International Journal of Current Engineering and Technology
E-ISSN 2277 – 4106, P-ISSN 2347 - 5161

Available at http://inpressco.com/category/ijcet

Research Article

Mobile Agent Cluster-Based Algorithm for Data Processing in Wireless Sensor Networks

Ahmed. A. A. Gad-ElRab[Å*], Afaf A. S. Zaghrout[Å] and Doaa M.Alhilaly[Å]

[Å]Department of Mathematics, Al-Azhar University, - Cairo, Egypt

Accepted 15 Nov 2014, Available online 31 Dec 2014, **Vol.4, No.6 (Dec 2014)**

Abstract

Minimizing energy consumption and maximizing network life time are concern issues for data processing in wireless sensor networks (WSNs). In this paper, we propose mobile agent cluster-based algorithms to satisfy these concern issues. In the proposed algorithms, we use a distributed multihop clustering algorithm to organize the sensor nodes in a WSN into clusters. Based on this clustering, we use a mobile agent system to enhance energy-efficient data processing by selecting its visiting set of nodes for facilitating agent-based data collection and aggregation by using satisfaction percentage of each node. Simulation results show that the proposed algorithms achieve higher improvements in network lifetime, load balance and energy consumption than the existing algorithms.

Keywords: *Multihop, mobile agent, clusterhead, round , satisfaction, sorting, unsorting.*

1. Introduction

The advances in Micro electromechanical System (MEMS) and wireless communication have enabled the development of a new kind of the wireless sensor network (WSN). A wireless sensor network (WSN) consists of hundreds or thousands of small sensor nodes, which are deployed over a hostile, in- habitable and harsh environment, possibly for a limited period with a common objective and collaborate to provide distributed sensing, storage and communication services. These sensor nodes can organize themselves in such a way that they act as front line observation for end users placed at a far distance (A. A. Abbasi, *et al*, 2007), (G. Anastasi, *et al*, 2009). WSNs support a wide range of useful applications like Military applications, Environmental applications, Health applications, Home applications and Commercial applications. The characteristic of WSNs brings up some important issues for cooperation communication, including energy efficiency, scalability and reliability (I. F. Akyildiz, *et al*, 2002), these are the challenges that the focus of the researchers, most of researches focus on prolonging the network lifetime, allowing scalability for a large number of sensor nodes or supporting fault tolerance (e.g., sensors failure and battery depletion) (H. Qi, *et al*, 2003) , (J. J. Chang, *et al*,2007). One the critical challenges in (WSNs) is overwhelming data traffic. To meet this challenge, one of the unique features of WSN applications is the necessity of cooperation. This cooperation come from using mobile agents reduce the network traffic, provide an effective means of overcoming network latency and helps you to construct more robust and fault-tolerant. Generally speaking an MA is a special kind of software that can execute autonomously, with identification, itinerary, data space and method as its attributes. For example, the sensory data of two closely located sensors are likely to have redundant or common parts when the data of two sensors are merged. Therefore, data aggregation is a necessary function in densely populated sensor networks in order to reduce the sensory data traffic.

Minimizing energy consumption and maximizing network life time are another concern challenges for data processing in wireless sensor networks (WSNs). The clustering algorithms were extensively used to meet these challenges and can increase the scalability, stability, and network lifetime. In clustering schemes, the communication between a sensor node and its designated cluster head CH is multihop communication is often required when the number of sensor nodes is very large in a network, the selected CHs collect data from member nodes in their respective clusters, aggregate the data, and send it to a base using multihop communication. In this paper, we propose mobile agent cluster-based algorithms to satisfy these concern issues by combining MA and clustering schemes. As a result, large amount of sensory data can be reduced or transformed into small data by eliminating the redundancy by using MA and reduced flooding by clustering. In the proposed algorithms, we use a distributed, multi-hop clustering algorithm to organize the sensor nodes in a WSN into clusters. Based on this clustering, we use a mobile agent system to enhance energy-efficient data processing by selecting its visiting set of nodes for facilitating agent-

*Corresponding author: **Ahmed. A. A. Gad-ElRab**

based data collection and aggregation by using satisfaction percentage of each nodes. One method for extending the sensor network lifetime is to divide the set of sensor nodes into independent sets such that at each round of data collection every set aggregates all required information. These independent sets are activated successively, such that at any round only one set is active. The nodes from the active set are into the active state and all other sensors are in a low-energy sleep state. The goal of this approach is to determine the number of independent sets according to maximum number of available rounds. Through simulations, we have confirmed that our algorithms achieve higher improvements in network lifetime, load balance and energy consumption than the existing algorithms. The rest of the paper is organized as follows. Section II includes a detailed survey of the related research. Section III and IV exhibits the detail of the proposed scheme. Simulation results and its discussion are presented in Sections V. Finally, Section VI concludes the paper.

