International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2020/205318 Volume 7, Issue 6, November – December (2020)

RESEARCH ARTICLE

Fractional Gaussian Firefly Algorithm and Darwinian Chicken Swarm Optimization for IoT Multipath Fault-Tolerant Routing

Salem Abdulla Awadh Ba hmaid
Computer Science Department, Rathinam College of Arts and Science, Bharathiar University, Coimbatore, India.
salem100323@gmail.com

V. Vasanthi
Department of Information Technology, Sri Krishna Adithya College of Arts and Science, Bharathiar University, Coimbatore, India.
vasarthika@gmail.com

Published online: 25 December 2020

Abstract – Wireless Sensor Networks (WSN) based Internet-of-Things (IoT) systems offer high efficient data transmission with enhanced Quality of Service (QoS). A multi-constraint based energy-efficient and fault-tolerant routing algorithm using Fractional Gaussian Firefly Algorithm (FGFA) and Darwinian Chicken Swarm Optimization (DCSO) are presented for performing optimal multipath communication. FGFA is an improved Firefly Algorithm in which the fractional theory and Gaussian function are incorporated to improve the convergence speed with higher efficiency. Likewise, the DCSO is an improved model of CSO based on the survival theory of Darwin to decrease the computation time and improve the convergence by eliminating the local optimal challenges. Initially, the network is clustered and the cluster heads (CH) are chosen optimally by FGFA based on the objective function with multiple QoS constraints. Then the best routing paths are chosen by DCSO through similar objective function with inter-cluster and intra-cluster delay additionally included. The optimal paths are sorted in a hierarchical order from which multiple paths are utilized for data communication. The FGFA+DCSO routing protocol is assessed in NS-2 simulator and the outcomes shown the proficiency of the suggested approach with 6.3% reduced delay, 6% improved throughput, 26.7% minimized energy, 11% increased lifetime, 20% higher PSNR, and hop count reduced by 1.

Index Terms – Internet-of-Things, Wireless Sensor Networks, Fault Tolerance, Energy Constraint Problem, Fractional Gaussian Firefly algorithm, Darwinian Chicken Swarm Optimization.

1. INTRODUCTION

IoT is the blooming technology that utilizes objects or things into the Internet to provide communication that ensures different control and processing applications [1]. Integration of IoT does not require specialized infrastructure but only application-oriented and connectivity possible objects. The fundamental virtual layer of the IoT networks is the WSN which enables the wide-area sensing and communication applications [2]. WSN consists of a large number of autonomous and multi-hop sensor nodes which increases the efficiency of the IoT networks. WSN based IoT has greater applications in the fields of military operations, agriculture, medical, transport, environmental and educational organizations [3]. Along with the benefits of the WSN integration for IoT networks, the challenges in WSN are also incorporated into the new IoT networks [4]. The major challenges are the energy constraint and link failure problems in addition to other pivotal problems of delay, packet loss and less throughput. As in the WSN, these problems are mostly related to the routing protocol and this enlightens that the development of an efficient multi-constraint routing protocol [5] for WSN based IoT networks can be the optimal solution for these problems.

Traditional protocols are single-path routing protocols that rely on the shortest path selection process [6]. The shortest paths are not exactly the optimal path since they are dependent only on a single objective and not all shortest paths are efficient for routing. Hence the concept of multi-objective routing protocol was created which considers multi-constraints for selecting the efficient path for data transmission [7]. Also, the Cluster Head (CH) selection in clustered WSN must be resolved to ensure energy efficiency [8]. The other problem with the single path routing protocols is that they are prone to packet loss due to congestion or link failures when the nodes expire. Once the selected CHs fail, the faulty CH cannot transmit the sensed data and results in data loss. The fault tolerance in the network using single-path routing is greatly prone to this problem. Multipath routing protocols considerably resolve this CH fault problem through the selection of two-or-three fault-free CH for transmitting the data when faulty CH occurs, so that it reduces the overhead

RESEARCH ARTICLE

and possibility of link faults [9]. Developing multi-constraints based multipath routing protocol using advanced optimization algorithms for CH selection and route selection ought to be the main objective. Many studies have developed similar optimization based routing protocols, but their limitations have made way for newer algorithms.

Previously, the Fractional Firefly Algorithm and Chicken Swarm Optimization (FFA+CSO) [10] based routing protocol in which the FFA optimally selected the CH and CSO selected the optimal routing path based on multiple objectives. Although efficient, the FFA and CSO algorithms can be enhanced by improving their global search abilities. Through extensive analysis, some interesting strategies have been devised to improve these algorithms. Thus in this paper, the FGFA+DCSO based routing protocol is developed by improvising the strategies to improve the optimization algorithms. Using this routing model, the multipath routing is achieved in IoT networks based on multi-constraints. The NS-2 simulations are performed to evaluate the effectiveness of the suggested routing model.

2. RELATED WORK

Numerous optimization procedures have been exploited in the advancement of energy proficient routing algorithm for WSN and WSN based IoT. Lalwani and Das [11] utilized bacterial foraging optimization for CH selection to achieve optimal routing based on energy and distance parameters. Though efficient, this optimization algorithm also has limitations in the convergence rate. Hasan and Al-Turjman [12] developed a multipath fault-tolerant routing in IoT using the bio-inspired particle multi-swarm optimization which reduced energy and delay. However, this model did not utilize more objective parameters that influence the link failures. Haseeb et al. [13] proposed a dynamic energy-aware fault-tolerant routing algorithm using uniform network partitioning. Although it provided higher tolerance and reduced energy consumption, it has limitations in terms of delay. Lin et al. [14] proposed Bipartite-Flow Graph Modeling for providing fault-tolerant routing in IoT WSN. Muhammed et al. [15] designed a hierarchical clustered fault-tolerant routing model to improve the network lifetime. However it has the drawback of high energy wastage. Rui et al. [16] developed self-adaptive fault-tolerant routing using autonomous failure detection. This approach reduced the data loss and network failures but increased complexity.

Lalwani et al. [17] established an energy effective routing algorithm using the biogeography optimization in which the CH is selected based on energy while routing based on energy and distance. Despite providing high-performance routing, it has slow convergence. Preeth et al. [18] presented an adaptive fuzzy rule for CH selection and immune-inspired optimization for route selection in WSN-assisted IoT to decrease energy depletion and improve data delivery. Although significantly efficient than other routing models, the load balancing in the CHs is not effectively achieved.

