

## Dynamic Aggregation Protocol for Wireless Sensor Networks

Adel Mounir Said<sup>1</sup>, Ashraf William Ibrahim<sup>1</sup>, Ahmed Soua<sup>2</sup>, Hossam Afifi<sup>2</sup>,

<sup>1</sup>Switching Department, National Telecommunication Institute – NTI, Cairo, Egypt

<sup>2</sup>RS2M Department, Telecom SudParis, Evry, France

amounir@nti.sci.eg, awilliam@nti.sci.eg, Ahmed.Soua@it-sudparis.eu, Hossam.Afifi@it-sudparis.eu

**Abstract**— Sensor networks suffer from limited capabilities such as bandwidth, low processing power, and memory size. There is therefore a need for protocols that deliver sensor data in an energy-efficient way to the sink. One of those techniques, it gathers sensors' data in a small size packet suitable for transmission.

In this paper, we propose a new Effective Data Aggregation Protocol (DAP) to reduce the energy consumption in Wireless Sensor Networks (WSNs), which prolongs the network lifetime. This work uses in-network aggregation approach to distribute the processing all over the aggregation path to avoid unbalanced power consumption on specific nodes until they run out.

Simulation results prove that DAP, compared to other protocols, achieves more data aggregation percentage and less power consumption for a one data harvesting round.

**Keywords**—component; Wireless Sensor Networks, Data Aggregation, network lifetime

### I. INTRODUCTION

Data aggregation is a key technique to minimize energy consumption in large scale information networks. Applying it in radio communication maximizes the network utilization by merging/compressing data coming from different sources into a single frame [1]. There are two approaches to implement the aggregation process. In the first approach, each node within a specific zone, such as a cluster, sends its data packet to a centralized point (aggregator) to be combined and mixed. This approach is called grid based. The second approach combines different nodes data packets while propagating the outcomes of the information sources to the sink(s). It is called in-network aggregation [2].

Data aggregation schemes are generally static, and derived from tree, cluster, or chain based architectures [3]. Cluster and tree aggregation approaches depend on the centralized point, while the chain depends on the in-network aggregation. Most of the contributions in this field focus on data gathering rather than the potential dynamicity of the gathering process.

As we are interested in data mining in wireless networks, we rely on Wireless Sensor Networks (WSNs). They are interconnected and spatially distributed with the help of autonomous devices using sensors nodes to cooperatively monitor physical or environmental conditions (e.g. temperature, sound, vibration, pressure, motion or pollutants) at different locations [4].

WSNs have many limited resources such as limited bandwidth, small memory size, and particularly battery power (or energy). Therefore, it is crucial to design a WSN that saves energy, and consequently prolongs the network lifetime [5].

Our objective is to design a scalable WSN data mining architecture. This architecture has to be able to respond efficiently to queries such as monitoring, control and data collection actions.

In this paper; we propose a Dynamic Aggregation Protocol (DAP) that uses a cooperative wireless network concept. This protocol yields to: i) reduction of the communication overhead needed to build a routing table for each network node and its periodic update as well, ii) making the aggregation path dynamic for each data gathering, iii) prevention of idle and low energy nodes from being involved in the transmission to save their energy for their own activities. The protocol is designed to collect neighbors' data at each stop (along the nodes path to the sink). Aggregated sensors information is supposed to be different. Additionally, in-network aggregation is used as a promising technique to distribute the processing energy over all the network nodes on specific conditions. This will be mentioned later in the section III. However, we restrict the study in this paper to the networking part, which serves as a bearer to the information system architecture.

Performance evaluation for the proposed protocol is compared with two other protocols, namely: i) Data Aggregation using Learning Automata protocol [6], and ii) the conventional routing protocol.

The rest of the paper is organized as follows. Section II presents the related work. Section III describes the proposed protocol model. Section IV shows the simulation results. Finally, section V contains the conclusion and the future work.