2. Related Work

Recently there has been a growing interest on the design, development, and deployment of MA systems for high-level inference in (WSNs) (Shukui Zhang, *et al*, 2012), (M. Chen, *et al*, 2007), (Y. C. Tseng, *et al*, 2004). In (Shukui Zhang, *et al*, 2012), author present a cooperative data processing algorithm based on mobile agent (MA-CDP)and considers MA in multihop environments, can autonomously clone and migrate themselves in response to environmental changes.

In our algorithm large amount of sensory data can be reduced or transformed into small data by eliminating the redundancy by using MA and reduces flooding by clustering. In (MACDP) algorithm depend on MA multi-hop but there is excessive use of power by flooding and we were able to overcome it by clustering furthermore, to have better understanding in evaluating the performance of the algorithm, we present detailed analytical algorithm of data dissemination. With appropriate parameters set, the results of our simulation show that cooperative data processing and cluster-based algorithm provides better performance in terms of packet delivery ratio, energy consumption and end-to-end delay. In (M. Chen, *et al*, 2007), the agent design in (WSNs) is decomposed into four components architecture, itinerary planning, middleware system design, and agent cooperation. Among the four components, itinerary planning determines the order of source nodes to be visited during agent migration, which has a significant impact on energy performance of the MA system. It has been shown that finding an optimal itinerary is a NP-hard problem. Therefore, heuristic algorithms are generally used to compute competitive itineraries with a suboptimal performance.

In (H. Qi, *et al*, 2001), two simple heuristics are proposed: (i) a local closest first scheme that searches for the next node with the shortest distance to the current node and (ii) a global closest first scheme that searches for the next node closest to the dispatcher. These two schemes only consider the spatial distances between sensor nodes and thus may not be energy efficient in many cases.

In (Dilip Kumar1, *et al*, 2011), authors propose a distributed randomized multihop clustering algorithm to organize the sensor nodes in a WSN into clusters.

In (G. Sharma, *et al*, 2005), the authors have investigated the use of limited infrastructure, that is, networks with a number of wired connections between sensor nodes. Their approach establishes a small-world graph by utilizing wired links between a subset of nodes to reduce the overall energy demands as well as the different energy consumption rates of participating nodes. The additional efforts required for the wiring however make it suited for long-term deployments of sensor networks only. In (G. Sharma, *et al*, 2005), the clustering routing algorithm is firstly presented by the LEACH.

In (] G. Xin, *et al*, 2008), the authors have proposed energy-efficient hierarchical clustering algorithm (EEHCA) for WSN which improves the performance of LEACH and HEED (O. Younis, *et al*, 2004). In (F. Xianging, *et al*, 2007), authors have studied LEACH scheme and proposed two new schemes (i.e., energy-LEACH and multihop LEACH). Energy-LEACH improves the CH selection method and multi- hop LEACH (M-LEACH) improves the communication mode from single-hop to multi-hop between CH and BS. Both the schemes have better performance than LEACH scheme. In (] I. Joe, 2007), (Shukui Zhang, *et al*, 2012), authors solve the overwhelming data traffic especially over low bandwidth links and MA selectively migrates among sensor nodes by moving the processing function to the target nodes, performs local processing by using resources available at the local nodes rather than bringing the data to a central processor (sink) and incrementally fuses the local decisions on each sensor node to reach a progressively accurate global decision, but this algorithms consumed energy by the flooding.

This limitation is tackled by MADD algorithm (M. Chen, et al, 2005). The processes involved in MADD are divided into some sections. First, the MA is dispatched from sink to the first source node, and in the next place, the MA migrates from first source node to last source node, visiting selected source nodes in between.

This algorithm does not always guarantee the best sequence of nodes to be visited and consumed energy by the flooding. Most of existing works do not meet the required challenges where some of them cannot maximize network life time and others generate high network traffic, so we propose a Mobile Agent Cluster-Based Algorithms to solve: (1) overwhelming data traffic (sending data to one place) by clustering (2) energy consumption (sending a query to all network as ooding) by an MA selectively migrates among clusters and sensor nodes inside each cluster by moving the processing function to the target clusters and nodes and performs local processing by using resources available at CHs and local nodes rather than bringing the data to a central processor (sink).

3. Data Processing Problem in WSNs

The main objective of this study is to propose an adaptive solution to maximize network lifetime for data processing in WSNs. In this section, we will describe the WSN model, its parameters, and formulate the data processing problem.