Thangaramya et al. [19] developed an energy-aware neuro-fuzzy rule for CH selection and optimal routing in IoT networks. This approach provided path selection with smaller delay and minimum power depletion but it does consider all nodes are trusted nodes which is not practically possible. Vijayalakshmi and Anandan [20] presented multi constraint Tabu PSO for optimal CH selection and routing based on network lifetime and energy consumption. However, this hybrid model has a higher computation complexity than individual models. Mittal [21] utilized Moth flame optimization for energy effective CH detection and stabilized routing based on load balancing. This approach reduced the route link cost and energy consumption to ensure optimal communication but it has a high delay in data transmission due to the data fusion at the CH. Awan et al. [22] designed Gray Wolf Optimization (GWO) based routing optimization through CH selection with minimized routing cost. However, GWO is also limited by its low convergence speed. Vinitha and Rukmini [23] developed Taylor Series-based hybrid optimization of Cat Swarm and Salp Swarm Algorithms for CH and hop selection in secured multi-hop routing. This algorithm based on Taylor series improves the convergence and increases secured and energy-efficient routing performance. Still, this algorithm has performance degradation due to computation complexity when the network size increases.

Pattnaik and Sahu [24] employed fuzzy approach for CH selection based on residual energy, node centrality, and neighbourhood overlap whereas Elephant Herding Optimization (EHO) - Greedy algorithm for optimal route selection with reduced sink energy. However, the CHs elected were not able to handle massive data in dense networks and also increased the data forwarding delay. Kavitha et al. [25] also proposed cluster-based routing using Gravitational Search Algorithm for CH selection and optimal route selection based on energy, communication cost and lifetime. Still, this approach did not consider the link failures and it has degraded the overall efficiency. Vinodhini and Gomathy [26] developed a multi-objective dynamic routing model using k-means based node clustering and Artificial Bee Colony (ABC) system for optimum CH and path assortment. This mixture model reduced the communication cost and energy wastage but it has limitations of the local optimum problem in ABC. For the same objective, Kumar et al. [27] recommended fractional artificial bee colony + Exponential Ant Colony Optimization (FABC+EACO). Correspondingly, Dhumane and Prasad [28] used Fractional Gravitational Search Algorithm (FGSA) + Fractional Grey Wolf Optimization (FGWO). But these two studies have limitations of slow convergence. Thus, from the literature studies, it is understood that irrespective of the efficient performance of all

RESEARCH ARTICLE

the algorithms, the room for improvement is vast. Particularly, the local optimum problem and the corresponding slow convergence of optimization algorithms is one prominent area requiring improvements. Likewise, the routing techniques ensure energy efficiency and fault tolerance. However, fault-tolerant routing in IoT networks is more challenging to obtain than in the general WSN. Hence considering these limitations, the proposed study develops a routing model based on FGFA+DCSO that provides high performance multi-constraint multipath routing for the IoT networks.

3. PROPOSED FGFA-DCSO BASED ROUTING PROTOCOL METHODOLOGY

The proposed methodology focuses on selecting the optimal CH and best routing paths based on multiple constraints. To achieve this objective, the FGFA and DCSO algorithms are developed and used in the development of FGFA+DCSO routing model built on Ad hoc On-Demand Distance Vector (AODV) model. Figure 1 illustrates the processes involved in the proposed routing model.

3.1. System Model

IoT network is built on the WSN architecture and hence the internet-connected things are linked to the Base station (BS). The equipment in IoT performs as the sensing operations of nodes as per the application requirements. The BS can be placed inside the network or outside the network dimension depending upon the user requirements. The IoT nodes are equipped with the capability to act as CH as well as cluster member nodes to transmit packets to the BS and also shift their energy for this process. Each node collects information and transmits them to the BS over the CH which is designated based on energy, delay, link quality and lifespan. When one selected CH runs out of energy or attains a threshold energy level, the CH is replaced and this process repeats until the predefined minimum number of nodes is not present in the network. When considering a simulation area of R square meters with N nodes, the initial energy of each node is set as $P_i = P_0$. The proposed routing protocol consists of two main processes: CH selection and Routing path selection. FGFA performs CH selection while DCSO obtains the best routing paths for multipath data transmission. First, the node positions are stored in the cache table and similar nodes are clustered to form groups. The nodes are distributed as similar node groups to form clusters C_D and each cluster contains one CH. Each node is positioned at (x_i, y_i), while the location of BS is represented as (X_B, Y_B). The energy model and the mobility models are utilized as similar to that of FFA+CSO [10]. The transmission model is constructed upon the AODV for path detection and preservation. The transmission paths are formed once the CH is chosen, using the AODV principle. Then the DCSO selects the multiple optimal paths based on the objective parameters.