### II. RELATED WORKS

The aim of data aggregation protocols is to reduce the energy consumption by gathering, as possible, the sensors' data. Data packets are then manipulated at each aggregator along the aggregation path in order to reduce, merge, and remove redundant information to get a small packet size suitable for transmission efficiently to the sink. Since data transmission is responsible for the most of energy consumption, combining the transmitted packets at each network aggregator yields to the reduction of the transmitted packet size, which saves the network energy.



Data aggregation can be classified into two categories to improve the aggregation ratio<sup>1</sup> [3]. One category focuses on timing control. The other category focuses on establishing a proper routing scheme to improve the aggregation ratio. The most famous approaches for aggregation protocols are cluster, chain, and tree based.

The cluster based approach organizes the sensor nodes into zones (clusters). Each zone has a cluster head responsible of receiving the sensed data from the cluster members (sensors), and sending it to the sink directly or through the other clusters heads in the network. Examples of this approach are Low-Energy Adaptive clustering Hierarchy (LEACH) [7], Hybrid Energy-Efficient Distributed clustering (HEED) [8], and Two-hops clustering algorithm with a composed metric for wireless sensor networks [9].

The chain based approach organizes the sensors nodes into a chain taking the shortest path along which packets are sent to the sink. The chain can be constructed by employing a greedy algorithm or the sink can determine the chain in a centralized manner [5]. An example of the chain-based data aggregation protocol is Power-Efficient Gathering in Sensor Information Systems (PEGASIS) [10], which employs the greedy algorithm to construct the chain.

The tree based approach organizes the sensor nodes into a tree, where the aggregation is performed at intermediate nodes along the tree, and a concise representation of the data is transmitted to the root node, which is usually the sink. The sources are located at the leaves of the tree, and the parent nodes are responsible of the data aggregation. Examples of this approach are: TRee based Energy Efficient Protocol for Sensor Information (TREEPSI) [11], and Energy-aware distributed heuristic (EADAT) [12].

The work of this paper is compared to data aggregation using Learning Automata protocol [6], and the default routing protocol using the shortest path to the sink. Learning Automata protocol relies on gathering similar sensors' data only, and it consists of two phases: route discovery and route selection. The route discovery phase is used to build the routing table of each network's node using a broadcast packet pass through all the nodes. It is expected to need a periodic update for these tables. The route selection phase is used to select a route among the available routes to maximize the aggregation ratio in the network. These available routes consider the shortest discovered paths to the sink, and have the same length. In case a node has no similar neighboring sensors to the sink, it chooses a neighboring node with more residual energy to forward its packets as a repeater and not as an aggregator.

One of the aggregation protocols challenges in WSNs is the data compression. A few literatures discuss the compression of multiple data types in WSN. One of these recent techniques uses adaptive arithmetic coding with low updating cost that could achieve approximately 54% data compression [13].

### III. DAP: DYNAMIC AGGREGATION PROTOCOL

#### A. Network Model

The proposed network model consists of randomly distributed sensors on a rectangular field  $G(V,E)$ , where  $V = \{v_1, v_2, \dots, v_n\}$  is a finite set of nodes and  $E = \{e_1, e_2, \dots, e_n\}$  is a finite set of links. Each node is aware of its location and capable of sensing, transmitting, receiving, and aggregating data. Fig.1 shows the network model with a sample of 100 nodes randomly distributed in the plan area. However, the actual simulation scenarios use a network with up to 500 nodes. The simulation model uses one sink at the end of the network.

In the simulation, different sensor types are used. Each sensor is responsible of a specific parameter (pressure, temperature...etc) and each set of these types has a trigger plan to send its data according to the applications needs.

The used nodes have limited energy and transmission range. Therefore, the neighbors of each node are different depending on its location within the network. These nodes transmit their sensed information periodically to the sink. Fig.2 shows the links between neighbor network nodes as dashed lines.

