DOI: 10.21917/ijct.2019.0320

ENERGY EFFICIENT SINK PLACEMENTS WITH MOBILE DATA AGGREGATION IN WIRELESS SENSOR NETWORK

S. Praveen Kumar[1] and V. Rajanesh[2]
[1]*Department of Electronics and Communication Engineering, JNTUH College of Engineering, Jagtail, India*
[2]*Department of Electronics and Communication Engineering, JNTUH College of Engineering, Sultanpur, India*

Abstract

The modelling of wireless sensor networks has always given priority to energy efficiency. The implementation of mobile agent technology in integrated signal and wireless sensor networks has created an effective platform for data storage and aggregation. The distributed paradigm based on mobile agents provides many advantages over the current, widely used device paradigm for clients / data centres in wireless sensor networks. The agent-based paradigms are one of the major problems with mobile agent-based paradigms. Many sinks mitigate the problem by distributing traffic across several discharges, and reducing energy consumption at nodes and extending network service life. This paper presents an Energy-efficient multiple sink positioning algorithm to maximize average network life and limit average sensor network energy consumption. Several sinks are used here for data communication. Result of the proposed algorithm maximizes an average network life and reduces the energy consumption average. The results of the simulation shows that the Protocol proposed exceeds the end-to - end delay, energy consumption and output.

Keywords.

WSN, Distributed COMPUTING Paradigm, Mobile Agent, Multiple Sinks, Average Energy Consumption, Average Network Lifetime

1. INTRODUCTION

Recent developments and progress have opened the stage for Wireless Sensor (WSN) networks in the area of microelectric and wireless communications. Due to its ability to transform the way people and the physical world communicate with each other, WSN became recognized as a significant field of study. WSN has proved to be an area of enormous interest in the last few years. It consists of a variety of small sensor nodes with minimal computing resources. WSN is a basic network with low infrastructure. The sensor nodes are densely used in WSN for monitoring factors such as temperature, moisture, pressure, etc. under a variety of environmental conditions.

Dense and distributed systems that could perform complex tasks and inferences are a prodigious replacement for conventional, centralized architectures. A DSN was a set of amount of homogeneous or heterogeneous sensor nodes which are physically, spatially or geographically distributed and interlinked [3]. The sensors capture, process and then communicate data across the network continuously from their environments. In order to achieve final inferences [4], the information gathered from various network parts is then combined. Given that sensors are distributed randomly in unsafe and unstable conditions, human interference is therefore not very necessary. Sensors are autonomous entities that are self-conscious, self-configurable and collect and transmit data. WSN's significant applications include. military applications, remote control of the environment, safety surveillance, home automation, and control of ecosystems etc. [1].

Sensor nodes are devices powered by battery with limited computational capacities and a restricted sensing range, so collaborative efforts are needed in order to compensate the nodes for each other's shortcomings in order to control the redundant network data flow, thus enabling it to be energy efficient, i.e. the correlation between spatially located nodes. Distributing computer paradigms are needed [5] [6] for the purpose of facilitating collaborative data processing in WSN.

In order to address these issues, WSN introduced a mobile agent model [7] [8]. In this model MAs migrate from node to node to accomplish the assigned task using node resources. MA model has many benefits, such as network load reduction, network latency, sturdiness and fault tolerance overcoming.

In this paper, we propose a series of sinking positioning strategies in WSNs, which iteratively select possible sinking locations. During our work, the sensor data are moved via multiple sinks to the cloud computing environment. An energy-efficient multi-disk placing method was introduced to maximize network life and decrease the average energy consumption of sensor system. Iteratively, our algorithm maximizes average network life and reduces average energy usage after a small number of iterations. Various network scenarios are used to evaluate and compare the efficiency of multi-disk placement techniques with mobile agent data aggregation methods and random placement algorithm (RPA).

2. RELATED WORK

The discovery of optimum MA traversal itinerary was a big research problem for effective information assortment in the mobile sensor network from various sensor nodes [9]. In applications such as target monitoring or the monitoring of environment the sequence of node in one itinerary has a decisive effect on data fusion quality and accuracy which ultimately influences the major aim of the WSN. Itineraries can be divided into two types. first, static or dynamic; second, when only a single agent or a multiple agent is used. Statically or dynamically, itineraries can be planned [9].