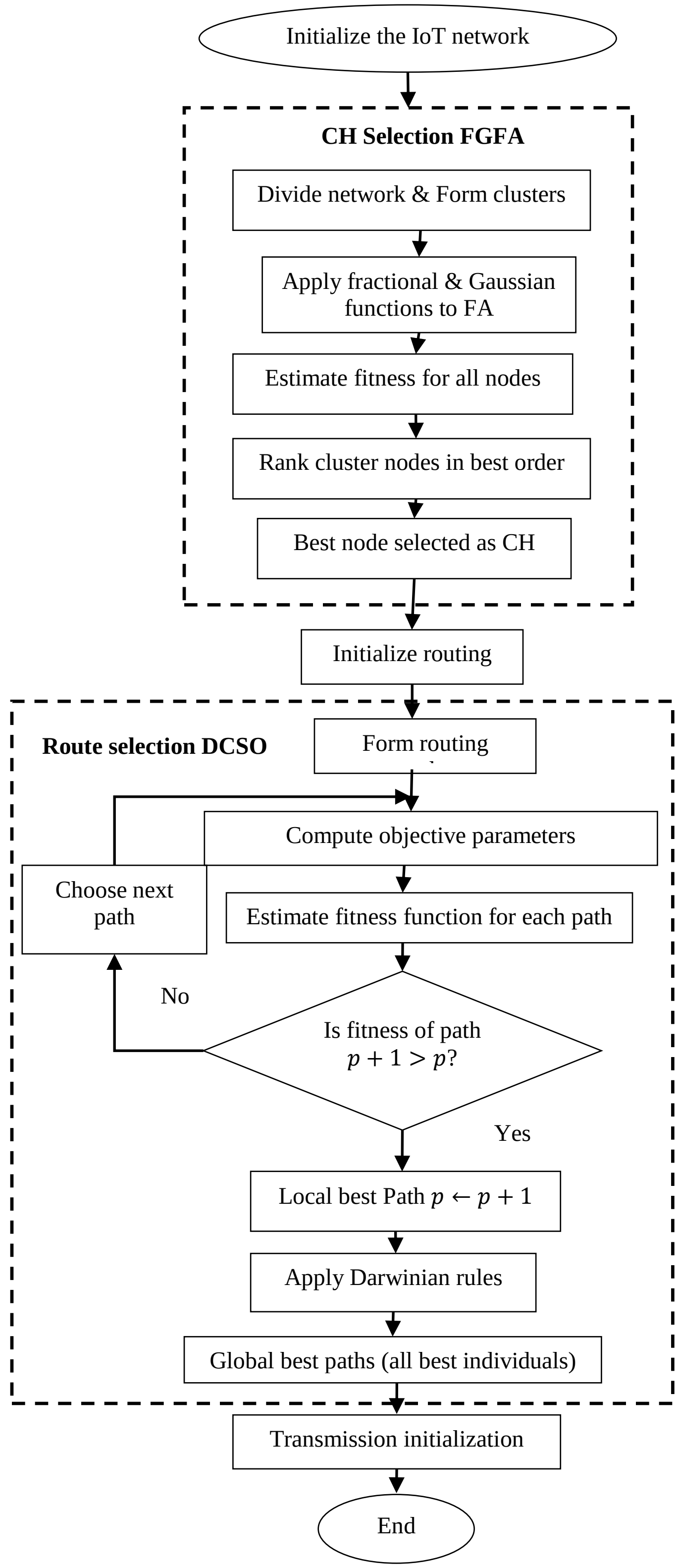

Figure 1 Proposed Workflow of FGFA+DCSO Based Routing Model

International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2020/205318 Volume 7, Issue 6, November – December (2020)

RESEARCH ARTICLE

3.2. Fractional Gaussian Firefly Algorithm (FGFA) For CH Selection

The FGFA algorithm is designed by integrating the fractional theory and Gaussian distribution function into the standard Firefly algorithm. Some studies have developed the fractional theory based optimization algorithms [27], [28] and Gaussian Firefly algorithm [29]. Based on such studies, the FGFA algorithm is designed. The main advantage of utilizing FGFA over standard FA is that it improves the global search ability of the algorithm and increases the chances of obtaining global optimum solutions. This eradicates the problem of algorithm getting stuck at local optima and reducing the convergence rate. This enhancement is greatly handy in improving the CH selection in fault-tolerant routing performance for IoT networks. A good CH must satisfy multiple QoS constraints and still provide effective data transmission to the BS. Hence the FGFA models a fitness function using four essential parameters for CH selection.

The fitness function thus contains metrics: fault tolerance through link quality metric, time through delay metric and energy efficiency through energy and lifetime. The fitness function formulated is given as follows:

$$F(x) = w_1 \times Energy + w_2 \times delay + w_3 \times Lifetime + w_4 \times Link\ quality \quad (1)$$

Here w_1, w_2, w_3 and w_4 are the weight values assigned to the four parameters computed for each node. The values for w_1, w_2, w_3 and w_4 are chosen such that $w_1, w_2, w_3, w_4 > 0$ and $w_1 + w_2 + w_3 + w_4 = 1$. The four objective parameters are calculated based on the equations used in FFA+CSO [10].

The FGFA selects the CH based on the higher fitness values and solves the Pareto-optimal problem caused by the multiple objectives. The energy and link quality metrics are given higher priority since they affect the efficiency and fault tolerance of the routing protocol. FGFA follows the general rules of FA and adjusts the residents of fractional fireflies' society. The fractional principle with the Gaussian utility is mapped to the regular FA. The fractional function is given by

$$n_x^y(l+1) = \gamma n_x^y(l) + \frac{1}{2}\gamma n_x^y(l-1) + c^i(l+1) \quad (2)$$

Where c is the fractional input coefficient, γ is the attractiveness coefficient ($\gamma \in [0,1]$), $n_x^y(l+1)$, $n_x^y(l)$ and $n_x^y(l-1)$ are the node locations of x with respect to y at $l+1$, l and $l-1$, respectively. Integrating this function, the FGFA resembles fractional optimization. Using the fitness function from Eq. (1), the fractional fireflies are categorised in descending order.

The fractional desirability ($F\beta$) and fractional light strength (FI) is specified as:

$$F\beta = \frac{1}{FT^\alpha}\beta_0 \exp(-\gamma r^{m+F\alpha}) \quad (3)$$

$$FI = D^\alpha(r).I_0 \exp(-S^\alpha \gamma r^2) \quad (4)$$

Here FT^α is a fractional interval, $F\alpha$ is the fractional repetitions in accumulation to m iterations, $D^\alpha(r)$ is the smallest distance of fireflies, α is the step regulatory constraint, S^α is the aggregate strength index, r is the distance assessed amongst any two fireflies (nodes) and β_0 is the power variation at r=0.

The position of fractional fireflies is updated based on

$$Fx_i(m+1) = x_i(m) + \frac{1}{2}\alpha_F \beta_0 e^{-\gamma r_{i,j}^2}(x_j - x_i) + \alpha\epsilon \quad (5)$$

Here ϵ is the Gaussian distribution trajectory, α_F is the fractional constraint to regulate the phase size of the fireflies, $x_i(m)$ point of node i and $r_{i,j}^2$ is the square of the distance between two nodes (i, j).