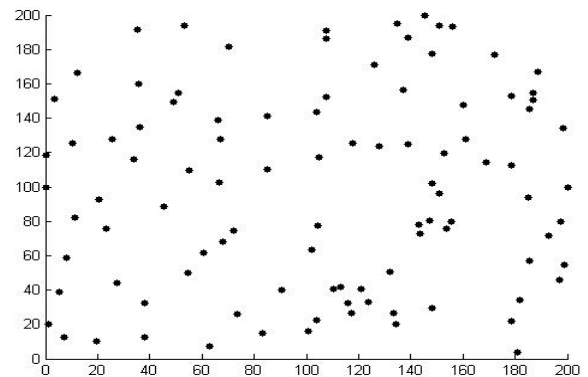


Figure 1. Network model with 100 sensor nodes

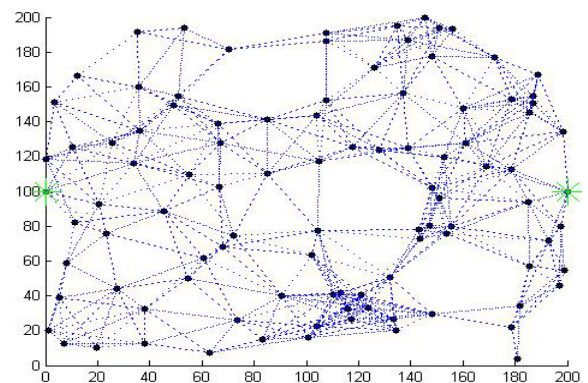


Figure 2. Link connections between neighbors

<sup>1</sup> Aggregation ratio is defined as the ratio of the number of packets the applications generated to that the sink received.

## B. Algorithm Description

DAP is considered a dynamic protocol as the aggregation path is dynamically selected according to the contention process. Consider a node having a sensed data and needs to send it to the sink.

Step 1: the initiator of the aggregation process is the first sensor having the data to be sent to the sink.

Step 2: this initiator node broadcasts a message asking all the surrounding nodes: “*who can be my next hop to the sink?*”.

Step 3: surrounding nodes receive this message and start a contention process to elect one of them (contention winner) to handle the transmission of the source node packet.

Every node receiving this contention announcement must conform to the following conditions to participate in the contention process:

- It must have data to be sent to the sink even if it is of different sensed information type. This conserves the energy of the idle nodes to its activity only.
- It must have enough residual energy for processing and transmission. This is to make sure the intended node is capable of finishing this task successfully.
- Distance from the sink to the node willing to participate in the contention process must be less than that to the source node, and can be determined as in [14]. This makes the path as short as possible and prevents looping. Each candidate can decide whether to get into the contention or not by calculating the Euclidean distance using (1):

$$P(S, D, F) = \text{dist}(F, D) - \text{dist}(S, D) \quad (1)$$

where S, D, and F represent the source, the destination, and the next forwarding node respectively. If the value of (1) is negative, so the forwarding node is closer to the sink than the source, otherwise it is farther from the sink and will not get into the contention.

- This node must check its Link Quality Indication/Energy Dedication (LQI/ED) to make sure it receives the source node signal with a good SNR, which alleviates the need for retransmissions.

After fulfilling these conditions:

Step 4: each conformed node starts a counter according to (2), and the first two nodes whose counters expires send a broadcast message informing the source node and neighbor nodes. The first contention winner will be considered as the next forwarding node (next hop), and the second winner will act as a relay for the source in case its data is received corrupted or lost at the forwarding node.

$$T_R = \left( \frac{T_{\max}}{L + E} \right) - \text{rand} \left( \frac{T_{\max} / 2}{L + E} \right) \quad (2)$$

where  $L$  and  $E$  are the LQI/ED and the residual energy.  $L$  and  $E$  are normalized  $\in(0,1)$ .  $T_{\max}$  represents the maximum delay time that the source node has to wait for a next hop node to

answer. The  $\text{rand}(x)$  function generates a random value between 0 and  $x$  to avoid having more than one winner with equal values.

