 874–880. <https://doi.org/10.1142/S0218126620501297>
- [3] Alireza Souri, Nima Jafari Navimipour and Amir Masoud Rahmani, Formal verification approaches and standards in the cloud computing: a comprehensive and systematic review, *Computer Standards & Interfaces* **58** (2018), 1–22.
- [4] Jian Shen, et al., An efficient centroid-based routing protocol for energy management in WSN-assisted IoT, *IEEE Access* **5** (2017), 18469–18479.
- [5] Yang Liu, Boon-Chong Seet and Adnan Al-Anbuky, An ontology-based context model for wireless sensor network (WSN) management in the Internet of Things, *Journal of Sensor and Actuator Networks* **2**(4) (2013), 653–674.
- [6] Fatemeh Safara, et al., PriNergy: A priority-based energy-efficient routing method for IoT systems, *The Journal of Supercomputing* (2020), 1–18.
- [7] Amol V. Dhumane and Rajesh S. Prasad, Multi-objective fractional gravitational search algorithm for energy efficient routing in IoT, *Wireless Networks* **25**(1) (2019), 399–413.
- [8] S.K. Sathya Lakshmi Preeth, et al., An adaptive fuzzy rule based energy efficient clustering and immune-inspired routing protocol for WSN-assisted IoT system, *Journal of Ambient Intelligence and Humanized Computing* (2018), 1–13.
- [9] Yaning Wang and Zhaofeng Wang, Routing algorithm of energy efficient wireless sensor network based on partial energy level, *Cluster Computing* **22**(4) (2019), 8629–8638.
- [10] Muhammad Faheem and Vehbi C. Gungor, Energy efficient and QoS-aware routing protocol for wireless sensor network-based smart grid applications in the context of industry 4.0, *Applied Soft Computing* **68** (2018), 910–922.
- [11] Juan Luo, et al., Optimal energy strategy for node selection and data relay in WSN-based IoT, *Mobile Networks and Applications* **20**(2) (2015), 169–180.
- [12] Najem Naji, et al., Context-Aware Wireless Sensor Networks for Smart Building Energy Management System, *Information* **11**(11) (2020), 530.
- [13] Fahad Khan, et al., IoT Based Power Monitoring System for Smart Grid Applications, *2020 International Conference on Engineering and Emerging Technologies (ICEET)*, IEEE, 2020.
- [14] S. Sankar and P. Srinivasan, Fuzzy logic based energy aware routing protocol for Internet of Things, *International Journal of Intelligent Systems and Applications* **10**(10) (2018), 11.
- [15] Thien Duc Nguyen, Jamil Yusuf Khan and Duy Trong Ngo, A distributed energy-harvesting-aware routing algorithm for heterogeneous IoT networks, *IEEE Transactions on Green Communications and Networking* **2**(4) (2018), 1115–1127.

- [16] K. Thangaramya, et al., Energy aware cluster and neuro-fuzzy based routing algorithm for wireless sensor networks in IoT, *Computer Networks* **151** (2019), 211–223.
- [17] Muruganantham Arunraja, Veluchamy Malathi and Erulappan Sakthivel, Energy conservation in WSN through multilevel data reduction scheme, *Microprocessors and Microsystems* **39**(6) (2015), 348–357.
- [18] Muruganantham Arunraja, Veluchamy Malathi and Erulappan Sakthivel, Distributed similarity based clustering and compressed forwarding for wireless sensor networks, *ISA Transactions* **59** (2015), 180–192.
- [19] Nura Modi Shagari, et al., Heterogeneous energy and traffic aware sleep-awake cluster-based routing protocol for wireless sensor network, *IEEE Access* **8** (2020), 12232–12252.
- [20] V. Sutha Jebakumari, D. Shanthi Saravanan and D. Devaraj, Seizure detection in EGG signal with novel optimization algorithm for selecting optimal thresholded offset Gaussian feature, *Biomedical Signal Processing and Control* **56** (2020), 101708.
- [21] Dan Simon, Biogeography-based optimization, *IEEE Transactions on Evolutionary Computation* **12**(6) (2008), 702–713.
- [22] A.S. Sadiq, A.A. Dekhordi, S. Mirjalili, J. Too and P. Pillai, Trustworthy and efficient routing algorithm for IoT-FinTech applications using non-linear Lévy Brownian generalized normal distribution optimization, *IEEE Internet of Things Journal* (2021). DOI: 10.1109/JIOT.2021.3109075