For all sensor network applications, the overall tasks of sensor networks are monitoring, collecting and processing the sensed data from the covered area and sending the data back to the observers. However, the computation abilities, storage capacities, communication bandwidth and power of sensors are limited, this brings a lot of challenges. We define the network aggregation process as follows: network aggregation is the global process of gathering and routing information through a multihop network, processing data at intermediate nodes with the objective of reducing resource consumption (in particular energy), thereby increasing network lifetime. Data processing techniques improve the energy efficiency and it is a typical parameter measure of performance in sensor networks, therefore we use both: (1) clustering in order to create an independent clusters in data aggregation and processing processes and (2) a mobile agent, MA, which is a special process that can autonomously migrate and aggregate data across nodes. MA systems provide new capabilities for energy-efficient data processing by flexibly planning its itinerary for facilitating agent-based data collection and aggregation.

3.1 Definition, Assumptions and Models

We assume there is a wireless sensor environment which contains a set of sensor nodes *V*. We assume that Q_i is an information generation rate at sensor node i . We denote a set of neighboring nodes of sensor i as S_i . We assume that e_{ij}^T and e_{ji}^R are energy consumed at sensor i to transmit a data unit to sensor j and energy consumed at sensor j to receive the data unit transmitted by sensor i , respectively. We assume that q_{ij} and $q_{ij}^{\max}$ are a rate at which information is transmitted from sensor i to its neighboring node j and a maximum possible rate at which information can be transmitted from sensor i to its neighboring node j , respectively. We denote a maximum power that sensor node i can spend at every round as p_i . We denote an initial amount of battery energy at sensor node as E_i . Also, we denote a number of aggregation rounds as *R* and a number of collection times by node i as t_i .

Generally, we divide the sensor network into clusters. Each cluster has a cluster head which identifies all nodes in his cluster, contacts with other clusters and determines MA path to mobility between clusters. We provide MA with memory for storing previous queries to make cluster an effective with nodes failure.

Based on our model and defined parameters, we can define the overall network consumed energy as follows:

$$CE(V) = \sum_{n\in C_{ch}} \sum_{j\in S_i^{C_{ch}}} e_{ij}^T q_{ij}^{C_{ch}} t_i + \sum_{n\in C_{ch}} \sum_{j:i\in S_j^{C_{ch}}} e_{ji}^R q_{ji}^{C_{ch}} t_i \qquad (1)$$

3.2 Problem Formulation

The problem of maximizing overall sensor network lifetime depends on how to we reduce energy consumption. So, based on the clustering scheme which will be used, we define a number of collection times, t_i , for each node, $i \in V$. Our problem is how to minimize the total energy consumed at sensor *i* during network operation. This problem is the combinatory optimization problem. We use linear programming formulation to develop new heuristic algorithms that take into account this optimization problem which will capable of expressing network lifetime in terms of the number of collection times for a node. The problem of maximizing the overall network lifetime can be written as follows:

Minimize $CE(V)$ (2)

Subject to:

$$t_i \geq 0 \qquad (3)$$

$$0 \leq q_{ij}^{C_{ch}} \leq q_{ij}^{\max}, i \in V, j \in S_i^{C_{ch}} \qquad (4)$$

$$CE(V) \leq E_i, i \in V \qquad (5)$$

$$\sum_{j\in S_i^{C_{ch}}} e_{ij}^T q_{ij}^{C_{ch}} + \sum_{j:i\in S_j^{C_{ch}}} e_{ji}^R q_{ji}^{C_{ch}} \leq p_i, i \in C_{ch} \in V \qquad (6)$$

$$\sum_{j:i\in S_j^{C_{ch}}} q_{ji}^{C_{ch}} + Q_i = \sum_{j\in S_i^{C_{ch}}} q_{ij}^{C_{ch}} i, j \in C_{ch} \qquad (7)$$

Equation (3) describes that a number of collection times any sensor node *i* is larger than or equal 0. In other words, t_i must be a positive value. Equation (4) represents the rate at which information is transmitted from sensor *i* to its neighboring node j and to its cluster head inside cluster must be less than or equal to maximum possible rate at which information can be transmit- ted from sensor *i* to its neighboring node *j* inside cluster and must be greater than or equal to zero. Equation (5) represents we disregard any specific details of battery technology and assume that the power and energy expenditures are directly proportional to the rate at which information is transmitted/received. Given the number of times of collection of each node t_i and the information transfer rates $q_{ij}^{C_{ch}}$ the power (energy per time unit) consumed at sensor node *i* must be less than or equal maximum power that sensor node $i \in c_{ch}$ can spend inside cluster.