Step 5: when the nodes in the vicinity receive these two winner messages, they stop their counters. If there are nodes outside the communication range of the winners, they will not receive their broadcast messages. However, these nodes will eventually detect this election when the source node starts sending its data packet to the contention winner.

Step 6: the neighbor nodes stop their counters, and they are aware now of the winner. It will be considered as the data aggregator for this vicinity. These nodes will start sending their data to the winner as a next hope to the sink.

Step 7: the winner starts processing the received packets. Using statistics, it combines, compresses, and removes redundancy ...etc to fit them into one small packet suitable for transmission.

Step 8: the winner node becomes now the new source and needs to pass its packet to the sink. It repeats steps 2 to 7 until the data is delivered to the next hop. The whole process repeats until the packet reaches the sink.

DAP algorithm behaves like a train moving from one station to another (source to next hop). As the train moves, it advertises its location to the vicinity nodes. So every node having data can forward it to this temporary train stop until reaching the last station (sink).

DAP algorithm can be easily practically fulfilled for the following reasons:

- As the sensor networks consist of thousands or at least hundreds of nodes, there are for sure neighbors for each source node. It is also expected that a number of them will have enough energy for the data processing and transmission.
- Keeping the idle nodes silent (according to their triggering time) conserves their energy, and prolongs network lifetime.
- Determining nodes position with respect to the sink prevents loops and selects the shortest path to the sink. Many approaches as in [15] propose algorithms to identify the node location relative to the sink.
- Ensuring a good communication link between the source and neighbors participating in the contention prevents retransmissions.

## C. Data Recovery

Since wireless sensors suffer from low power, the transmitted data maybe received corrupted or lost by the next hop (contention winner). To guarantee the reliability of data transmission, a new technique is proposed. After sending its data, the source node waits for an acknowledgment for a timer 2 period. If the next hop receives the data without errors, it sends an ACK within the timer 2 period and starts processing the data. On the other hand, if the next hope receives the data corrupted, it replies with a broadcast error message. Consequently, the source node waits for a timer 1 period before

trying to retransmit the data. This is because the relay node (the second winner) may have already received the source node's data correctly. In this case, as the relay node follows the same conditions of step 3 within timer 1 duration as in Fig.3 (a). Moreover, the relay node will not transmit the source packet only, instead it will transmit an aggregated packet containing both the source node's data and its own data. In case that no neighbors reply within the timer 1 period, the source node will know that there is no relay that can serve instead of it and will retransmit its data after timer 1 expiry as in Fig.3 (b).

#### D. Comparing DAP with Learning Automata

- DAP sets up each node along the aggregation path (shortest path) as a temporary station for collecting sensors' data, which increases the aggregation ratio. On the contrary, Learning Automata collects packets of the path's nodes only.
- DAP provides aggregation for different sensors' data types that maximizes the aggregation ratio, while Learning Automata focuses to similar nodes data only.
- DAP establishes the shortest path to the sink in parallel with the aggregation process i.e. it performs these two functions (path discovery, and aggregation process) in one step, while Learning Automata performs the both function in two phases.

