Intelligent communication of WSN based on a multi-agent system

Mostefa Bendjima [1] and Mohamed Feham [2]

[1] University of Tlemcen, STIC Laboratory, Tlemcen, Algeria

[2] University of Tlemcen, STIC Laboratory, Tlemcen, Algeria

Abstract

The abstract A wireless sensor network (WSN) consists of a large number of sensor nodes with limited batteries, which are deployed randomly over an area to collect data. Therefore, it is important to minimize energy consumption of each node, which leads to the extension of the life of the network. As many of data that detected could be redundant or not important, optimization for data collection could be a good technique for saving energy in sensor nodes to extend the network lifetime. Our goal is to design an intelligent WSN that collects a maximum data and processes them intelligently. To achieve this goal we used a system of multi-agent (SMA) to process data, reduce redundancy, eliminate non-useful data and establish cooperation between sensor nodes. We also used mobile agents (MAs) to collect this data and send them to the base station (Sink). Due to the mass of data, and in order to reduce the messages sent between sensor nodes, the network is decomposed into clusters. In addition, nodes cooperate with their neighbors in order to collect the maximum data and eliminate redundant data between sensor nodes. Successive simulations in large-scale WSNs show the ability of the proposed data collection system to extend the life of the network in terms of energy consumption and packet delivery rate.

Keywords: *wireless sensor network, data aggregation, multi-agent system, data processing, communication, clustering.*

1. Introduction

The technological advances made in the field of wireless networks, micro-fabrication and the integration of microprocessors have created a new generation of large-scale sensor networks adapted to a range of varied applications.

A wireless sensor network (WSN) consists of a set of nodes capable of collecting data from a monitored area and transmitting them to a base station (Sink) via a wireless medium. The sensor nodes that we consider are small objects equipped with modules for the communication and acquisition of data, particularly in energy constrained areas. These objects need to save their energy while observing their environment properly. Nodes are constrained in terms of energy resources, and therefore, they cannot emit data directly to the base station. Hence, it is essential to use communications and routing protocols to conserve energy during data transfer to increase the lifetime of the network. It is assumed that WSNs are intelligent, independent, and knowing the context in which they operate. A WSN is often characterized by dense deployment in large-scale environments limited in terms of resources. The limitations are in terms of processing, storage and energy capacity mainly because they are usually powered by batteries. Constraint on the size of a sensor node requires designers to reduce the size of the battery and therefore the amount of energy available. Replacing a battery is rarely possible, for reasons of cost or constraints due to the environment. It is widely recognized that energy limitation is an unavoidable issue in the design of WSNs because it imposes strict constraints on the operation of the network. In fact, energy consumption has become the predominant criterion of performance in this area. If we want the network to function satisfactorily as long as possible, these energy constraints force us to compromise between different activities, both at the node and network levels.

WSNs have given rise to many research issues to improve the performance of the network, including the maximization of their life. WSNs-related literature recognizes that the data is a prominent consumer of energy, the majority of this work has stretched to techniques involving this field. One technique for minimizing energy consumption is the technique of cluster; it is used to partition the network into groups with a Cluster-Head (CH) for each. This latter supports data exchange with the base station, and receives sensored data from all nodes in their cluster to be sent to the Sink.

During the last decade, systems of multi-agent (SMA) have greatly developed, and are applied to various fields such as, simulation and artificial life, robotics, image processing. Ant, spider colonies, etc., are examples of SMA, which are applied in WSN for sensored data processing, routing, detection of shortest paths, etc. Recent works have focused on the use of the paradigm of multi-agent systems to allow a more generic modeling, and to describe more easily new types of sensor nodes. We used this kind of system for managing intelligent communication between the sensor nodes in order to increase the lifetime of the network. Eventually, mobile agents (MAs) have even been integrated in the WSN to collect data and send it to the base station (Sink).

Our goal is to design a network that collects a maximum of data, and minimizes the energy dissipation. This network can be defined as a group of agents able to interact and cooperate to achieve a specific goal.
To achieve this objective, we have integrated, into each sensor node, an agent to process data, reduce redundancy, eliminate non-useful data and enable cooperation between sensor nodes. Each node in the network is seen as an autonomous agent with its own characteristics and attitudes towards the various events they receive. In addition, nodes cooperate with their neighbors in order to collect the maximum data and eliminate redundant ones.

This solution is useful for intelligent processing of data collected by the nodes in terms of redundancy reduction, assessment of the importance of data, and the elimination of non-relevant ones. For more efficiency, we have implemented the clustering technique for better organization, and to send data easily to the Sink. We applied the algorithm Low Energy Adaptive Clustering Hierarchy (LEACH) to decompose the network into clusters, each with a head.

The rest of the paper is organized as follows: Section two provides an overview of the literature in which techniques about effective approaches to save energy in WSN are cited. Section three presents the communication strategy that is based on the grouping of nodes into clusters. In the following section, we present the mobile agent packet structure. The main roles of nodes (agents) during the data collection are described in Section five. Then, in section six we present the parameters used by each node (agent) to calculate data relevance, and to cooperate with other nodes during data collection. Then, Section seven sets forth the purpose of our application, i.e. to establish a system to simulate the communication between a set of sensors and a base station forming a wireless sensor network. Section eight has been devoted to results and their analysis. Finally, Section nine summarizes and concludes this paper.