The firefly movement is dependent on the value of r which means the smaller r value makes the firefly move in the small distance and vice versa. This will have a greater impact on the computation period and also makes the agents move in a fixed distance to match the firefly. But it might make the firefly lose the contact of agents and roams in the local search space providing only looped local optimum solutions. Thus a random step length is used for firefly movement where the firefly initially searches the feature space comprehensively and provided an effective global solution that is adaptive to all phase periods to overcome the local optimum problem. The weight of consecutive arbitrary step length α is evaluated by the following equation. Its value must always be closer to unity and is bounded by the values of maximum iteration m_{max} and current iteration m.

$$W_m = A + \frac{(m_{max}-m)^g}{(m_{max})^g} + (A-B) \quad (6)$$

where A=0 & B=1 as $\alpha \in [0,1]$, W_m is the weight amongst A & B and its value decreases by the time. g would be a linear or non-linear co-efficient and it is influenced by the capacity of every monitoring individual. If the capacity is big, the assessment of g is small which means the algorithm can converge more accurate. Its value is determined by

$$g = 10^{(-dimension)} \quad (7)$$

If α adapts Gaussian distribution, then the random walk movement becomes the Brownian motion. After each m, the standard Gaussian distribution is presented in directive to transfer all of the fireflies to overall best and is revealed in the following equation.

$$p = f(e|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\left(\frac{-|x-\mu|^2}{2\sigma^2}\right)} \quad (8)$$

Where μ and σ^2 are its mean, variance and e is the variation between the obtained fitness value and best solution of firefly i.

RESEARCH ARTICLE

$$e = f(best) - f(e_i) \quad (9)$$

Because of the use of the standard normal distribution, $\mu=0$ and $\sigma^2 = 1$. Then from this Gaussian distribution, a random number is chosen which is related to the swarm of each agent probability (p). The behaviour of the agent is then presented to the FGFA by modifying equation (5) as given below

$$Fx_i(m+1) = x_i(m) + \frac{1}{2}\alpha_F\beta_0 e^{-\gamma r_{i,j}^2}(x_j - x_i) + \alpha \times (1 - p) \times U(x,z) \quad (10)$$

Where $U(x,z) \in [0,1]$ is an arbitrary integer to increase the likelihood. If the evaluated new solution cost $Fx_i(m+1)$ better than the current position $x_i(m)$, then the firefly will move towards that new position. The apprising procedure is performed at the end of each search operation. The assessment of the fitness function, attractiveness and the location changing of firefly towards the best firefly is obtained based on a calculation by Eq. (1), (3) and (10) respectively. Thereby the procedure of firefly updating is iterated until the result is satisfied. The best firefly returned is the CH node with higher fitness valuation. Algorithm 1 shows the CH assortment procedure using the above established FGFA.

Begin

Initialize N fraction firefly population as n nodes

m=1;

Choose two nodes i and j as first set fractional fireflies

Map the locations of nodes to the cache table

While ($m < m_{max}$)

    If (Lifetime of i-th firefly < Lifetime of a j-th firefly)

    Allocate the firefly in new location stochastically

    Appraise the firefly solution list

    End if

Estimate the point of i, j from the cache table

Assessment of the cost-utility for fireflies

For i=1 to n (all n fireflies)

    For j=1 to n (all n fireflies)

    Evaluate fractional function based generation

    Compute the multiple objective parameters

    Evaluate fitness using Eq. (1)

    If ($F_j > F_i$)

        Move i towards j;

    Else

        Retain i and remove j;

        Elect i+1 node;

    End if

    Arrange the fractional fireflies

    Apprise the node's positions (fractional firefly update) through Eq. (10)

    Control unrestrained node activities using $F\beta$ (Eq. (3))

    Appraise the solutions list

    End for j

End for i

Re-Rank the fireflies and define the present best

End while

m= m+1;

Return CH

End

Algorithm 1: FGFA Based CH Selection

When all the nearest available CHs run out of energy or removed due to malfunctions, the cluster must select a new CH to function properly. In some cases, the CH might be moved away from the cluster range for network redesigning, there is a chance that there is no appropriate CHs to cover the entire cluster. To tackle such scenarios, a backup process is predetermined, i.e. the outgoing CH has all access to the latest information about the position of the destination. Using this information, the CH has maintained an updated list of best CH candidates and when leaving the cluster, the outgoing CH broadcasts a message to all its neighbours. Then the cluster is fragmented to form cluster fragments. Each similar neighbour nodes of the cluster are placed in the same cluster fragment while other nodes are transferred to nearby cluster fragments. Then the new CHs are selected based on the list as per the initial CH selection using FGFA.

3.3. Darwinian Chicken Swarm Optimization (DCSO) for Selecting Optimal Paths

DCSO is developed based on the implementation of Darwin's survival theory of living organisms into the standard CSO algorithm. This integration is modelled into an optimization problem similar to the Darwinian Particle Swarm optimization [30] and solved using the CSO metrics. Darwin's theory states that the fittest organism survives the adverse conditions of this planet. Although this theory is partially disowned by scientists, it has a significant impact on the selected survival environments. Using this theory to Cso

RESEARCH ARTICLE

forms the hierarchical survival strategy and enhances the solution searching capability.

First, the DCSO initializes the chicken population with RX, HX, CX and MX as rooster, hen, chick and mother hens. These hens are assigned with the routing paths. The best fault-tolerant paths are the results of this DCSO algorithm. They are obtained based on the fitness model that comprises energy, inter-cluster and intra-cluster delay, link quality, hop count and lifetime parameters. Similar to the CH selection fitness function, the weight values are assigned and the fitness $F_p(x)$ is given as

$$F_p(x) = w_1 \times Energy + w_2 \times P_d + w_3 \times Lifetime + w_4 \times Link\ quality + w_5 \times Hop \quad (11)$$

Here w_1, w_2, w_3, w_4 and w_5 are the weight values assigned to objective parameters computed for each available routes. P_d is the path delay computed as the sum of intra-cluster delay and inter-cluster delay.

Based on this fitness function, the chickens (paths) are evaluated. The chicken with the highest fitness value is chosen as the leader (rooster). The co-leaders (hens) and elders (mother hens) are cumulatively ordered while the members (chicks) are usually the lowest fitness routes. The chicken swarm is reordered hierarchically based on the fitness values and the positions are modified during operation. The movement and the position of the leaders can be updated as

$$X_{i,j}^{T+1} = X_{i,j}^{T} \times (1 + rand(0, \sigma^2) \quad (12)$$

Where $rand(0, \sigma^2)$ is a Gaussian distribution with zero mean and standard deviation σ^2, $X_{i,j}^{T}$ and $X_{i,j}^{T+1}$ are the points of present best and next best chicken, respectively.