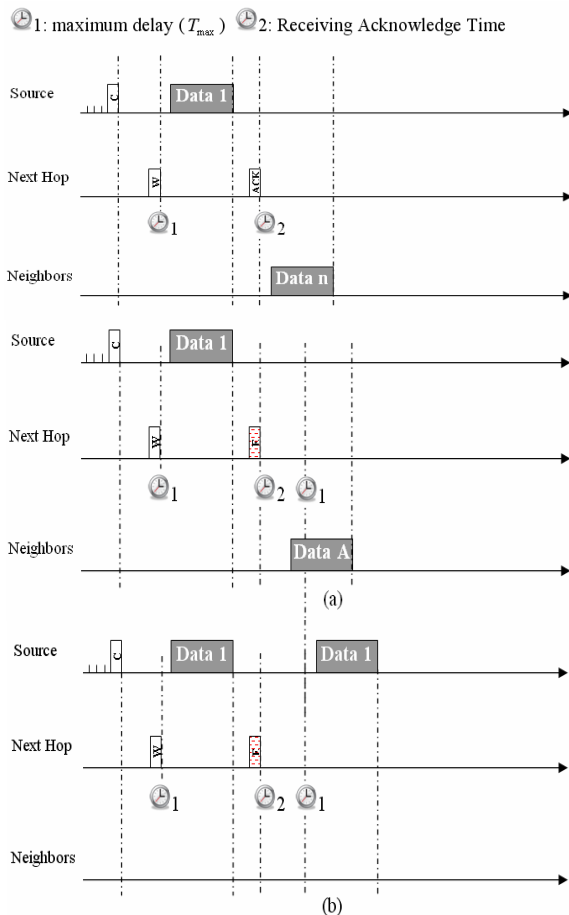


Figure 3. Data recovery

## IV. PERFORMANCE EVALUATION

This section is divided into three parts. The first part for the simulation model, which contains the network specification and the different protocols used to compare in between. The second part evaluates the performance of DAP, and discusses the simulation results. Part three evaluates the analytical model of DAP. MATLAB R2011a was used for the simulation.

### A. Simulation Model

A network model of 100 up to 500 nodes was created on a square plan of 200m x 200m. The network contained one sink. Using only one sink does not prohibit the system from its generality. This is because even if we have more than one sink, we already use the shortest path to the closer sink available. Network nodes are randomly distributed and have a limited communication range. Consequently, it is expect that not all the nodes are neighbors to each other (no direct link in between).

We will show the results in the next part and compare between three relevant protocols: the proposed protocol DAP, the data aggregation protocol "Using Learning Automata"[6], and the conventional routing protocol using the shortest path between a source and the sink.

Real values are used in the following simulation scenarios for sensor bit rate, time needed for transmission, and power calculations [16].

### B. Simulation Results and Discussions

Experiment 1: in this scenario, the effect of the number of network nodes on the aggregation ratio was studied for the three algorithms. Fig.4 shows that the number of aggregated nodes for DAP is more than for the other two protocols. It is also clear from the figure that the aggregation ratio increases with the number of network nodes in case of DAP, while it remains almost constant for the Learning Automata and the default routing protocols. This is due to that Learning Automata focuses on discovering the shortest path, and hence the average path size will not change due to the fixed network dimension. As for the conventional routing, it achieves the lowest behavior due to delivering the nodes packets independently without any aggregation.

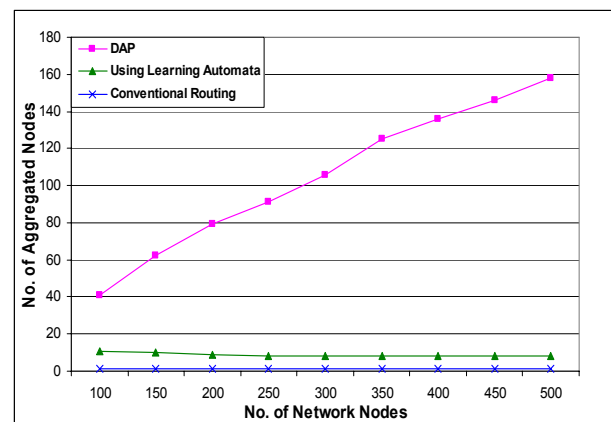


Figure 4. The aggregation ratio for the three algorithms

Experiment 2: in this experiment, we calculate the power consumption per node for each algorithm. Because there is a neighbor discovery phase for building routing tables in both the Conventional Routing and Learning Automata protocols, the power consumption per node is direct proportional to the number of network nodes. On the other hand, DAP discover its routes while transmitting and does not build any routing tables, so it is more efficient in power budget with respect to number of aggregated nodes as shown in Fig.5.