Equation (6) represents the total energy consumed at sensor i during network operation is the sum of the quantities in (4) all times in collection must be less than or equal to initial amount of battery energy at sensor node $i \in V$. Equation (7) represents the total incoming information transfer rate plus the information generation rate from the sensor. In the next section, we will propose an algorithm to deter- mine these independent sets.

4. Mobile Agent Cluster-Based Algorithms (MACB)

4.1 Basic Idea

Based on clustering, our basic idea is based on: (1) using mobile agents (MAs) to move among nodes inside each cluster and among existing clusters heads, (2) dividing a set of member nodes inside each cluster CH into independent sets. These independent sets are activated successively, such that at any round **r** ∈ R only one set is active. The nodes from the active set will be in their active state and all other sensors are in a low-energy sleep state, (3) using a required satisfaction percentage which is requested by application task through a sink. Also, we assume that there is a satisfaction percentage for each node i called sat(i) which represents the ability percentage of node to satisfy the requested task based on its available memory and its remaining energy defined as follows.

$$sat(i) = \frac{e_i M + m_i E}{2ME} \tag{8}$$

Where

M maximum memory threshold
E maximum energy threshold

Based on our ideas, the problem is how to determine the number of independent sets for each cluster according to maximum number of available rounds R and required satisfaction rs to maximize the number of collection times, t_i of a sensor node, i and the number of collection times, t_{ch} of a cluster head CH. Our problem is related to Maximum Independent Set (MIS) problem which is known as NP-completeness (Richard M. Karp, 1972). To solve this problem, we need to find the optimal independent sets for each cluster in the target area which can maximize the network lifetime as much as possible. To satisfy this goal, we propose three different methods that use a satisfaction percentage of each node and task required satisfaction rs. In each proposed method a cluster head ch selects its independent sets inside its cluster as follows: $ch \in CH$ construct a set dn_j of nodes by selecting these nodes one by one from a set of nodes of CH and check if they can achieve the required satisfaction (rs) according to the following equations:

$$\{ \sum_{i \in dn_j} sat(i) / |dn_j| \} \geq rs \tag{9}$$

$$DN = \bigcup_j dn_j, 1 \leq j \leq R \tag{10}$$

In the next subsections, we will describe our three proposed methods, in details, then we will describe steps of our data processing scheme by using these proposed methods.

4.2 Proposed Selection Methods

By using MA, clustering, and required satisfaction percentage, we propose a new algorithm based on clustering scheme called Mobile Agent Cluster-Based Algorithm (MACBA). MACBA use three different methods to minimize the total energy consumed for a cluster head and a node.

1) Random Selection Method: In this method, the proposed algorithm will determine a independent sets of nodes randomly this nodes will be visited by MA inside each cluster where each set will be activated for online data aggregation in round $r \in R$.

***Algorithm 1**: Random data processing with clustering [RC]*
Input: Required satisfaction rs

1. Sink broadcasts the query packet and required satisfaction to $ch \in CH$.
2. $ch \in CH$ broadcasts the query packet and required satisfaction rs to nodes inside cluster.
3. $ch \in CH$ will construct a set dn_j of nodes by selecting these nodes randomly one by one from a set of nodes of ch and check if they can achieve the required local satisfaction (rs)
4. Nodes which achieved will be visited by MA.
5. MA do online data aggregation in round $r \in R$.

Output: independent sets of nodes, DN.

2) Unsorting Satisfaction Selection Method: In this method, the proposed algorithm will determine a independent sets of nodes by unsorting Satisfaction Selection method(nodes use One-time)this nodes will be visited by MA inside each cluster where each set will be activated for online data aggregation in round $r \in R$.

***Algorithm 2:**Unsorting satisfaction data processing with clustering [USC]*
Input: Required satisfaction rs

1. Sink broadcasts the query and required satisfaction to S_{ch}
2. $ch \in CH$ broadcasts the query packet and required satisfaction rs to nodes inside cluster.
3. $ch \in CH$ will construct a set dn_j of nodes and check if they can achieve the required local satisfaction (rs) this nodes use One-time and don't repeat in another round until the task is finish.
4. Nodes which achieved will be visited by MA.
5. MA do online data aggregation in round $r \in R$.

Output:- independent sets of nodes, DN.

3) Sorting Satisfaction Selection Method: In this method, the proposed algorithm will determine a independent sets of nodes according to required satisfaction value by sorting their values for nodes in descending order this nodes will be visited by MA inside each cluster where each set will be activated for online data aggregation in round r $\in$ R.