The movement and position of the co-leaders and elders is updated as

$$X_{i,j}^{T+1} = X_{i,j}^{T} + S_1 \times rand \times \left(X_{r1,j}^{T} - X_{i,j}^{T}\right) + S_2 \times rand \times \left(X_{r2,j}^{T} - X_{i,j}^{T}\right) \quad (13)$$

Where $rand \in [0,1]$ is a constant arbitrary integer, S_1, S_2 are the community graded coefficient of chickens and $r1, r2 \in [1, \dots, N]$ are the index values of the chicken flock and $r1 \neq r2$. S_1 and S_2 are premeditated as in the regular CSO.

The movement and position of the members is updated as

$$X_{i,j}^{T+1} = X_{i,j}^{T} + fl \times (X_{m,j}^{T} - X_{i,j}^{T}) \quad (14)$$

Here $X_{m,j}^{T}$ is the location of the j-th chick's mother hen and $fl \in [0,2]$ is the flocking constraint.

The survival theory of Darwin has been implemented after this update stage. In this process, the swarms and new chicks are created by evolution and their fitness is evaluated. The labelled *Main Program Loop* and the *Evolve Swarm* processes are performed as activation at each step of the routing algorithm. The flock is allowed to perform the creation of new flock swarm with constant likelihood and a minimum number of individuals. Once the new flocks are created, the flocks with increasing fitness are selected while those with decreasing fitness are eliminated. The fitness estimation leads to the evolution of new individuals in the flock to choose the best optimal solutions. Then based on the best fitness, the location of the individuals is updated. The new individuals are created continually only when the swarm records the best fitness. If not, the current individual with the lowest fitness is eliminated in m iterations. The creation and elimination of the flock and the individuals are performed by the following steps.

3.3.1. Forming New Flocks and Individuals

A new flock can be created only when it has null eliminated individuals (i.e. $N_{kill} = 0$) and the number of flocks is not maximum. In this manner, new flocks can be created with the least number of individuals until maximum flocks are reached. After the maximum flocks are formed, new flocks cannot be formed openly. It can be formed only through the elimination of existing flocks and forming new flock to replace them i.e. it must satisfy $N_{kill} = 0$ to form new flocks. The newly formed flocks will have the likelihood of $p(f/N_s)$ with the constant arbitrary integer $f \in [0,1]$ and quantity of flocks N_s. $(1/N_s)$ in this likelihood function has been employed to limit the flock formation process when it reaches around the maximum number. The newly created flock imitates the characteristics of two-parent flocks selected to mutate them, resulting in the new flock. But the parent flocks retain their characteristics even after the formation of a new flock. The ratio of the characteristic of the two parents is equally captured by the new flock, yet they might choose the speciality of one parent flock to apply the DCSO design enhancement. The new flock formation is halted until the recently formed flock achieves global best fitness.

3.3.2. Discarding Flocks

There is a pre-defined lower threshold for the population of individuals in a flock so that it achieves at least reasonable fitness. If a flock eliminates individuals and reaches below this threshold, then the flock can be discarded in favour of a new flock. i.e. A flock population X must lie between $X_{min} \leq X \leq X_{max}$. When $X < X_{min}$, the flock faces elimination from the search space.

3.3.3. Discarding Individuals

Similar to the flock elimination, the individuals in the flock are also discarded when they achieve continually worst fitness. A search counter (SC) is set for monitoring this continual fitness drop. During the formation of a flock, this SC is set as zero. With increasing iterations without an

RESEARCH ARTICLE

improvement in the fitness of an individual in the flock, the SC value gets incremented by 1. The maximum value is set as the threshold SC_c^{max}, on reaching this value, the corresponding individual is discarded from the flock. Once elimination occurs, the counter is readjusted to a lesser value near the threshold. The reason for not resetting the value of SC to zero is that it will only degrade the delay tolerance. Hence the readjustment is made and the next individual is monitored. The readjustment value of SC is not random and it is computed using the N_{kill} value.

$$SC_c(N_{kill}) = SC_c^{max}\left[1 - \frac{1}{N_{kill}+1}\right] \quad (15)$$

Using these SC values, the flocks are formed and discarded to achieve optimal routing path solutions. The efficient selection of best routing paths is obtained in the order of leader, co-leaders, elders and member individuals. The entire procedure of DCSO is précised in Algorithm 2.

```
Begin
Route discovery through RREQ packet broadcasting
Set initial X individuals
Assign required metrics for each individual
Map the objective routing paths over the individuals
Set m=0; SC = 0;
For all individuals
    Assess the fitness through Eq. (11)
    While (m < m_max)
     If (m% G == 0)
        Dispense the flock into numerous sets of solution paths
        Perform fitness comparisons
        Move individuals towards global best
        Arrange individuals in descending fitness values
        Establish hierarchical order besides relations
        Equate the routes i, i+1;
        Re-rank the routes as numerous leaders and co-leaders
        Until finish
    End if
    For i = 1 : N
        If i == leader, then
            Update location via Eq. (12)
        End if
        If i == co-leader or elder, then
            Update location via Eq. (13)
        End if
        If i == member, then
            Update location via Eq. (14)
        End if
        Appraise the new best routes;
        If the new route > previous route
            If N_kill = 0 and N_s < N_s^max then
            Construct new swarm from parents
            Create a minimum chicken population
        Else
            Update search counter by one;
         If SC = SC_c^max then
            Delete the chicken
            Increment N_kill by one
         End if
         End if
       End if
    End for
End while
End for
Return multiple-path list
Allocate each leader/co-leader for multi-path transmission
End
```

Algorithm 2: DCSO Based Route Selection

4. RESULTS AND DISCUSSIONS

Simulations and performance evaluations of the suggested FGFA+DCSO protocol is achieved using the NS-2 simulator based on the settings given in Table 1.

Simulator	NS-2.34
No. of Nodes	100
Area Size	1000 X 1000 m
Channel type	Wireless Channel
Propagation model	Two Ray Ground
Link Layer	LL
Antenna model	Omni Antenna

RESEARCH ARTICLE

Traffic type	CBR
Mobility model	Random Waypoint
MAC	802.11
Initial energy	100 Joules
Radio Range	250m
Simulation Time	300 seconds
Number of packets	1000
Packet rate	8 packets/sec
Data payload	512 bytes/packet

Table 1 Simulation Environment

The evaluated performance of FGFA+DCSO protocol is compared with existing routing models that used optimization algorithms for CH selection and route optimization. FABC+EACO [27], FGSA+FGWO [28], standard FA+CSO and FFA+CSO [10] are evaluated for performance comparison. Before the complete routing comparison, the CH selection process of each of these methods is compared in terms of time as shown in Table 2.