2. Previous work and problem formulation

Recent advances in wireless communications and electronics have enabled the development of smaller and cheaper sensors. A sensor network is composed of a large number of sensor nodes deployed in a geographical area and that communicate with each other through a wireless medium. The aim behind their use is to gather data from the environment to send them to a base station in order to perform calculations. This gathering must take into account the battery life of each node to maintain network continuity. However, advances in WSN technology enabled the deployment of large amounts of sensors that are smaller and cheaper. The energy constraint is important for data aggregation in WSN, in order to reduce the messages exchanged in the network. The data aggregation techniques have recently been studied, and effective approaches to save energy in the WSN have been developed.

Most efficient proposals are based on the traditional Client/Server (C/S) approach, to manage multi-sensor data fusion in the WSN. Several studies [1]-[8] were performed to optimize the architecture of this model. In this architecture, when a sensor node detects data from the environment, it sends them in their raw state through other Sink nodes to be processed. The transmission of raw data does not eliminate unnecessary or redundant ones, which is energy-costly.

A number of papers have proposed algorithms for data Compression/Decompression (C/D) to reduce the amount of the data transmitted by the sensors. The authors of [2], [3] proposed a correlation algorithm that compresses data in a distributed way. In this proposal, a single node is elected to send raw data to the Sink while the others send coded data only. After the Sink receives data, it decodes them through the correlations between the compressed and uncompressed data. However, it is quite difficult to find a non-complex and energy-efficient coding algorithm suitable for the sensor nodes.

In [4], the authors proposed Data Fusion (DF) of a maximum number of sensors. When a node sends its data to the Sink, the intermediate nodes fuse their data with others coming from the first node. Therefore, this data is merged into a single message instead of many, thereby saving energy. However, the intermediate nodes do not always have to send important data, and do not eliminate the unimportant or redundant ones. In addition, the authors failed to consider the importance of the scalability of such networks.

Other researchers [6] have shown that clustering is a fundamental technique in WSNs. Their objective is to minimize data aggregation processing required at sensor nodes and to move the load to the Sink. Heinzelman and al. [7] proposed a clustering algorithm in which the sensors elect themselves as cluster heads with some probability and diffuse their decisions. Once the data from each node are received, the cluster head forwards them directly to the base station. In this way, it achieves a significant reduction in energy consumption. Unfortunately, the authors did not address the problem of complexity and the amount of energy required to build such cluster-based sensor networks.

In addition to [5] and [6], the authors of [9]-[12] have also proposed a structured strategy. They set up a multicast tree by iteratively adding source nodes to the existing tree until all source nodes are included. Whenever the algorithm detects a remaining source node, which is the closest to the existing tree, it adds the shortest path between this source node and the existing tree. This process continues until all source nodes are included in the tree. However, according to [11] and [12], structured approaches are not practical for dynamic scenarios, due to excessive communication cost and centralized management of the WSN structure.

Inspired by ant colonies, Dorigo *et al.* [13], [14] proposed an Ant Colony Optimization (ACO) algorithm to solve the problem of data aggregation. Every ant explores all possible paths from the source node to the Sink. In the papers [13]-[15], the authors proposed the idea of an agent ant that combines the mobile agent technology with ACO algorithms for network control in the choice of adaptive trace by a data aggregation tree that is built by accumulated pheromone. However, the construction of this appropriate tree depends largely on the deployment of nodes, which is usually random, and consumes a significant amount of energy. As the communication range of a node is limited, the nodes can communicate only with their neighbors in a hop, hence the Euclidean distance between the source node and the receiving node is not reliable.

Researchers have proposed to use system of multi-agent (SMA) as a solution for the adaptation of distributed and complex WSNs. The SMA is a group of agents that interact and cooperate to achieve a specific goal. SMA is well suited to distributed control systems such as WSN. This partly explains the considerable contribution of agent technology when it was introduced in this area.

Researchers [15] have made an intelligent artificial community with intelligent sensors by using this system. Intelligent sensor nodes operate as autonomous agents who develop a network. The authors [16] proposed a group of agents to interact and cooperate in order to reach a specific objective. They proposed a multi-agent approach in which an agent is put in each sensor node to process locally detected data and to cooperate with neighboring agents to eliminate the inter-sensor-nodes redundancy [17]. This problem seems far from being a solution to the density constraint imposed in many WSN applications.

The authors of [1], [7], [8], [17]-[20] proposed the technology of mobile agents (MAs) in WSNs. In these proposals, the MA program refers to a standalone application, able to move between networks nodes by moving the data collected in each node instead of bringing them in their raw state to the Sink by the node itself. It contains an application code, a list of source nodes predefined by the Sink and an empty field to put the data. The calculation model based on MA moves the processing code for data transfer rather than the raw data to the Sink. In addition, the MA is programmed to perform local data processing and filtering on each sensor node according to the data it already carries. The MA then transports the merged data to the next node, and the same procedure is repeated. Due to the bandwidth limitation and density of WSNs, the MA can be used to significantly reduce the cost of communication, in particular the elimination of redundancy, and the reduction of the network overload associated with data transfer [17].

This approach also involves more reasonable use of radio node unit, thus a longer life of the network. When mobile agents are used for data fusion, route selection for agents becomes critical because it greatly affects the overall energy consumption and the cost of data fusion. The drawback of such solutions is the difficulty of creating a list of source nodes and setting the start time of data collection. Another limitation is the definition of areas to be addressed by the MA.