***Algorithm 3**: Sorting satisfaction data processing with clustering [SC]*
Input: Required satisfaction rs

1. Sink broadcasts the query and required satisfaction to S_{ch}.
2. $ch \in CH$ broadcasts the query packet and required satisfaction *rs* to nodes inside cluster.
3. $ch \in CH$ will sort all actual satisfaction values of nodes in descending order.
4. $ch \in CH$ will construct a set dn_j of nodes by selecting these nodes in descending order one by one from a set of nodes of *ch* and check if they can achieve the required local satisfaction (rs).
5. Nodes which achieved will be visited by MA.
6. MA do online data aggregation in round r $\in$ R.

Output:- independent sets of nodes, DN.

4.3 Data Processing Steps

By using MACBA algorithm, we will describe the steps of our data processing scheme as follows.

1) Generally, a sink receives a task request assigned by an application, sink broadcasts the query packet and required satisfaction percentage *rs* to $ch \in CH$ directly or by getaway nodes, sink will deals with all clusters where each cluster will combine in data aggregation in round $r \in R$.

2) Each $ch \in CH$ creates its independent sets of nodes by using RC, USC, or SC methods based maximum number of rounds R and required satisfaction percentage *rs*.

3) For all rounds *R*, each cluster among the target source nodes to be visited, $ch \in CH$ will choose the first and last source nodes According: the size of an MA is the minimum in FirstNo while it becomes the maximum in LastSrc so the target source which is the last (first) to send exploratory messages to the sink is chosen as FirstNo (LastSrc). When the MA arrives at the first source node, it is stored in it to construct another MA from its memory and dispatch it to initiate the new round. In the first round MA moves from source to source to collect and aggregate information and also copies processing code into the memory of each source node so MA does not carry the processing code any more in the following rounds When the whole task is finished, all the source nodes will discard the processing code When MA in firstNo, the sequence of visiting the other source nodes is dynamically decided by each target sensor based on its ToSourceEntry. ToSourceEntry is used for MA to roam among source nodes, target source nodes send exploratory messages enable sensor nodes to set up ToSourceEntry inside the cluster only. Finally, the MA will carry the data results to the $ch \in CH$ along the reinforced path.

4) Finally, MA moves among clusters to combine and aggregate data and finally to the sink.

5. Experimental Results and Analysis

This section validates the efficiency of using the proposed method for data processing using MACBA algorithm through applying it on three different methods, RC, USC, and SC and compares it with traditional random method (Shukui Zhang, *et al*, 2012),. The OMNet++ (A. Varga, *et al*, 2001), simulator was used to evaluate our proposed algorithm MACBA and TR (Shukui Zhang, *et al*, 2012),. Also, we use a fixed number of rounds R and it was 10 in all simulation experiments.

6.1 The average working times of a node

Figure.1 shows the average working times against the required satisfaction where the number of requests was 30 and the number of nodes was 100. As shown in figure. 1 the average working times of a node increases as satisfaction value increases. This is because to satisfy a higher satisfaction, we need to select more nodes or each node will work for more times. Also, this figure shows that the traditional random method TR (Shukui Zhang, *et al*, 2012), achieves high average working times compared to RC, USC, and SC methods. This is because TR does not consider required satisfaction as RC, USC, and SC do. This indicates that the proposed three algorithms are more efficient than the traditional random method (Shukui Zhang, *et al*, 2012).

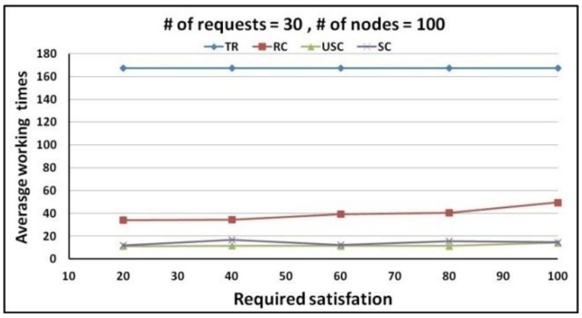

Fig.1 The relation between required satisfaction and average working times

Figure.2 shows the average working time against the number of requests where the number of nodes was 100 and the required satisfaction was 75%. As shown in figure 2 the average working times of a node increases as number of requests increases. This is because to satisfy a higher number of requests, we need to select more nodes or each node will work for more times. Also, the average working times of RC, USC, and SC is much lower than TR (Shukui Zhang, *et al*, 2012). Hence, the RC, USC, and SC are better than TR (Shukui Zhang, *et al*, 2012).