Authors have proposed two approaches in [10]. Local Closest First (LCF), where MA is found at the bottom of its node. A similar approach to LCF is given by Mobile Agency Direct Diffusion (MADD) was proposed for the MA to find a second node with the smallest distance from PE, Global Closest First (GCF). The LCF is only different in the MADD section, but the LCF does not choose the closest node from the PE. However only the spatial positions of nodes are taken into account and are energy efficient.

Genetic algorithms (GA) [12] do not need any definite node for the execution of the algorithm, rather it selects an active node as original node. Since every node must account its position in order to keep global PE information, GA is involved in a lot of overhead communication. The first-source energy minimum (IEMF) and the minimum energy algorithms (IEMA) are presented in two energy-efficient approaches in [13].The IEMF will pick the first source node, which will have the lowest estimated energy cost, among other itineraries. LCF will then plan the rest of the itinerary. In order to further improve energy efficiency, IEMA iterates the IEMF *k* times

In [14], the authors suggested a strategy to eliminate maximum delays in WSNs for Geographic Sink Placement (GSP). In GSP a segment of a circle places sinks in the center of gravity. To measure the center of gravity, GSP uses field radius and number of sensors. Smart Sink Placement [14] determines the best sink locations to minimize delays in the worst case. The ISP uses the number of sensors, time, transmission distance and number of sinks to obtain the best possible positions.

A number of constrained sink placement techniques were introduced by authors in [15]. The authors believed the well-known algorithm K-means could be used as an optimum placement in sinks and the final centroids. In [16] and [17], with and without WSNs, the authors proposed several sink sites. Their research seeks to minimize total connectivity and computing. In [18], two sink positioning algorithms were introduced to minimize deployment costs, ensuring at least two sinks cover each of the sensors.

Authors [19] suggest an approach based on the MA data aggregation that decides on the route on a fly. The remaining power of the node has also been incorporated as an important cost parameter so that energy consumption is balanced between nodes. This method can also balance the consumption of energy in the nodes, which increases the existence of the sensor network.

3. PROPOSED SYSTEM

3.1 NETWORK MODEL

We take into account a wireless sensor network with a set of sensor nodes V, a set of sink positions S_p and a set of E connections that E stands for communication among sensors. Sensor nodes in the $L\times L$ square region are placed randomly. We assume all sensors and sinks will be static after deployment. Each sensor and sink has a position in its own right, which is known accordingly. A single I_d starts at 1 to $|V|$, for each node and contains a set of transmissions (R). We assume that the transmission range of all nodes is similar. Two sensor nodes u and v are linked to each other by a connection only if they are within each other's range. If there is less distance from the node to the sink it will create a relation between the sink and the node. The sinks are powerful without limits. However, each node of sensors u is only restricted in power $RE(u)$. Let f_u be the rate at which the sensor u transmits the data to the sink. The sink receives only the data, and every node sends and receives the data.

3.2 CONTROLLER

Control networks are generally as small as data networks. The network needs to monitor the quality of small packets, limited packet delay and packet delivery. Packets that are delayed and dropped lower network performance. Therefore, it can control and set the correct path controller support using the routing information. It can support the generation of trace files, transfer packets, and protect the data. The behavior of the system and the protocols are just like a report from a controller for that permission. All processes are carried out on the network. The networks have set the individual node parameters and transmit packets via the system.

3.3 SYSTEM ANALYSIS

We suggest an energy-efficient strategy for multiple sinks, i.e. to increase the mean network length and average power consumption of network systems. The algorithms proposed were four phases. the deployment phase, the implementation phase, the potential position of LS sinks, and the final location of sinks. During the deployment process, nodes and sinks will be randomly distributed in the area. During the setup phase the initial sink locations are selected and nodes are grouped. In the third phase, local sinks are investigated to maximize or minimize the objective function. The last phase identifies the final sink locations. We perform second and third phases in each iteration and take the best solution.