After analyzing the solutions presented above, we can also infer that there is still a lot of work in terms of energy efficiency with attention to the packet delivery ratio and network density. In addition, the solution should be independent of network deployment.

3. Communication strategy

The main purpose of our strategy is to collect the maximum data in a WSN, with improved network performance, including the maximization of its life.

The step of local data processing consumes much less energy than the communication phase; the example presented in [21] illustrates this disparity. Indeed, the energy cost required for transmitting 1KB over a range of 100m is approximately equal to that necessary to perform 03 million instructions at a rate of 100 million instructions per second. Therefore, it is clear and preferable to promote

IJCSI International Journal of Computer Science Issues, Vol. 10, Issue 1, No 2, January 2013
ISSN (Online): 1694-0814 | ISSN (Print): 1694-0784
www.IJCSI.org

data processing at the node level before their transmission. As many of data that could be detected redundant or not important, optimization for data collection could be a good technique to save sensor nodes energy and extend the network lifetime.

In this work, we propose an intelligent strategy that collects a maximum of data and treats them intelligently. This strategy takes into consideration several parameters to ensure better management of data collection in a WSN.

Among these parameters, we saved energy at each node by reducing the amount of data that is useless and unimportant, and by redundancy elimination.

To achieve this goal we used a multi-agent system to process data, reduce redundancy, eliminate non-useful data and create cooperation between sensor nodes.
Our strategy is described as follows: An agent is introduced in each node, which processes the data locally and judges their importance in order to remove any data that is not useful or redundant. In addition, nodes cooperate with their neighbors in order to collect the maximum data and eliminate redundant data between different sensors. Due to the mass of data, instead of sending data by each node to the Sink separately, we decomposed the network into clusters. Each cluster is composed of Cluster Head, and the data for each cluster are grouped together to be sent. Due to the energy consumption at the highest level of communication, to send an amount of data in a single message is less energy consuming than sending the same amount in several short messages [18]. For this reason, we have proposed mobile agents to concatenate data processed by a node and its neighbors to other nodes in located on the same path to the CH, to send them in a single message.

The main idea is based on grouping nodes in a cluster, and after the Sink elects source nodes, it sends mobile agents to the cluster heads; each MA contains the source nodes of the cluster. Once the CH receives the MA, it will make for each source node in the list a copy of the MA. Then, for each copy of MA is added a list of intermediate sensor nodes between the source node and the CH. The latter will send the mobile agent to the source node so that it flows between the source node and the CH according to the list of intermediate sensor nodes. Each node in the intermediate list invites its neighbors in a single hop to join a session of cooperation for data collection. The MA aggregates the data collected and processed by the list of intermediate nodes and neighboring nodes of each intermediate node in a well-defined strategy.

As shown in Fig. 1, we assume first that during data aggregation an intermediate neighboring node detects data. After processing, it is considered that the neighboring node believes that this data is important and useful. Based on this estimate, it runs a formula defined to make the appropriate decision to cooperate or not. Fig. 1 explains the general scheme of a data collection.

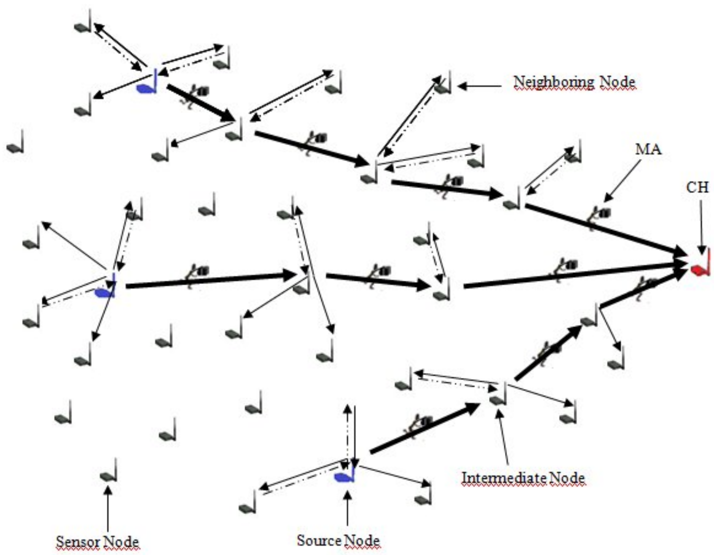

Fig. 1 General scheme of a data collection.

4. Mobile Agent Packet Structure

After the Sink uses direct diffusion to designate the source nodes, it sends mobile agents to the clusters heads; each one containing its source nodes. Our MA is a data packet that circulates in the cluster, and is used to assemble the data collected by the sensor nodes in one cluster. The data contained in an MA packet is shown in Fig. 2.

Sink_ID	MA_SeqNum	CH_ID	Src_List	SN_Nbr
Processing_Code		Data_Cooperation		

Fig. 2 MA Packet Structure.

Both attributes Sink_ID and MA_SeqNum are used to identify a packet MA. Whenever Sink sends a new packet to MA, it increments the MA_SeqNum. Src_List specifies the source nodes of the cluster head CH_ID, which will be visited by the MA. SN_Nbr attribute is the number of source nodes in the cluster. Processing_Code is used to process the data collected and to manage the MA. Data_Cooperation as well as carry aggregated sensor nodes data.
After the cluster head receives a mobile agent, it will make a copy of MA for every node in the source Src_List, so that the number of source nodes equals the number of MAs.