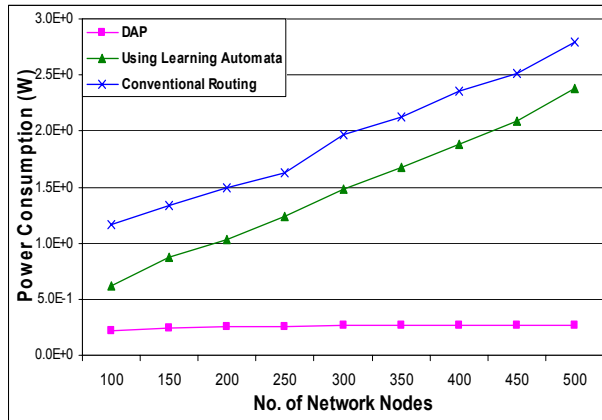


Figure 5. Power consumption per node

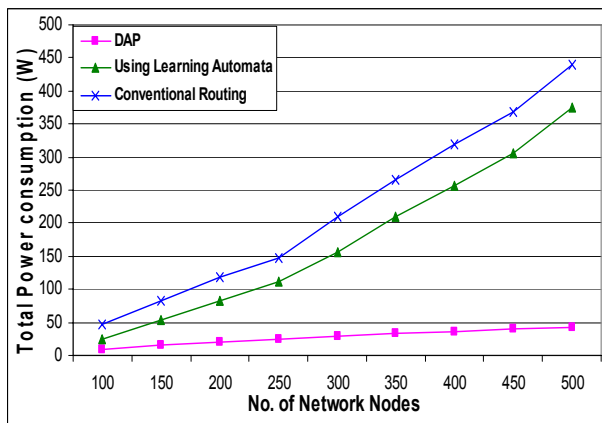


Figure 6. Total power consumption for each algorithm

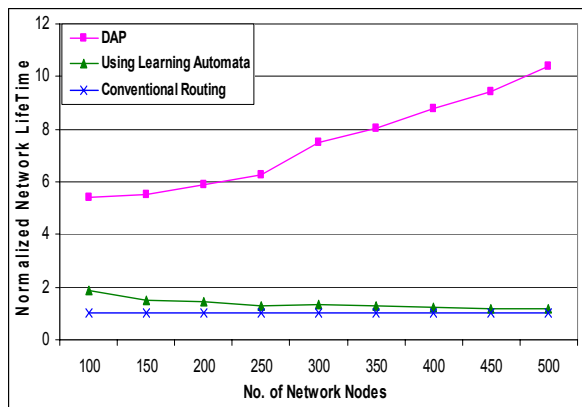


Figure 7. Network lifetime

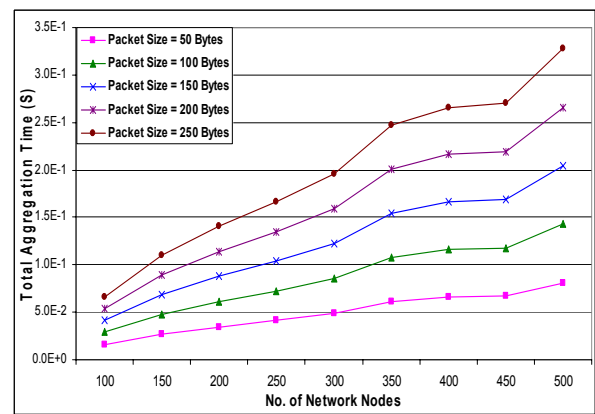


Figure 8. Aggregation time with respect to the packet size

Experiment 3: in this scenario, the total power consumption of each protocol was calculated to determine the protocol that will provide more power saving with respect to the number of network nodes. Fig.6 shows that Learning Automata protocol achieves the lowest power consumption. However, DAP collects data from a number of nodes more than the Learning Automata protocol due to the higher aggregation it achieves as proved in experiment 1. Taking this into consideration, DAP achieves the best efficient power consumption among the three protocols with respect to the aggregated data packets.