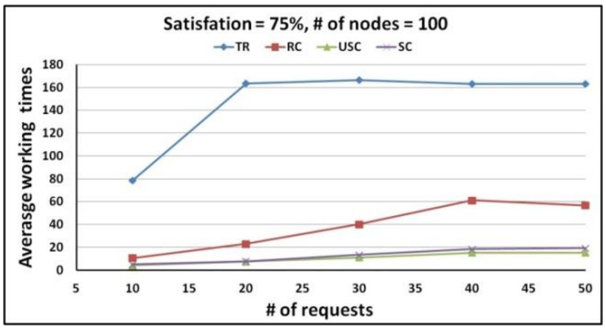

Fig.2 The relation between number of requests and average working times

Figure.3 shows the time working against the number of nodes where the number of requests was 40 and the required satisfaction was 75%. As shown in figure.3 the average working times of a node decreases as number of nodes increases. Also, the average working times of RC, USC, and SC less than TR, this demonstrates to that proposed three algorithms are better than traditional random method TR (Shukui Zhang, *et al*, 2012),.

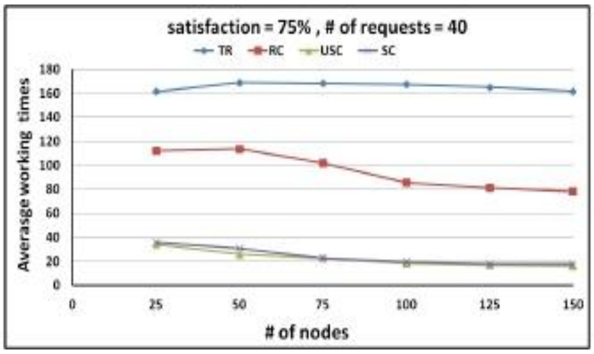

Fig.3 The relation between number of nodes and average working times

6.2 Total number of debilitated nodes

Figure.4 shows the number of debilitated nodes against the required satisfaction where the number of requests was 30 and the number of nodes was 100. As shown in figure 4 the number of debilitated nodes increases as satisfaction value increases.

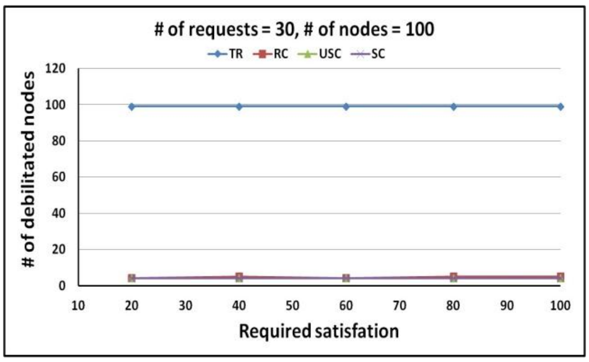

Fig.4 The relation between required satisfaction and number of debilitated nodes

This is because to satisfy a higher satisfaction, we need to select more nodes or each node will work for more times which leads to the death of nodes. Also, this figure shows that the traditional random method, TR (Shukui Zhang, *et al*, 2012), has the largest number of debilitated nodes compared to RC, USC, and SC methods. This is because TR does not consider required satisfaction as RC, USC, and SC do. This indicates that the proposed three Algorithms are more efficient than the traditional random method (Shukui Zhang, *et al*, 2012).

Figure.5 shows the number of debilitated nodes against the number of requests where that the number of nodes was 100 and the required satisfaction was 75%. As shown in figure.5 the number of debilitated nodes increases as number of requests increases. This is because to satisfy a higher number of requests, we need to select more nodes or each node will work for more times which leads to the death of nodes. Also, this figure shows that the traditional random method TR (Shukui Zhang, *et al*, 2012), has the largest number of debilitated nodes compared to RC, USC, and SC methods. Hence, the proposed three algorithms are better than TR (Shukui Zhang, *et al*, 2012),.

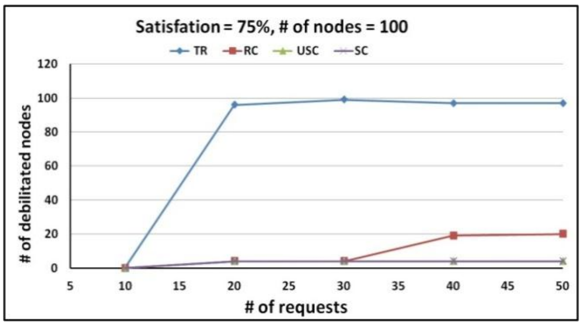

Fig.5 The relation between number of requests and number of debilitated nodes

Figure.6 shows the number of debilitated nodes against the number of nodes where the number of requests was 40 and the required satisfaction was 75%.