Copies of MA will change when the CH receives the MA packet from the Sink.

For each copy of MA, CH fills the Src_List attribute by nodes that are in the same path between a source node of the Src_List and CH.

Again for each copy of MA, the SN_Nbr attribute is removed, and we add a Src_Next attribute which determines specifically the sequence of node identifiers that must be visited by the MA. If Src_Next is equal to CH, this means it is the last node visited by MA. Fig. 3 illustrates the case of a mobile agent with its copies.

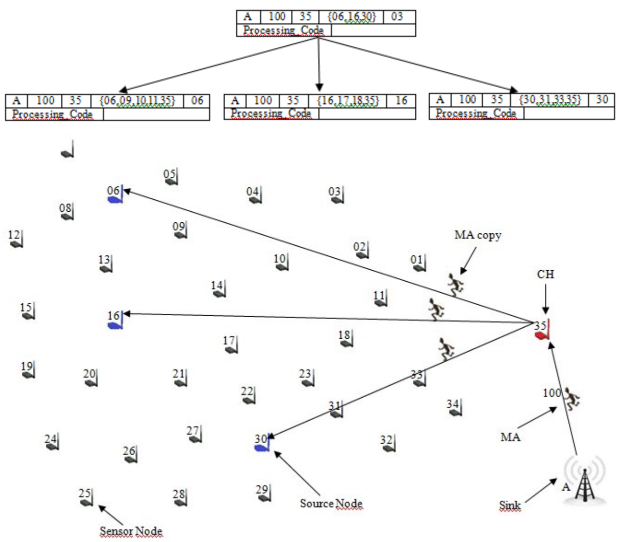

Fig. 3 An example of a MA with copies.

Algorithm 1: Creating copies of a MA at the Cluster-Head level
If CH receives MA **Then**
If MA@ Src_Next <> {} **Then**
For i = 1 **To** MA@ SN_Nbr **Do**
MA[i] = MA; // Make a copy of MA;
Remove MA[i]@ SN_Nbr ; // Delete the attribute SN_Nbr;
Add MA[i]@Src_Next ; // Create the attribute Src_Next;
MA[i]@ Src_Next = MA@Src_List[i];
MA[i]@Src_List = {the intermediate nodes between Src_List[i] and CH};
Sending MA[i] to MA[i]@Src_Next;
EndFor
Else
MA sending to the Sink;
EndIf
EndIf

5. Cooperative data collection

Fig. 4 illustrates the role of sensor nodes (agents) during of data collection sessions, which begins when a source node receives an MA. This source node sends a request for cooperation to its neighbors in a single hop, as shown in step (1). The request for cooperation is a short message, scheduled for a single hop, and the neighbors will be programmed not to replay the message. Thus, a node (agent) neighbor decides to cooperate or not in a very precise formula. After making the appropriate decision, each node sends its cooperating processed data (useful and non-redundant) to the source node. Thus, the source node waits for a fixed time, and then considers that nodes which do not send data are not cooperating. Then, the source node processes its data with the ones transferred by neighboring cooperating nodes to eliminate redundancies. After the elimination of redundancies by the source node, the latter collects its data with data from neighboring cooperating nodes, and sends the MA to the next node in the CH path that is in the next source attribute of MA. These data will be linked in the data attribute of cooperation after the elimination of redundancy, as shown in step (2).

Upon receipt of MA, node (A) sends its request for cooperation to its neighbors in a single hop to collect their processed data, as shown in step (3). After making the appropriate decision, each cooperating node sends its processed data to the node (A). Thus, node (A) waits for a fixed time, then considers that nodes that do not send data are not cooperating. Then, node (A) processes its data with the ones transferred by neighboring cooperating nodes to eliminate redundancies. After that, it sends its data with data from neighboring cooperating nodes in the MA to the next node in the path to CH, as shown in step (4). Node (B), and all intermediate nodes on the path to CH which are found in the list Src_List of MA, repeat the same procedures of preceding nodes until they reach the CH.

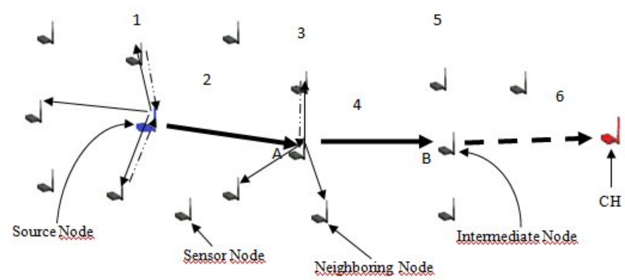

Fig. 4 Cooperation of data collection Proposed beam former.

Algorithm 2: the passage of MA at the intermediate nodes level
While MA arrives at a node not CH **Do**
For i = 1 **to** Number the neighbors of node **Do**
Node broadcasts ReqCoop;
EndFor
If delay = D **Then** // the time out for sent the MA to the next;
For i = 1 to Number of neighbors node cooperating +1 **do** // node + its neighbors cooperating;
If MA is empty **Then**

```
                    N=R1;   // Set the data packet MA;
                    Else
                        If the node data does not exist in the packet MA Then
                        N=N+ ∑ ρ x Ri ; // add data packet MA;
                        EndIf
                    EndIf
            EndFor
            EndIf
MA@Next_Src = read the new destination for MA@ListSrc ;
Send the MA at MA@NextSrc ; // node same path to CH
End.
```

Let N be the amount of sensored data accumulated after the MA collects the result of a node, and Ri the size of the data to be processed sensored and accumulated locally by the MA to node i. Hence, we have: $N = R_1$; $N = N + \sum_{k=2}^{i} \rho \times R_k$ (i >=2).