3.3.1 Node Deployment:

The topological usage of nodes is another factor, which depends on application and Impacts the routing protocol's efficiency. Deployment is deterministic or autonomous. The sensors are placed manually and the data is routed in deterministic circumstances along predetermined routes. The sensor nodes are distributed randomly but in the self-organizing system, they create a network ad hoc.

3.3.2 Setup Phase:

This process comprises four sub-phases, including (i) random choice of sink locations, (ii) node clustering, (iii) energy-aware generation of communication tree and (iv) average network lifetime and (v) average network power consumption estimate.

3.3.3 Random Selection of Sink Locations:

We choose K locations from set of sink locations S_G on a random basis at each iteration S_p, where $S_p \leq S_G$, is the candidate sink location.

3.3.4 Grouping of Nodes:

The grouping of sensors is the next step in selecting the candidate locations. In WSNs, grouping is important for local nodes management and for balancing nodes in terms of different quality of service (QoS) parameters. During this phase, V nodes are partitioned by a mechanism based on the distance into $K=|S_p|$ number of groups. A node should be aware of the location information of S_p to create a group. The distance of each node u from each sink $\{s_i\}\in S_p$ is calculated and then associated to the nearest sink. Distance is determined in our work on the basis of the time of arrival (ToA) method. Let, A_{si} be the set of nodes nearest to the sink and for each $u\in A_{si}$ link (u,s_i) is formed.

3.4 ENERGY AWARE COMMUNICATION TREE GENERATION

If a group has a node, we calculate the link, and the link represents the energy cost of communication through the link. The average life and energy consumption calculation is given below:

We select node u for the least life time of each $T_{S_i}(s_i \in S_p)$, and this duration of life is regarded as the life time of the sink$\{s_i\}$. Similarly, we calculate the total energy load of all nodes in A_{S_i}, which is the energy of S_i in T_{S_i}. Using Eq.(1) and Eq.(2) we will find $L(S_p)$ and $E(S_p)$, respectively, after calculating the total energy and life of each sink $\{s_i\} \in S_p$.

$$L(G) \equiv L(S_p) = \frac{1}{|S_p|} \sum_{s_i \in S_p} L(s_i) \tag{1}$$

$$E(G) \equiv E(S_p) = \frac{1}{|S_p|} \sum_{s_i \in S_p} E(s_i) \tag{2}$$

3.5 DETERMINATION OF SINK LOCATION USING LS

We search local Search (LS) for the possible sink locations in each iteration. Let S_p be number of candidate sites in sub-phase 1. Next, the remaining sinks are explored, to search for a subset of RS sinks that can serve to replace some of S_p existing sinks to maximize $L(S_p)$.

This is the local search tool for determining the position of the sink. Iteratively, our program works. We randomly choose Sp sink position from SG at the beginning of each iteration. The average lifetime of the network $L(S_p)$ or a medium energy consumption of the $E(S_p)$ network is then determined. The selected candidate sites are the best value at first, and the local hunt continues iteratively for the candidate sites.

Algorithm: Energy Efficient Sink Placement

Step 1. Deploy nodes and sinks randomly

Step 2. Check the cost factor of nodes

Step 3. Check the iterations of node points

- a. Select the sink randomly from setup phase
- b. Choose candidate sink locations
- c. Calculate distance between sink and individual nodes
- d. Check the every node to be nearer to sink or not

Step 4. Each sinks to be act as potential sink

Step 5. For each link check possibility for routing

- a. Calculate the energy levels for routing in network
- b. Apply method and check network lifetime possibility to check individual nodes energy
- c. Sink location identification starts
- d. Placement of sink for supports to best routing
- e. Communication process between the users and sink

Step 6. End

4. RESULT AND DISCUSSION

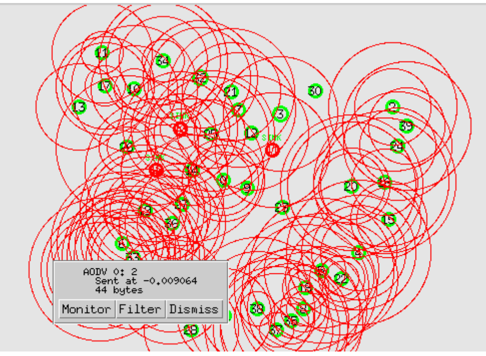
We conduct our experiments with the simulator NS-2.35. We carry out two stages of the experiments. The first step is to verify the viability of our plan and to investigate further the delays, energy consumption and performance.