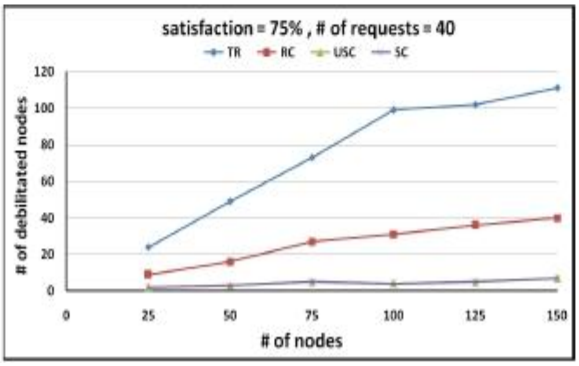

Fig.6 The relation between number of nodes and number of debilitated nodes

As shown in figure 6 the number of debilitated nodes increases as number of nodes increases. Also, this figure shows that the traditional random method TR (Shukui Zhang, *et al*, 2012), has the largest number of debilitated nodes compared to RC, USC, and SC methods. Hence, RC, USC, and SC are better than traditional random method TR (Shukui Zhang, *et al*, 2012).

5.3 Energy Consumption

Figure.7 shows the energy consumption against the required satisfaction where the number of requests was 30 and the number of nodes was 100. As shown in figure.7, the energy consumption increases as satisfaction value increases. This is because to satisfy a higher satisfaction, we need to select more nodes or each node will work for more times which leads to consume more energy. Also, this figure shows that the traditional random method TR (Shukui Zhang, *et al*, 2012), consumes more energy compared to RC, USC, and SC methods. This is because TR does not consider required satisfaction as RC, USC, and SC do. This indicates that the proposed three algorithms are more efficient than the traditional random method (Shukui Zhang, *et al*, 2012).

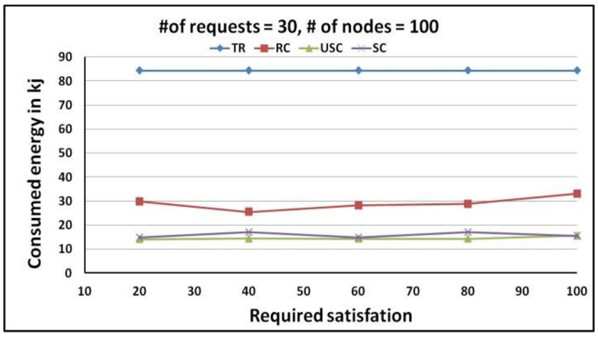

Fig.7 The relation between required satisfaction and consumed energy

Figure.8 shows the energy consumption against the number of request where the number of nodes was 100 and the required satisfaction was 75%. As shown in figure8, the energy consumption increases as number of requests increases.

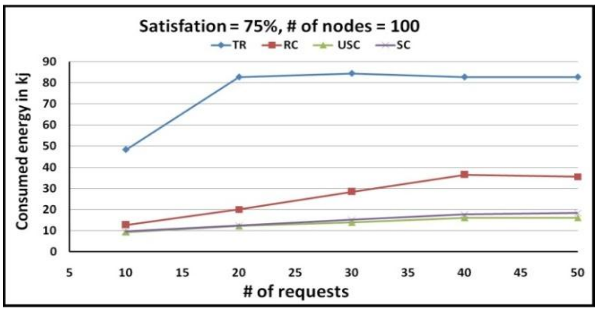

Fig.8 The relation between number of requests and consumed energy

This is because to satisfy a higher number of requests, we need to select more nodes or each node will work for more times which leads to consume more energy. Also, this figure shows that the traditional random method, TR (Shukui Zhang, *et al*, 2012), consumes more energy compared to RC, USC, and SC methods. Hence, the proposed three algorithms are better than traditional random method TR (Shukui Zhang, *et al*, 2012).

Figure.9 shows the energy consumption against the number of nodes where the number of request was 40 and the required satisfaction was 75%. As shown in figure 9, the energy consumption increases as number of nodes increases. Also, TR consumes more energy compared to RC, USC, and SC methods. Hence, the proposed three algorithms are better and more efficient than traditional random method TR (Shukui Zhang, *et al*, 2012).