Algorithm 3: Cooperation of a neighboring node

```
If node receives ReqCoop Then
    If node contains important data Then
    Calculate P;   // the Pertinence of cooperation;
        If P > S Then   // if the pertinence > to a predefined threshold;
        Transfer data;
        EndIf
    EndIf
EndIf
```

6. Agent Strategy

We present the parameters used by each node (agent) to calculate the pertinence of cooperation during a data collecting session (P). These parameters are Energy (E), the Distance from the cluster head (D), and the degree of Importance of the data (I).

We express the parameters defined by equation (1) to calculate the pertinence (P) of cooperation. Each parameter is multiplied by a coefficient which is the impact factor according to its importance. Hence, according to a predefined threshold (S), the neighbor node decides to cooperate (P > S) or not (P < S).

$$P = Ce \times E + Cd \times D + Ci \times I \qquad (1)$$

Where, Ce, Cd, and Ci are energy coefficients, the distance, and the degree of importance of data respectively.

6.1 Energy (E)

The energy E is only the remaining battery level. The more one node has energy; the more it is requested to cooperate. A low battery level decreases the value of P. In this case, for the agent to cooperate, other parameters have to be at their highest.

6.2 Distance (D)

A head is elected for each group and each node within a group is identified by a particular subnet address. Each elected leader transmits to each node in the group a distance vector separating the group leader. When a node is not close to the head, it is requested to participate in cooperation, but if it is closer, it might lose energy by cooperating with several nodes.

6.3 Degree of Importance of data (I)

The degree of importance of data depends heavily on the desired application. This parameter is calculated by local processing in the node, after the data are gathered. The data processing at the node level can estimate the degree of importance of the data collected. Such data is considered a priority node if it is the first data containing the desired information. In another example, for temperature monitoring, the node saves the last data collected to compare with new ones gathered. If the difference between both is greater than a predetermined threshold, we consider that the data is important. However, if the difference increases, we consider that this data has a higher priority, so the degree increases.

7. Simulation setup

The goal of our application is to simulate the communication between a set of sensors and a base station within a wireless sensor network. Our technique is based on partitioning the network into clusters and on using multi-agent systems and mobile agents as a mechanism to save energy. Thus, we will make an application in terms of efficient energy consumption and packets delivery rate, for a node to call its one hop distant neighbors in order to cooperate during a data collection session.

We performed our simulations on a 500m x 500m area with a random distribution of 1000 sensor nodes in 1000 seconds. Thus, we used a single base station that is located to the right of the field, and some source nodes are randomly distributed in the network. We have limited the radio range and the data bit rate of each node to 80m and 1Mbps, respectively, as suggested in [16]. Local processing time is 40ms. The parameters of power transmission and reception, which directly affect the radio range, are selected from the ranges defined in [17].

Table 1: Basic simulation parameters

Simulation parameters	**Values**
Node distribution	Random
Radio range	80m
Bit rate	1Mbps

Sensored data interval	10 seconds
Simulation time	1000 seconds
Local processing time	40ms
Processing code size	0.4Ko
Raw Data Size	2024 bits
MA Code Size	1024 bits
MA Accessing Delay	10 ms
Data processing Rate	50 Mbps
Raw Data Reduction Ratio	0.8
Aggregation Ratio	0.9
Fusion Factor (ρ)	01
Ce , Cd , Ci	0.5
Network Size	500m x 500m
Number of Sensor Nodes	1000

8. Results and analysis

Mobile agents have been proposed for efficient data dissemination in wireless sensor networks. In traditional, Client/Server architecture based WSN (CSWSN), data from multiple source nodes are transferred to the base station while the mobile agents can be used to significantly reduce the communication costs, which have a significant impact on the effectiveness of WSNs. In addition, to understand the performance of our strategy, we compare it with our previous work [22] Multi-Agent-based Wireless Sensor Network with Clustering (MAWSNC) where we have used the agent–based strategy. However, it is assumed that the nodes are not agents, that is to say, they do not process data as in our approach. This paper proposes to use a strategy based on a multi-agent system for the reduction and aggregation of a maximum of data in sensor network. The proposed strategy is called a Multi-Agent-based System of wireless sensor networks (SMAWSN).

To demonstrate the performance of our approach in wireless sensor networks, we compare it with other approaches, namely CSWSN and MAWSNC according several criteria that we will describe later.

In our simulations, we assume that the sensor nodes have batteries (energy limited), except the Sink, which is assumed to have an infinite supply of energy. We assume again that the Sink and sensor nodes are stationary and that the Sink is located on the right side of the field. To check the scaling property of our algorithms, we select a large-scale network with 1000 nodes.

In this section, we present the main performance criteria and the evaluation of their effectiveness through simulations:

We examine the impact of the number of source nodes on the criteria of energy efficiency. Therefore, energy consumption is the parameter that defines the lifetime of the wireless sensor network. Therefore, we consider this parameter as the most important criterion, according to which we will evaluate the performance of our approach as shown in Fig. 5.