The first step is to provide the network with 43 mobile nodes and communication begins from source to destination. Here contact takes place and the distance can be determined based on the location of a single node. Number of data flows calculated for the individual contact between server and device. Here we will know each node transmission rate based on the time of simulation. In our work, we can retain resources, delay and find the optimal routing path for individual nodes.

We use CBR network traffic with 1000 bytes/0.1 ms in radio transmission of 250 m. We use application traffic. Data are sent in the form of packs with a size of 1000 bytes and a channel data rate of 2Mbps and a maximum speed of 25m/s. The simulation time is 10s, 1000×1000 is over the network area, and we use AODV to route it. Some forms of routing include RPA, DMADA, and EEMSP.

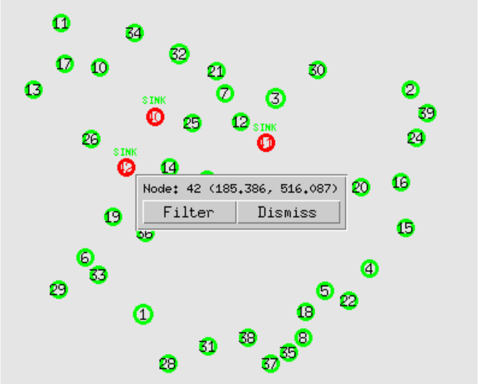
Table.1. Simulation Table

Parameter	Value
Application Traffic	CBR
Transmission rate	1000 packets/0.1ms
Radio range	250m
Packet size	1000 bytes
Channel data rate	2Mbps
Maximum speed	25m/s
Simulation time	10s
Number of nodes	43
Area	1000×1000
Routing protocol	AODV
Routing methods	RPA, DMADA, EEMSP



```
AODV 0: 2
  Sent at -0.009064
  44 bytes
```

Fig.1. Network Deployment with Broadcasting

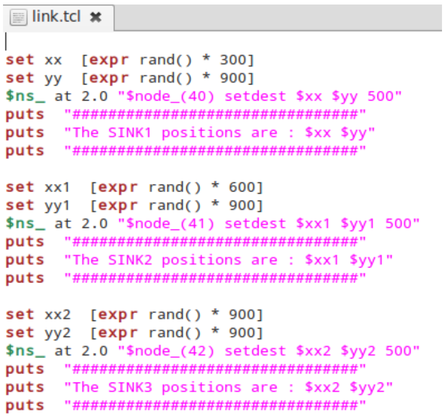


```
Node: 42 (185.386, 516.087)
```

Fig.2. Hello Packets Exchange in network

In Fig.1, above, we can see the deployment of nodes on the network. The display is expressed that is based on node topology values and also tracks the name window's properties. In this screenshot, we see the network's broadcast for communication. In this process, all nodes participate.

With the help of routing protocols, all nodes exchange packets. Every topological node value shown in Fig.2. Each node deployed by the random waypoint model in the network.