Experiment 4: the objective of this scenario is to compare between the three protocols with respect to the network lifetime. Considering the lifetime of the conventional routing is the default (the normalized lifetime), results are achieved by calculating the power consumed by each protocol with respect to number of aggregated nodes. Fig.7 shows that the normalized network lifetime of DAP is higher than the two other protocols.

Experiment 5: since applications could be sensitive to time delay, the aggregation time<sup>2</sup> is a vital factor to be studied with respect to the packet size. Therefore, this scenario calculated the aggregation time (in seconds) for DAP with respect to the sensors packet size. Fig.8 shows that, as the packet size increases, more time is required to collect and aggregate the packets from the neighbor nodes before transmitting them. Note that MAC layer delays and carrier sense mechanism were taken into consideration in the simulation program for calculating time needed for data gathering.

Experiment 6: the objective of this scenario is to compare between the number of aggregated nodes for the three protocols corresponding to percentages of sensor data similarities (100%, 50%, and 33%). For the Learning Automata protocol, the longest aggregation path for the three supposed percentages was used to increase the protocol aggregation ratio. As shown in Fig.9, DAP achieves superior performance even when the aggregation process occurs between the 33% of network nodes.

<sup>2</sup> The aggregation time is defined as the time allowable at each path's node for collecting neighbor's data, and it depends on the application type (i.e. real or non-real time).

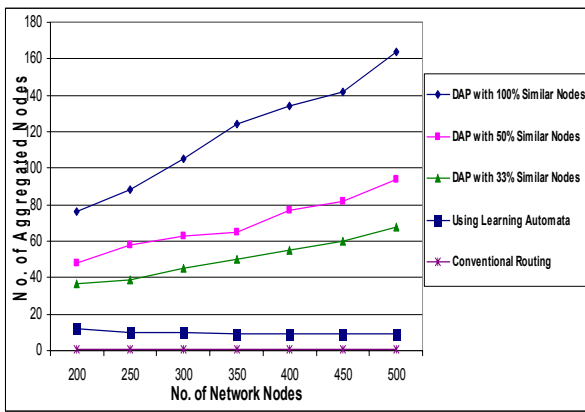


Figure 9. Comparison between the aggregation capability of the three algorithms in case of different similar nodes percentage

In [6], the Learning Automata results and performance were compared with three other protocols: i) LEACH [7], ii) method given in [17] with no learning, and iii) method given in [18] with Q-learning. Based on their results, we can consider DAP as a better protocol compared to these three protocols as well.

### C. Evaluation of The Aggregation Ratio Analytically

We applied some simplifications so as to have a rapid calculation tool that gives an indication of the probable gain when using DAP aggregation technique.

## V. CONCLUSION

In this paper, we proposed a new data aggregation protocol to achieve energy-efficient utilization for WSNs. The main energy consumption reason is the transmission power. Data aggregation presents the optimum solution to decrease it by collecting sensors' data into one packet and transmitting it to the sink. The problem is how to collect sensors' data packets from the vicinity. There are many approaches to this point depending on the network topology (cluster, tree, and chain). DAP is considered a compromise of all these methods.

From the simulation results, DAP achieves superior performance compared to the conventional routing, and data aggregation using Learning Automata protocols. From the simulation figures, DAP achieves better energy consumption, and consequently extends network life time. Even assuming data similarity of 50% and 33% among nodes, we achieve aggregation with better performance than the other protocols. The simulation results prove also that the aggregation ratio increases as the number of nodes and their density in the network increase compared to other solutions that are not affected with these parameters.

## REFERENCES

[1] J. Jeong, J. Kimt, W. Cha, H. Kim, S. Kim, and P. Mah, "A QoS-Aware Data Aggregation in Wireless Sensor Networks," The 12th International on Advanced Communication Technology (ICTACT), vol. 1, pp. 156–161, 2010.