We use E_{CS}, and E_{MA} to describe energy consumption in CSWSN and MAWSNC, respectively. However, in our approach, we add to E_{MA}, energy E_{TR} for local treatment at the source nodes. We fixed the number of source nodes from 05 to 50 per steps of 05, and get a set of results for each case. We represented the results obtained in Fig. 5, which shows the impact of the number of nodes on energy sources, for obtaining sensory data from all source nodes.

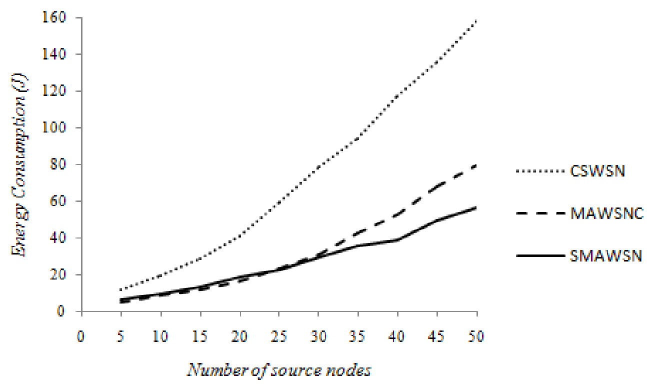

Fig. 5 Comparison of Energy Consumption.

The analysis of the figure above highlights several interesting elements: first, obviously, when the number of source nodes is increased, more energy is needed to perform the duties of each of the three approaches. At the beginning, our approach is always better than CSWSN in terms of energy. However, for an equivalent number of source nodes, the consumption of our approach SMAWSN is always lower than MAWSNC and CSWSN approaches. Moreover, the difference between our approach and the approach CSWSN begins to widen from 05 source nodes, while the gap with other MAWSNC begins at 25 sources nodes. When the number of source nodes is little, energy consumption of our approach is less important compared to the MAWSNC approach, that from 25 source nodes or more, the difference between our approach and the MAWSNC approach becomes increasingly important, and this difference increases continuously with the increase in the number of source nodes. However, from 30 nodes, our approach significantly outperforms the other two approaches; for example with 40 nodes where MAWSNC and CSWSN consume 25% of energy and 67% more than SMAWSN, respectively. However, at 50 source nodes, consumption of our approach is 28.5% lower than in the MAWSNC approach and 64% less compared to the

CSWSN approach. Our SMAWSN approach is therefore more energy efficient.

In addition, and in another experiment, we show the performance comparison of the three approaches in terms of task duration. In a CSWSN, this criterion is the average latency required for carrying messages from source nodes to the base station. In a MAWSNC, the task duration is equivalent to the average end-to-end reporting delay, from the time when a MA is dispatched by the sink to the time when the agent returns to the sink. However, the time includes processing time data by the mobile agent to eliminate redundancy. So our approach for several mobile agents working in parallel, there must be an agent who is the last to return to the Sink, therefore, the duration of the task is the delay of this agent. However, duration includes data processing time at the node level; the longer the cooperation time with neighboring nodes, the more is the mobile agent time. We use T_{CS} and T_{MA} to designate average end-to-end packet delay in CSWSN and MAWSNC, respectively. However, in our approach, we add the T_{TR} time to T_{MA} for local processing at the source nodes level. The results are shown in Fig. 6:

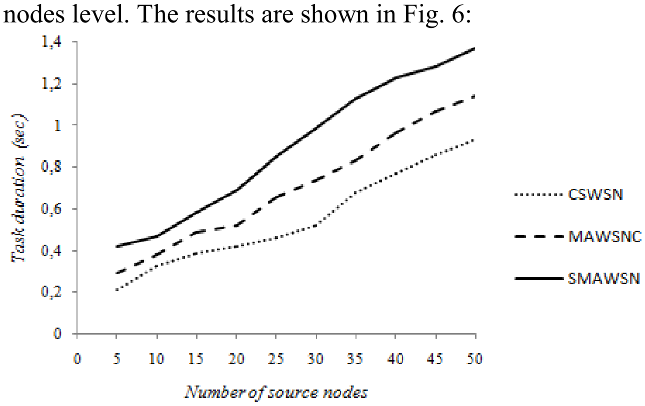

Fig. 6 Comparison of task duration.

In Fig. 6, we show the task duration between the three approaches, and we notice additional latency with SMAWSN in relation to MAWSNC and CSWSN. This gives the false impression that the performance of our approach in terms of duration of a task is poor. However, this is not true, as the duration of a task in the CSWSN approach does not include processing time. If you add the processing time to the latter, we believe that the difference is that in SMAWSN the duration of a task will be relatively small.

This latency is related to the local processing time and the cooperation of neighboring nodes necessary for our approach. We can also notice that the latency difference between SMAWSN and MAWSNC is lower compared to the latency difference between SMAWSN and CSWSN because MAWSNC spends latency time to eliminate redundancies by the mobile agent at each source node, which is not the case with CSWSN which sends the data collected to the base station.

In addition, observing the results, and comparing our approach with CSWSN, which shows the biggest difference, we can notice that values are below 0.45 seconds. This means that the accuracy could be affected for applications that are very sensitive and require less than 0.45 seconds.

In addition, these differences could be explained easily, because with more source nodes to visit, MAs sizes increase and many transmissions will be made with neighboring nodes. However, if we consider the association between these two criteria, the extra time consumption in our approach, shown in Fig. 7, will be easier to understand.