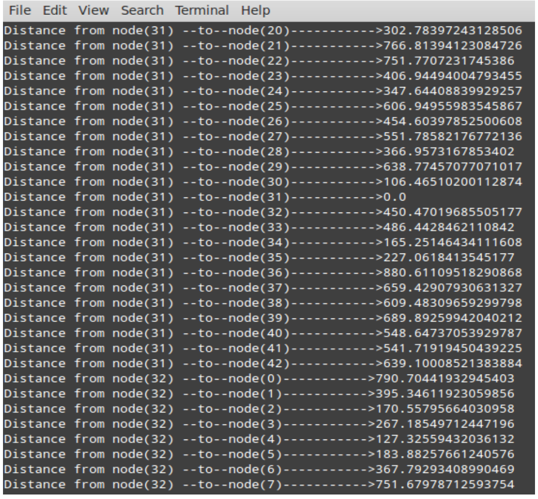
```
set xx  [expr rand() * 300]
set yy  [expr rand() * 900]
$ns_ at 2.0 "$node_(40) setdest $xx $yy 500"
puts  "##############################"
puts  "The SINK1 positions are : $xx $yy"
puts  "##############################"

set xx1  [expr rand() * 600]
set yy1  [expr rand() * 900]
$ns_ at 2.0 "$node_(41) setdest $xx1 $yy1 500"
puts  "##############################"
puts  "The SINK2 positions are : $xx1 $yy1"
puts  "##############################"

set xx2  [expr rand() * 900]
set yy2  [expr rand() * 900]
$ns_ at 2.0 "$node_(42) setdest $xx2 $yy2 500"
puts  "##############################"
puts  "The SINK3 positions are : $xx2 $yy2"
puts  "##############################"
```

Fig.3. Sink positions displayed

In Fig.3 above the sink positions in the network are shown and represented. Placement of sinks depends on the random number and address of the destination. Based on the sink node location, further data is transferred.



```
Distance from node(31) --to--node(20)----------->302.78397243128506
Distance from node(31) --to--node(21)----------->766.81394123084726
Distance from node(31) --to--node(22)----------->751.7707231745386
Distance from node(31) --to--node(23)----------->406.94494004793455
Distance from node(31) --to--node(24)----------->347.64408839929257
Distance from node(31) --to--node(25)----------->606.94955983545867
Distance from node(31) --to--node(26)----------->454.60397852500608
Distance from node(31) --to--node(27)----------->551.78582176772136
Distance from node(31) --to--node(28)----------->366.9573167853402
Distance from node(31) --to--node(29)----------->638.77457077071017
Distance from node(31) --to--node(30)----------->106.46510200112874
Distance from node(31) --to--node(31)----------->0.0
Distance from node(31) --to--node(32)----------->450.47019685505177
Distance from node(31) --to--node(33)----------->486.4428462110842
Distance from node(31) --to--node(34)----------->165.25146434111608
Distance from node(31) --to--node(35)----------->227.0618413545177
Distance from node(31) --to--node(36)----------->880.61109518290868
Distance from node(31) --to--node(37)----------->659.42907930631327
Distance from node(31) --to--node(38)----------->609.48309659299798
Distance from node(31) --to--node(39)----------->689.89259942040212
Distance from node(31) --to--node(40)----------->548.64737053929787
Distance from node(31) --to--node(41)----------->541.71919450439225
Distance from node(31) --to--node(42)----------->639.10008521383884
Distance from node(32) --to--node(0)----------->790.70441932945403
Distance from node(32) --to--node(1)----------->395.34611923059856
Distance from node(32) --to--node(2)----------->170.55795664030958
Distance from node(32) --to--node(3)----------->267.18549712447196
Distance from node(32) --to--node(4)----------->127.32559432036132
Distance from node(32) --to--node(5)----------->183.88257661240576
Distance from node(32) --to--node(6)----------->367.79293408990469
Distance from node(32) --to--node(7)----------->751.67978712593754
```

Fig.4. Distance between users

Each node to node distance calculation is shown in Fig.4. For calculating the distance between the users, we use the Euclidian Formula.

The delay and also the simulation time vs. time are shown in Fig.5. This means that several SINKs are executed, which reduce the delay in time by reducing the contact time between nodes compared to earlier methods such as RPA, DMADA and EEMSDA.

The Fig.6 above shows the throughput, i.e. the network output that shows the simulation time versus throughput. This shows the enhanced efficiency that increases throughput compared to previous methods such as RPA, DMADA and EEMSDA.