CH Selection Algorithms	Time (ms)
FABC	0.2833
FGSA	0.2621
FA	0.1785
FFA	0.1540
FGFA	0.1487

Table 2 CH Selection Time Comparison

Table 2 shows that the CH selection time taken by the proposed FFA is very less than the other compared algorithms. This reduced time for CH selection is due to the integration of fractional Gaussian theory to the standard FA.

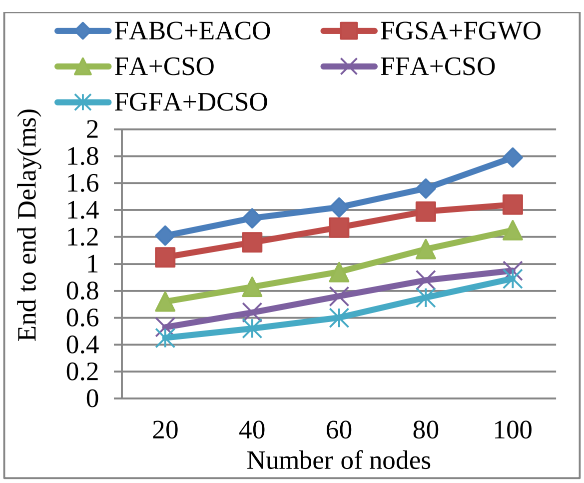

Figure 2 End-to-End Delay Comparisons

Figure 2 validates the delay comparison of FABC+EACO, FGSA+FGWO, FA+CSO, and the suggested FGFA+DCSO routing conventions. The proposed FGFA+DCSO have outperformed other models with better delay reduction due to the faster convergence rate and reduced retransmissions because of node failures. FGFA+DCSO has a delay of 0.89 milliseconds for 100 nodes, which is 6.3%, 28.8%, 38% and 50.2% smaller than that of FFA+CSO, FA+CSO, FGSA+FGWO and FABC+EACO, respectively.

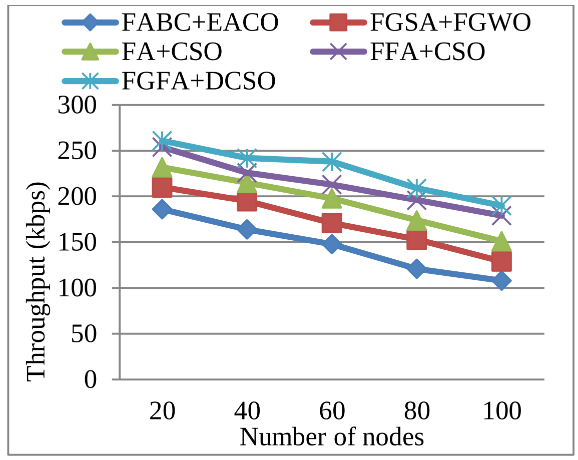

Figure 3 Throughput Comparisons

Figure 3 shows the throughput comparison of the proposed FGFA+DCSO and existing routing models. The proposed FGFA+DCSO have outperformed other models with improved throughput which is because of the reduced failures and packet loss. FGFA+DCSO has significant improvement due to the fast convergent route selection which resulted in 190dB throughput for 100 nodes, which is 6%, 25.8%, 47% and 75.9% higher throughput than that of FFA+CSO, FA+CSO, FGSA+FGWO and FABC+EACO protocols, respectively.

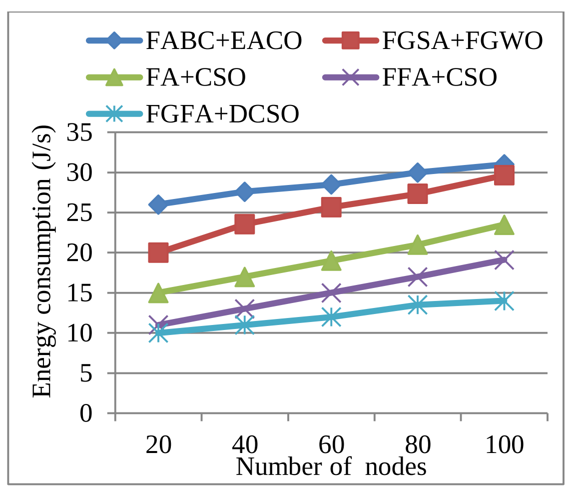

Figure 4 Energy Consumption Comparisons

RESEARCH ARTICLE

Figure 4 shows the energy comparison of the proposed FGFA+DCSO and existing routing models. It is displayed that the proposed FGFA+DCSO has outperformed other models because of the increased fractional theory efficiency and global solution selection in both FGFA and DCSO. FGFA+DCSO has consumed 14J energy for 100 nodes, which is 26.7%, 40.4%, 52.7% and 54.8% lesser energy than FFA+CSO, FA+CSO, FGSA+FGWO and FABC+EACO protocols, respectively.

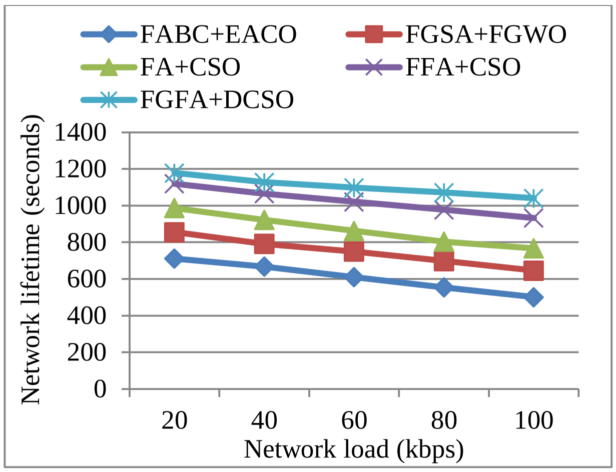

Figure 5 Network Lifetime

Figure 5 shows the comparison of the lifetime between the proposed FGFA+DCSO and the existing routing models. It is observed that the proposed FGFA+DCSO has increased lifetime because of the less energy wastage in CH selection and node failures. FGFA+DCSO has 1040 seconds lifetime, which is 11%, 35.6%, 60.1% and 107% higher than compared FFA+CSO, FA+CSO, FGSA+FGWO and FABC+EACO protocols, respectively.