We use E_{CS} and E_{MA} to describe energy consumption in CSWSN and MAWSNC, respectively. We still use T_{MA} and T_{CS} to describe the average end-to-end packet delay in CSWSN and MAWSNC, respectively. However, in our approach, we add energy E_{TR} and time T_{TR} to E_{MA} and T_{MA}, respectively, for local treatment at the source nodes. For this purpose, we adopt the following definitions:

$$E_{CSWSN} = E_{CS} \times T_{CS} \quad (2)$$

$$E_{MAWSNC} = E_{MA} \times T_{MA} \quad (3)$$

$$E_{SMAWSN} = (E_{MA} + E_{TR}) \times (T_{MA} + T_{TR}) \quad (4)$$

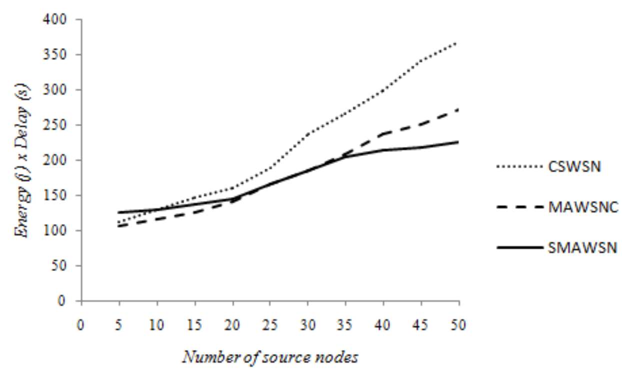

Fig. 7 Comparison of energy x delay.

Fig. 7 compares the performance of the three approaches in terms of energy x delay. Compared to the energy performance, the number of source nodes has a greater impact on this criterion. For time-sensitive applications over energy constrained WSNs, this criterion is defined to facilitate assessment of the overall energy and delay performance of the algorithms. The smaller this value is, the better the performance will be.

First, when the number of source nodes is increased, energy x delay is increased to perform the duties of each of the three approaches. When the number of source nodes is small, energy consumption using our approach is less important compared to other approaches.

It may be noted again that the difference between our SMAWSN proposal and the other two approaches is becoming increasingly important, and this difference increases continuously with the increase of the number of source nodes.

However, for an equivalent number of source nodes, energy consumption using our SMAWSN approach is always lower than that of MAWSNC and CSWSN. Moreover, the difference between our approach and the CSWSN approach begins to widen from 10 source nodes, while the gap with MAWSNC begins at 35 source nodes. Our approach is advantageous with increasing sources nodes; with 50 source nodes; MAWSNC and CSWSN consume 17% and 38.5% more energy than SMAWSN.

In addition, and in another experiment, we changed the size of the data collected at each source node 0.5Ko to 04Ko. The results are expressed in Fig. 8:

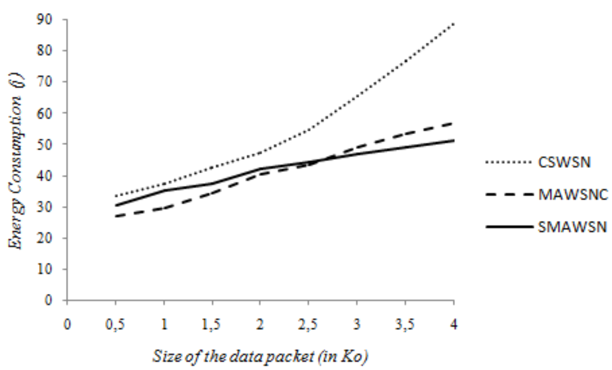

Fig. 8 Energy consumption depending on size.

After analyzing the figure below, we notice that when the packet size is increased, more energy is needed to perform the duties of each of the three approaches.

The first observation is that the difference between our approach and the CSWSN approach begins to widen from 0.5 Ko, while the gap with MAWSNC starts at 2.5 Ko. Moreover, when the packet size is small, energy consumption of our approach is less important compared to the MAWSNC approach, varying between 0.5 Ko and 2.5 Ko, the latter makes additional energy savings of 16% higher than SMAWSN. It may be noted again that from 2.5 Ko the difference between SMAWSN and the two approaches becomes increasingly important, and this difference increases continuously with increasing packet size. However, at 4Ko, energy consumption of SMAWSN is 11% less than in MAWSNC and up to 42% less than CSWSN. This means that the difference in energy consumption gap between our approach and the two others is steadily widening with increasing packet size. By comparison, our SMAWSN proposal solution has better energy efficiency.

9. Conclusion and future works

In an environment where the sensor nodes are close to each other, and where the redundancy is considerable, the sensor nodes generate a large amount of transmission over the wireless channel, which consumes a lot of battery power. Not to send the data in their raw state, as they are captured as in the Client/Server model, and to collect the maximum relevant data, we proposed in this work a solution for data collection using a multi-agent system based on clustering for managing communication between the sensor nodes, and to increase the life of the network. The objective of our work is to compile the maximum possible amount of data from source nodes that are in the same path to the Cluster-Head in a single message.