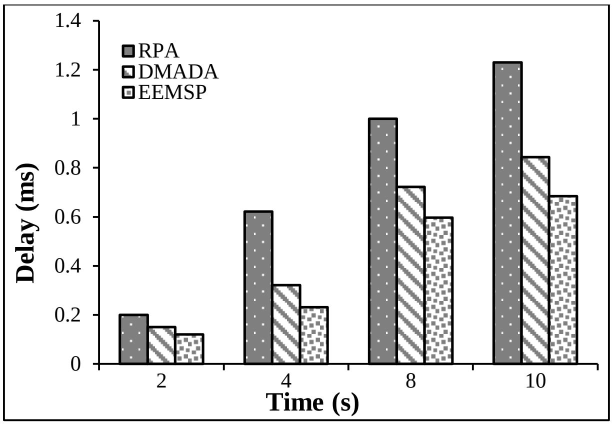


Fig.5. Routing Delay

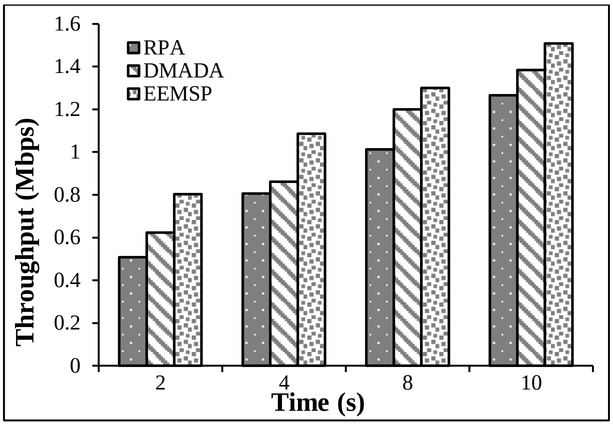


Fig.6. Network Performance

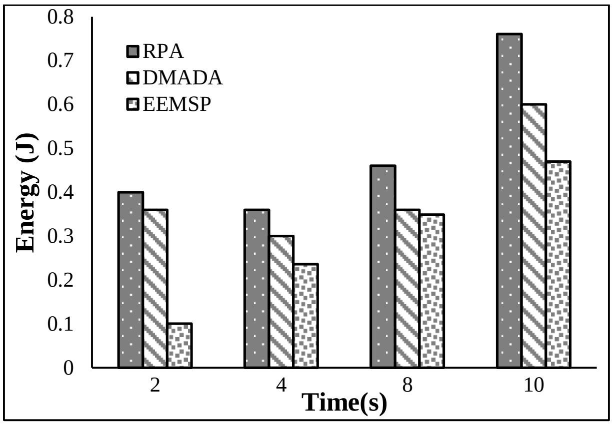


Fig.7. Energy Consumption

The Fig.7 above demonstrates the energy utilization and time versus energy simulation. This shows that the execution of the Multiple SINK scheme reduces energy values compared to previous methods such as RPA, DMADA and EEMSDA.

5. CONCLUSION

In this paper, we have proposed several strategies for positioning WSN energy efficiency sinks. Our goal is to increase the total network energy usage and decrease the overall network energy utilization. Our technique is proposed, which runs a limited number of iterations in order to increase network energy efficiency. Algorithms proposed are simulated in relation to specific performance metrics in addition to data aggregation from Dynamic Mobil Agents and the random positioning algorithm. Experimental outcomes indicate that algorithms proposed are good than algorithms already used. To simulate the network process, we use NS-2 simulation tool.

REFERENCES

[1] I.F. Akyildiz, W. Su, Y. Subramaniam and E. Cayirci, "A Survey on Sensor Networks", *IEEE Communications Magazine*, Vol. 40, No. 8, pp. 104-112, 2002.

[2] J. Yick, B. Mukherjee and D. Ghosal, "Wireless Sensor Network Survey", *Computer Networks*, Vol. 52, No. 12, pp. 2292-2330, 2008.

[3] H. Qi, S. Iyengar and K. Chakrabarty, "Distributed Sensor Networks-A Review of Recent Research", *Journal of the Franklin Institute*, Vol. 338, pp. 655-668, 2001.