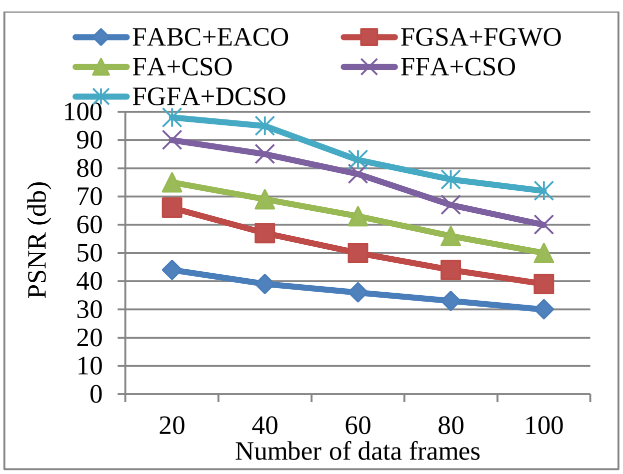

Figure 6 PSNR Comparisons

Figure 6 displays the PSNR comparison of the proposed FGFA+DCSO and the existing routing models. FGFA+DCSO have outperformed other models, which is mainly due to the high convergence rate and global best solution with minimum delay. FGFA+DCSO has PSNR of 72dB, which is 20%, 44%, 84.6% and 140% higher PSNR than FFA+CSO, FA+CSO, FGSA+FGWO and FABC+EACO protocols, respectively.

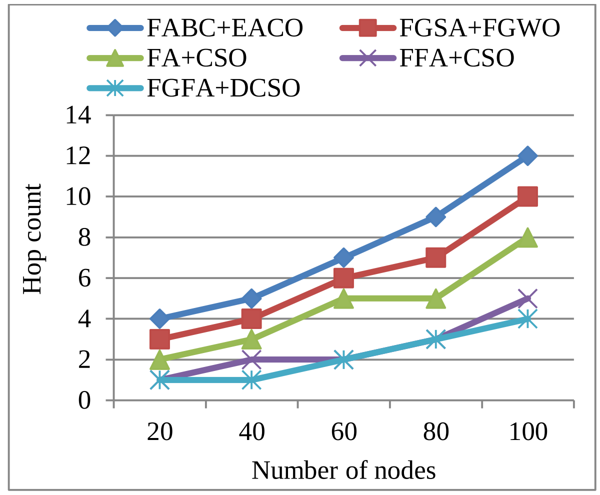

Figure 7 Hop Count Comparisons

Figure 7 indicates the hop count comparison of the proposed FFA+CSO against the existing routing models. FGFA+DCSO have outperformed other models with fewer hops. It is because of the selection of best routes which is optimal in terms of multiple constraints. For 100 nodes, FGFA+DCSO has taken only 4 hops while FFA+CSO has 5, FA+CSO has 8, FGSA+FGWO has 10 and FABC+EACO has 12 hops. Thus, the proposed FGFA+DCSO model has taken less number of hops and has also minimized the overhead considerably.

The superiority of the FGFA+DCSO routing model is attributed to the improved convergence rate and efficient global solution determination through the fractional and Gaussian function in CH selection and Darwinian theory in route optimization. These integrations have improved the global search abilities of the optimization algorithms and enhanced the overall routing performance in IoT networks.

5. CONCLUSION

Energy efficiency and fault tolerance are vital factors in any routing protocol. This paper has studied the energy-efficient routing algorithms and suggested the development of FGFA+DCSO algorithms for cumulative allotment of CH and optimal routing paths in WSN based IoT networks. FGFA+DCSO routing protocol was designed by enhancing the standard FA and CSO algorithms through efficient strategies to improve their global search proficiency. The efficiency of the FGFA+DCSO model has been evaluated and it showed that it has outperformed other similar routing models with 6.3% reduced delay, 6% improved throughput, 26.7% minimized energy, 11% increased lifetime, 20% higher

RESEARCH ARTICLE

PSNR, and hop count reduced by 1. In future, the possibility of including other fitness parameters such as packet delivery ratio, degree of load imbalance, etc. will be inspected. Likewise, the efficiency of utilizing the strategies like node sleep scheduling, error correction schemes for security improvement will also be investigated. More importantly, with the recent advancements in nature-inspired optimization algorithms, the possibility of developing hybrid optimization algorithms for efficient routing models in IoT networks will also be explored.