Our system consists of two types of agents, including stationary and mobile agents. The role of stationary agents consists in integrating an agent in each sensor node, and then each agent processes the locally sensored data by its corresponding sensor node and considers its importance. Then, each source node cooperates with its neighboring nodes in the same cluster to gather their data and eliminate redundancies by a method based on several important parameters for determining the relevance of cooperation. We used as a mobile agent data packet to transmit sensored data to the Cluster-Head to eliminate redundancies between nodes. The result is that a mobile agent gathers data from a source node and its neighboring nodes that are cooperating in the same path to CH. This plan limits communications except for relevant data, and consequently reduces the amount of traffic and energy consumption. Cooperating nodes eliminate redundancies and concatenate all the data processed during a collection session in a single message. This means a gain with regard to amount of data and the load needed to send them. The plan presented is based on a strategy that takes into consideration several parameters deemed important for a longer WSN lifetime. These parameters enable us to determine energy, distance from the CH, and the degree of importance of the data.

Through successive simulations, we have been able to prove that our approach has better performance compared to the Client/Server approach and our previous work [22] Multi-Agent-based Wireless Sensor Network with Clustering (MAWSNC) where we used the strategy of mobile agents. In fact, the only shortcoming of our approach is the latency required to communicate additional data. This latency was required to enable neighboring nodes to cooperate, and can easily be justified by taking into account the gain obtained in terms of packet delivery ratio and energy consumption in dense wireless sensor networks.

As future work, we will attempt to reduce latency to a maximum, and add other parameters to the cooperation formula.

References

[1] M. Chen, T. Kwon, Y. Yuan, and V.C.M. Leung, "Mobile Agent Based Wireless Sensor Networks", University of British Columbia, 2006, pp. 14–21.

[2] J. Chou, D. Petrovic, and K. Ramchandran, "Tracking and exploiting correlations in dense sensor networks", Conference Record of the Thirty-Sixth A silomar Conference on Signals, Systems and Computers, 2002, pp. 39–43.

[3] J. Chou, D. Petrovic, and K. Ramchandran, "A distributed and adaptive signal processing approach to reducing energy consumption in sensor networks", Joint Conference of the IEEE Computer and Communications Societies, 2003, pp. 1054–1062.

[4] S.R. Das, "Serial Data Fusion Using Space-filling Curves in Wireless Sensor Networks", In IEEE, ed. First Annual IEEE, Communications Society Conference on Sensor and Ad Hoc Communications and Networks, 2004, pp. 182–190.

[5] W.H. Liao, Y. Kao, and C.M. Fan, "Data aggregation in wireless sensor networks using ant colony algorithm", Netwerk Computer, 2008, pp. 387–390.

[6] M. Chen, H. Mineno, and T. Mizuno, "Adaptive data aggregation scheme in clustered wireless sensor networks", Comput. Commun, 2008, pp. 3579–3585.

[7] L. Tong, Q. Zhao, and S. Adireddy, "Sensor networks with mobile agents", Multidisciplinary University Reseanh Initiative (MURI) under the Omce of Naval Research, And Army Research Laboratory CTA on Communication, and Network under Grant DAADlP-OI-2-CQII, 2003.

[8] Q. Wang, and Y. W. Xu, "Mobile-agent-based collaborative signal and information processing in sensor networks", Proceedings of the IEEE, 2003, pp. 91-98.

[9] J.N. Karaki, R. Mustafa, and A.E. Kamal, "Data aggregation and routing in wireless sensor networks: Optimal and heuristic algorithms". Computer Network, 2009, pp 945–960.

[10]S. Upadhyayula, and S.K.S. Gupta, "Spanning tree based algorithms for low latency and energy efficient data aggregation enhanced converge cast (dac) in wireless sensor networks", AdHoc Network, 2007, pp. 626–648.

[11] S. Jie, and Y. Ning, "A spanning tree algorithm for data aggregation in wireless sensor networks", IEEE explorer, 2008, pp. 5014–5018.

[12]B. Yu, and J. Li, "Distributed data aggregation scheduling in wireless sensor networks", In IEEE INFOCOM, 2008.

[13]S. C. Chu, J. F. Roddick, and J. S. Pan, "Ant colony system with communication strategies", Information Sciences, 2004, pp. 63–76.

[14]M. Dorigo, and L. M. Gambardella, "Ant colony system: a cooperative learning approach to the traveling salesman problem", IEEE Computational Intelligence Society, 1997, pp. 53–66.

[15]R. Tynan, D. Marsh, and D. Kane, "Intelligent Agents for Wireless Sensor Networks. Adaptive Information Cluster", Smart Media Institute, University College Dublin Belfield, Dublin 4, Ireland, 2004.

[16] A. Sardouk, M. Boulahia, and L. Gaïti, "Information-Importance Based Communication for Large-Scale WSN Data Processing", In: WMNC, Gdansk, Poland, IFIP, 2009.

[17]A. Sardouk, M. Boulahia, and L. Gaïti, "Agent-Cooperation Based Communication Architecture for Wireless Sensor Network", In: IFIP/IEEE Wireless Days/Adhoc and Wireless Sensor Networks, UAE, IFIP, 2008.

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

[19] Q. Wu, N.S.V Rao, and J. Barhen, "On computing mobile agent routes for data fusion in distributed sensor networks", IEEE Transactions on Knowledge and Data Engineering, Vol.16, NO. 6, 2004, pp. 740-753.

[20]G. Vigna, "Mobile Agents: Ten Reasons For Failure Proceedings", Berkeley, CA, 2004, pp. 298-299.

[21]J. Gregory, and W. J. Kaiser, "Wireless integrated network sensors", Communications of the ACM, 2000, pp. 43-47.

[22]M. Bendjima, M. Feham: "Intelligent Wireless Sensor Network Management Based on a Multi-Agent System", IJCST, 2012, pp. 9-14.