[4] F. Zhao, J. Shin and J. Reich, "Information-Driven Dynamic Sensor Collaboration", *IEEE Signal Processing Magazine*, Vol. 19, No. 2, pp. 61-72, 2002.

[5] Y. Xu, H. Qi and P.T. Kuruganti, "Distributed Computing Paradigm for Collaborative Processing in Sensor Networks", *Journal of Parallel and Distributed Computing*, Vol. 64, No. 8, pp. 945-959, 2004.

[6] M. Sivaram, V. Porkodi and A.S. Mohammed, "Improved Enhanced DBTMA with Contention-Aware Admission Control to Improve the Network Performance in Manets", *Computers, Materials and Continua*, Vol. 60, No. 2, pp. 435-454, 2019.

[7] H. Qi, Y. Xu and X. Wang, "Mobile-Agent-Based Collaborative Signal and Information Processing in Sensor Networks", *Proceedings of the IEEE*, Vol. 91, No. 8, pp. 1172-1183, 2003.

[8] M. Chen, T. Kwon, Y. Yuan and V.C.M. Leung, "Mobile Agent based Wireless Sensor Networks", *Journal of Computers*, Vol. 1, No. 2, pp. 6-10, 2006.

[9] M. Chen, S. Gonzalez and V.C. Leung, "Applications and Design Issues of Mobile Agents in Wireless Sensor Networks", *IEEE Wireless Communications Magazine*, Vol. 14, No. 6, pp. 20-26, 2007.

[10] H. Qi and F. Wang, "Optimal Itinerary Analysis for Mobile Agents in Adhoc Wireless Sensor Networks", *Proceedings of IEEE International Conference on Communications*, pp. 1-6, 2001.

[11] M. Chen, T. Kwon, Y. Yuan, Y. Choi and V.C. Leung, "Mobile Agent-Based Directed Diffusion in Wireless Sensor Networks", *EURASIP Journal on Advances in Signal Processing*, Vol. 1, No. 3, pp. 219-227, 2007.

[12] Q. Wu, N.S.V. Rao, J. Barhen, S. Sitharama Iyengar, V.K. Vaishnavi, H. Qi and K. Chakrabarty, "On Computing Mobile Agent Routes for Data Fusion in Distributed Sensor Networks", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 16, No. 6, pp. 740-753, 2004.

[13] M. Chen, V. Leung, S. Mao, T. Kwon and M. Li, "Energy-Efficient Itinerary Planning for Mobile Agents in Wireless Sensor Networks", *Proceedings of IEEE International Conference on Communications*, pp. 1-5, 2009.

[14] W.Y. Poe and J.B. Schmitt, "Minimizing the Maximum delay in Wireless Sensor Networks by Intelligent Sink Placement", Technical Report, Distributed Computer Systems Lab, University of Kaiserslautern, No. 362-07, pp 1-20, 2007.

[15] J. Flathagen, O.Kure and P.E.Engelstad, "Constrained-Based Multiple Sink Placement for Wireless Sensor Networks", *Proceedings of 8th IEEE International Conference on Mobile Ad hoc and Sensor Systems*, pp. 1-7, 2011.

[16] W.Y.Poe and J.B. Schmitt, "Self-Organized Sink Placement in Large Scale Wireless Sensor Networks", *Proceedings of IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, pp. 111-117, 2009.

[17] W.Y.Poe and J.B. Schmitt, "Sink Placement Without Location Information in Large Scale Wireless Sensor Networks", *Proceedings of Asian Conference on Internet Engineering*, pp. 1-7, 2009.

[18] L. Sitanayah, K.N. Brown and C.J. Sreenan, "Multiple Sink and Relay Placement in Wireless Sensor Networks", *Proceedings of the 1st Workshop Artificial Intelligence for Telecommunications and Sensor Networks*, pp. 31-39, 2012.

[19] Divya Lohani and Shirshu Varma, "Energy Efficient Data Aggregation in Mobile AgentBased Wireless Sensor Network", *Wireless Personal Communications*, Vol. 89, No. 4, pp. 1-23, 2016.