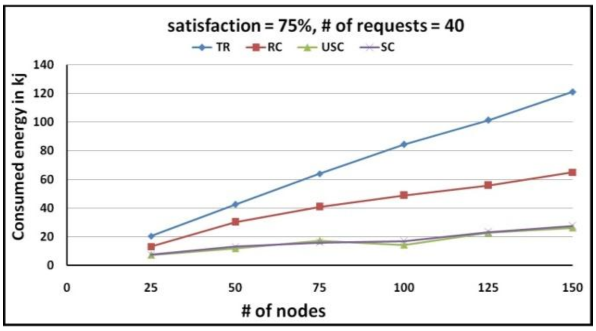

Fig.9 The relation between number of nodes and consumed energy

Conclusions

In this paper, we proposed a new data processing algorithm in wireless sensor networks called Mobile Agent Cluster- Based Algorithm, MACBA, to extend the life time and minimize the total energy consumption of WSNs. MACBA is a cluster based algorithm and it uses a mobile agent entity for each cluster in the network to process and aggregate all sensing data inside it. MACBA proposes three different methods are called clustering with random set selection (RC), clustering with unsorting set selection (USC), and clustering with sorting set selection (SC). All of these methods can reduce the consumed energy, the number of debilitated nodes and the average working time of a node in WSN, compared to the traditional random method (Shukui Zhang, *et al*, 2012),. Hence, MACBA can extend the life time of WSNs in efficient way compared to the existing approaches. In the future work, we will apply the proposed methods with different clustering algorithms. Also, we will study and modify our proposed algorithm in the case of the presence of the mobile sensor nodes in the network.

References

Shukui Zhang,1, 2 Yong Sun,1 Jianxi Fan,1 and He Huang1 ,(2012),Cooperative Data Processing Algorithm Based on Mobile Agent in Wireless Sensor Networks.

M. Chen, S. Gonzalez, and V. C. M. Leung, (2007), Applications and design issues for mobile agents in wireless sensor networks, IEEEm Wireless Communications, vol. 14, no. 6, pp. 20–26.

Y. C. Tseng, S. P. Kuo, H. W. Lee, and C. F. Huang, (2004), Location tracking in a wireless sensor network by mobile agents and its data fusion strategies, Computer Journal, vol. 47, no. 4, pp. 448–460.

H. Qi and F. Wang, (2001), Optimal itinerary analysis for mobile agents in adhoc wireless sensor networks, in Proceedings of the IEEE International Conference on Communications (ICC '01), Helsinki, Finland, June.

Dilip Kumar1 and R. B. Patel2 , (2011),Multi-Hop Data Communication Algorithmfor clustered Wireless Sensor Networks.

G. Sharma and R. Mazumdar, (2005), Hybrid sensor networks: a small world, in Proceedings of the 6th ACM International Symposium on Mobile AdHoc Networking and Computing (MOBIHOC '05), pp. 366–377, USA.

G. Xin, W. H. Yang, and B. DeGang, (2008), EEHCA: an energy efficient hierarchical clustering algorithm for wireless sensor networks, Information Technology Journal, vol. 7, no. 2, pp. 245–252.

O. Younis and S. Fahmy, (2004), HEED: a hybrid, energy-efficient,distributed clustering approach for ad hoc sensor networks, IEEE Transactions on Mobile Computing, vol. 3, no. 4, pp. 366– 379.

F. Xianging and S. Yulin, (2007), Improvement on LEACH protocol of wireless sensor networks, in Proceeding of International Conference on Sensor Technologies and Applications, pp. 260– 264.

A. A. Abbasi and M. Younis,(2007), A survey on clustering algorithms for wireless sensor networks, Computer Communications, vol. 30, no. 14-15, pp. 28262841.

G. Anastasi, M. Conti, M. D. Francesco, and A. Passarella, (2009), Energy conservation in wireless sensor networks: a survey, Ad Hoc Networks, vol. 7, no. 3, pp. 537568.

I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, (2002), A survey on sensor networks, IEEE Communications Magazine, vol. 40, no. 8, pp. 102105.

H. Qi, Y. Xu, and X.Wang, (2003),Mobile-agent-based collaborative signal and information processing in sensor networks, Proceedings of the IEEE, vol. 91, no. 8, pp. 11721183

I. Joe, (2007), A path selection algorithm with energy efficiency for wireless sensor networks, in Proceedings of the 5th ACIS International Conference on Software Engineering Research, Management, and Applications (SERA 07), pp. 419423.

M. Chen, K. Taekyoung, and C. Yanghee, (2005), Data dissemination based on mobile agent in wireless sensor networks, in Proceedings of the IEEE Conference on Local Computer Networks (LCN 05), pp. 527528, Sydney, Australia, November.

Reducibility Among Combinatorial Problems. Richard M. Karp. (1972) Complexity of Computer Computations, page 85-103. Plenum Press, New York.

A. Varga and Andrs, (2001), The OMNeT++ Discrete Event Simulation System, in Proceedings of the European Simulation Multiconference (ESM'2001). Prague, Czech Republic.