REFERENCES

[1] J. Gubbi, R. Buyya, S. Marusic and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," Future generation computer systems, vol. 29, no. 7, pp. 1645-1660, 2013.
[2] S. Park, N. Crespi, H. Park and S. H. Kim, "IoT routing architecture with autonomous systems of things," In Proc. 2014 IEEE World Forum on Internet of Things (WF-IoT), pp. 442-445, March 2014.
[3] R. Fantacci, T. Pecorella, R. Viti and C. Carlini, "A network architecture solution for efficient IoT WSN backhauling: challenges and opportunities," IEEE Wireless Communications, vol. 21, no. 4, pp. 113-119, 2014.
[4] K. Tejasvit, "Challenges in integrating wireless sensor networks into the internet," International Journal of Engineering and Management Sciences, vol. 5, no. 1, pp. 7-11, 2014.
[5] P. Rajpoot, S. H. Singh, R. Verma, K. Dubey, S. K. Pandey and S. Verma, "Multi-factor-Based Energy-Efficient Clustering and Routing Algorithm for WSN," In Soft Computing: Theories and Applications, Springer, Singapore, pp. 571-581, 2020.
[6] S. Mohapatra and P. Kanungo, "Performance analysis of AODV, DSR, OLSR and DSDV routing protocols using NS2 Simulator," Procedia Engineering, vol. 30, pp. 69-76, 2012.
[7] Z. Fei, B. Li, S. Yang, C. Xing, H. Chen and L. Hanzo, "A survey of multi-objective optimization in wireless sensor networks: Metrics, algorithms, and open problems," IEEE Communications Surveys and Tutorials, vol. 19, no. 1, pp. 550-586, 2016.
[8] M. Elhoseny and A. E. Hassanien, "Optimizing cluster head selection in WSN to prolong its existence," In Dynamic Wireless Sensor Networks, Springer, Cham, pp. 93-111, 2019.
[9] M. Radi, B. Dezfouli, K. A. Bakar and M. Lee, "Multipath routing in wireless sensor networks: survey and research challenges," Sensors, vol. 12, no. 1, pp. 650-685, 2012.
[10] S. S. A. B. Hmaid and V. Vasanthi, "Multipath Data Transmission in IoT Networks Using Fractional Firefly Algorithm and Chicken Swarm Optimization," International Journal of Intelligent Engineering and Systems, vol. 13, no. 3, pp. 373-383, 2020.
[11] P. Lalwani and S. Das, "Bacterial foraging optimization algorithm for CH selection and routing in wireless sensor networks," In Proc. 2016 3rd international conference on recent advances in information technology (RAIT), IEEE, pp. 95-100, March 2016.
[12] M. Z. Hasan and F. Al-Turjman, "Optimizing multipath routing with guaranteed fault tolerance in Internet of Things," IEEE Sensors Journal, vol. 17, no. 19, pp. 6463-6473, 2017.
[13] K. Haseeb, K. A. Bakar, A. H. Abdullah, A. Ahmed, T. Darwish and F. Ullah, "A dynamic Energy-aware fault-tolerant routing protocol for wireless sensor networks," Computers & Electrical Engineering, vol. 56, no. 1, pp. 557-575, 2016.
[14] J. W. Lin, P. R. Chelliah, M. C. Hsu and J. X. Hou, "Efficient fault-tolerant routing in IoT wireless sensor networks based on bipartite-flow graph modelling," IEEE Access, vol. 7, no. 1, pp. 14022-14034, 2019.
[15] T. Muhammed, R. Mehmood, A. Albeshri and A. Alzahrani, "HCDSR: A hierarchical clustered fault-tolerant routing technique for IoT-based smart societies," In Smart Infrastructure and Applications, Springer, Cham, pp. 609-628, 2020.
[16] L. Rui, X. Wang, Y. Zhang, X. Wang and X. Qiu, "A self-adaptive and fault-tolerant routing algorithm for wireless sensor networks in microgrids," Future Generation Computer Systems, vol. 100, no. 1, pp. 35-45, 2019.
[17] P. Lalwani, H. Banka and C. Kumar, "BERA: a biogeography-based energy-saving routing architecture for wireless sensor networks," Soft Computing, vol. 22, no. 5, pp. 1651-1667, 2018.
[18] S. S. L. Preeth, R. Dhanalakshmi, R. Kumar and P. M. Shakeel, "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, pp. 1-13, 2018.
[19] K. Thangaramya, K. Kulothungan, R. Logambigai, M. Selvi, S. Ganapathy and A. Kannan, "Energy-aware cluster and neuro-fuzzy based routing algorithm for wireless sensor networks in IoT," Computer Networks, vol. 151, pp. 211-223, 2019.
[20] K. Vijayalakshmi and P. Anandan, "A multi-objective Tabu particle swarm optimization for effective cluster head selection in WSN," Cluster computing, vol. 22, no. 5, pp. 12275-12282, 2019.
[21] N. Mittal, "Moth flame optimization-based energy efficient stable clustered routing approach for wireless sensor networks," Wireless Personal Communications, vol. 104, no. 2, pp. 677-694, 2019.
[22] K. M. Awan, H. H. R. Sherazi, A. Ali, R. Iqbal, Z. A. Khan and M. Mukherjee, "Energy- aware cluster- based routing optimization for WSNs in the livestock industry," Transactions on Emerging Telecommunications Technologies, p. e3816, 2019.
[23] A. Vinitha and M. S. S. Rukmini, "Secure and energy-aware multi-hop routing protocol in WSN using Taylor-based hybrid optimization algorithm," Journal of King Saud University-Computer and Information Sciences, In Press, 2019.
[24] S. Pattnaik and P. K. Sahu, "Assimilation of fuzzy clustering approach and EHO- Greedy algorithm for efficient routing in WSN," International Journal of Communication Systems, p. e4354, 2020.
[25] A. Kavitha, K. Guravaiah and R. L. Velusamy, "A Cluster-Based Routing Strategy Using Gravitational Search Algorithm for WSN," Journal of Computing Science and Engineering, vol. 14, no. 1, pp. 26-39, 2020.
[26] R. Vinodhini and C. Gomathy, "MOMHR: A Dynamic Multi-hop Routing Protocol for WSN Using Heuristic Based Multi-objective Function," Wireless Personal Communications, vol. 111, no. 2, pp. 883-907, 2020.
[27] R. Kumar, D. Kumar and D. Kumar, "Exponential Ant Colony Optimization and Fractional Artificial Bee Colony to Multi-Path Data Transmission in Wireless Sensor Networks," IET Communications, vol. 11, no. 4, pp. 522-530, 2017.
[28] A. V. Dhumane and R. S. Prasad, "Fractional Gravitational Grey Wolf Optimization to Multi-Path Data Transmission in IoT," Wireless Personal Communications, vol. 102, no. 1, pp. 411-436, 2018.
[29] S. M. Farahani, A. A. Abshouri, B. Nasiri and M. R. Meybodi, "A Gaussian firefly algorithm," International Journal of Machine Learning and Computing, vol. 1, no. 5, pp. 448-453, 2011.
[30] J. Tillett, T. Rao, F. Sahin and R. Rao, "Darwinian particle swarm optimization," In Proc. 2nd Indian International Conference on Artificial Intelligence, Pune, India, 2005.

Authors

***Mr.*Salem Abdulla Awadh Ba hmaid** received his BCA from St. Philomena's College, Mysore, and M.IT from Dr G.R. Damodaran College of Science, Coimbatore. Presently he is pursuing his PhD in the field of Advance networking at Rathinam College of Arts and Science, Bharathiar University, Coimbatore, India. His research interests focus on Computer Networks and IoT.

RESEARCH ARTICLE

Dr.V.Vasanthi pursed M.Sc (CS), M.Phil. And PhD. Computer Science from Karpagam University, Coimbatore in 2014. She is currently working as Professor in the Department of Computer Science from the Department of Information Technology, Sri Krishna Adithya College of Arts And Science, Bharathiar University, Coimbatore. She has published more research papers in reputed international journals including Thomson Reuters (SCI & impact factor) and conferences including IEEE & Springer and it's also available online. Her main research work focuses on Ad-hoc and Sensor Networking.