[2] M. Watfa, W. Daher and H. Al Azar, "A Sensor Network Data Aggregation Technique," International Journal of Computer Theory and Engineering, vol. 1, No. 1, pp. 19-26, April 2009.

[3] J. Zhang, Q. Wu, F. Ren, T. He, C. Lin, "Effective Data Aggregation Supported by Dynamic Routing in Wireless Sensor Networks," in Proc. of the 2010 IEEE International Conference on Communications (ICC 2010), pp. 1-6, 2010.

[4] ITU-T Recommendation Y.2221 (2010), Requirements for support of ubiquitous sensor network (USN) applications and services in the NGN environment.

[5] H. Luo, H. Tao, H. Ma, and S. K. Das, "Data Fusion with Desired Reliability in Wireless Sensor Networks," IEEE Transactions on Parallel and Distributed Systems, vol. 22, no. 3, pp. 501–513, March 2011.

[6] M. Esnaashari, M. R. Meybodi, "Data Aggregation in Sensor Networks using Learning Automata," Wireless Networks, vol. 16, no. 3, pp. 687-699, April 2010.

[7] W.R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," Proc. 33rd Ann. Hawaii Int'l Conf. System Sciences, vol. 2, January 2000.

[8] O. Younis, and S. Fahmy, "Distributed Clustering in Ad-hoc Sensor Networks: A Hybrid, Energy-Efficient Approach," IEEE INFOCOM, 2004.

[9] T. Aguilar, H. Afifi, "Two-Hops Clustering Algorithm With a Composed Metric for Wireless Sensor Networks," 11th WPMC 08. Finland, September 2008.

[10] S. Lindsey, C. Raghavendra, "Data Gathering Algorithms in Sensor Networks Using Energy Metrics," IEEE Transactions on Parallel and Distributed Systems, vol. 13, no. 9, pp. 924–935, September 2002.

[11] S. S. Satapathy, N. Sarma, "TREEPSI: TRee based Energy Efficient Protocol for Sensor Information," IFIP International Conference on Wireless and Optical Communications Networks, 2006.

[12] M. Ding, X. Cheng, and G. Xue, "Aggregation Tree Construction in Sensor Networks," VTC 2003-Fall, IEEE 58th, vol. 4, pp. 2168–2172, 2003.

[13] T. Srisooksai, K. Kaemarungsi, P. Lamsrichan and K. Araki, "Energy-Efficient Data Compression in Clustered Wireless Sensor Networks using Adaptive Arithmetic Coding with Low Updating Cost," International Journal of Information and Electronics Engineering, vol. 1, no. 1, pp. 85-93, July 2011.

[14] J. A. Sanchez, R. Marin-Perez, and P. M. Ruiz, "BOSS: Beacon-less on demand strategy for geographic routing in wireless sensor networks," in MASS 2007: The 4th International Conference on Mobile Ad-hoc and Sensor Systems, pp. 1–10, 2007.

[15] T. Aguilar, M. C. Ghedira, S. Syue, V. Gauthier, H. Afifi, "A Cross-Layer Design Based on Geographic Information for Cooperative Wireless Networks," IEEE Vehicular Technology Conference, pp. 1–5, 2010.

[16] Moteiv Tmote hardware products, <http://sentilla.com/files/pdf/eol/tmote-sky-datasheet.pdf>.

[17] J. Beaver, M. A. Sharaf, A. Labrinidis, and P. K. Chrysanthos, "Location-Aware Routing for Data Aggregation in Sensor Networks," Proc. of the 2nd Hellenic Data Management Symposium, 2003.

[18] P. Beyens, M. Peeters, K. Steenhaut and A. Nowe, "Routing with Compression in Wireless Sensor Networks: a Q-learning Approach," In 5th European Workshop on Adaptive Agents and Multi-Agent Systems (AAMAS 05), France, 2005.