future internet

MDPI

Article

Intelligent Communication in Wireless Sensor Networks

Mostefa Bendjima * and Mohammed Feham

STIC Laboratory, University of Tlemcen, P.O. Box 230, Chetouane, Tlemcen 13000, Algeria; feham_m@hotmail.com
* Correspondence: mostefa.mostefa@yahoo.fr

Received: 20 July 2018; Accepted: 10 September 2018; Published: 15 September 2018

Abstract: Wireless sensor networks (WSN) are designed to collect information by means of a large number of energy-limited battery sensor nodes. Therefore, it is important to minimize the energy consumed by each sensor, in order to extend the network life. The goal of this work is to design an intelligent WSN that collects as much information as possible to process it intelligently. To achieve this goal, an agent is sent to each sensor in order to process the information and to cooperate with neighboring sensors while mobile agents (MA) can be used to reduce information shared between source nodes (SN) and send them to the base station (Sink). This work proposes to use communication architecture for wireless sensor networks based on the multi-agent system (MAS) to ensure optimal information collection. The collaboration of these agents generates a simple message that summarizes the important information in order to transmit it by a mobile agent. To reduce the size of the MA, the sensors of the network have been grouped into sectors. For each MA, we have established an optimal itinerary, consuming a minimum amount of energy with data aggregation efficiency in a minimum time. Successive simulations in large-scale wireless sensor networks through the SINALGO (published under a BSD license) simulator show the performance of the proposed method, in terms of energy consumption and package delivery rate.

Keywords: wireless sensor network; agent cooperation; mobile agent; itinerary; multi-agent system; source node; communication

1. Introduction

Advances in wireless technologies and technologies in micro manufacturing and microprocessor integration have created a new generation of wireless sensor networks (WSN) for a wide range of applications.

A wireless sensor network consists of a set of sensors capable of collecting information from a monitored environment and transmitting it to a base station (sink) via the wireless medium. WSNs are often characterized by dense, large-scale deployment in resource-constrained environments. The limits imposed are reduced processing capacity, storage and especially energy because they are usually powered by batteries. The constraint of the size of a source node (SN) forces designers to limit the size of its battery and, therefore, the amount of energy available. Replacing a battery is rarely possible, either for cost reasons or for environmental constraints. This causes a problem related to energy consumption during operation of the various sensors of the network.

WSNs have given rise to many research issues in order to improve network performance, including maximizing its lifespan. As a result, unlike traditional networks that are concerned with ensuring a good quality of service, research in WSNs must give primary importance to energy conservation. It is, therefore, widely recognized that energy limitation is a key issue in the design of the WSN because of the strict constraints it imposes on the operation of the network. Generally, experimental measurements have shown that data transmission is very costly in terms of energy consumption, while data processing consumes much less [1]. The cost of the information bit transmission in terms of energy is approximately the same as that required for processing a thousand operations in a typical SN [2]. The lifespan of a sensor network can be extended by the application of different techniques of energy consumption. One of these techniques consists in partitioning the network into sectors, as in our previous work [3].

For the past ten years, multi-agent systems (MAS) have grown rapidly and are applied to a wide variety of fields, such as the field of simulation and artificial life, robotics, and image processing. It is integrated in the WSN because of their intelligence and adaptation to different fields. Ant colonies, spiders, etc., are examples of MAS, which are applied to the WSN to process captured data, routing, detection of shorter itinerary, etc.

The integration of mobile agent (MA) into the WSN solves several problems that can harm the wireless sensor network. Indeed, it can be used to reduce the cost of communication significantly, in particular the elimination of redundancies, by moving the processed data by an agent introduced into each sensor instead of bringing them to the raw state at the sink by the sensor itself.

In this work, we will construct a system capable of simulating a network of wireless sensors in an environment, in order to minimize energy dissipation. The multi-agent system principle of operation is used for the intelligent processing of the data captured by the sensors, in terms of reduction of redundancy, evaluation of their importance, elimination of non-useful information, and cooperation between the neighboring SN. The main purpose of this technique is to group the data by a mobile agent and send them simultaneously to minimize transmission, instead of each sensor sending data through intermediate sensors. It also incorporates the principle of multi-agent system operation for the intelligent processing of the data captured by the sensors, in terms of the reduction of redundancy, the evaluation of their importance, and the elimination of non-useful information. A mobile agent activated by the sink has been proposed and sent to the sector. This agent will circulate in the sector nodes in a specified itinerary schema, to gather the captured and processed data by the agents (nodes).

In other words, an agent has been integrated in each sensor node, in such a way that each node of the network is seen as an autonomous agent having its own characteristics and behaviors towards the various events they receive. A mobile agent is used to reduce the cost of communication dramatically, in particular, the elimination of redundancies between source nodes, by moving information processed by an agent introduced into each sensor instead of bringing them to the sink in the raw state. In addition, the SN cooperates with these neighbors to ensure the movement of the MA. Therefore, these agents (introduced in the nodes) eliminate the redundant information contained in the MA and the sensor by itself. Iteratively, the network was partitioned into sectors [3] where the SNs in each sector are included in an itinerary. This technique makes it possible to find an optimal itinerary for an MA to collect data from several distributed sensor nodes. The SN within each sector can be obtained by choosing the angle in an adaptive itinerary. The main purpose of this technique is to group data by a mobile agent and send them at a time to minimize transmission, instead of each sensor sending their data. To collect data in each area, we have created an itinerary algorithm that allows the mobile agent to:

- Reduce the response time;
- Travel the route from the furthest sensor to the sink;
- Start aggregating data from the SN as much as possible until they reach the sink;
- Consume less energy on the route back to the base station;
- Avoid exhausting energy when visiting the SN;
- Decrease the chances of being lost due to energy shortage from the SN; and
- Use neighboring sensors to help SNs that have less energy.

The rest of the article is organized as follows: Section 2 provides an overview of the literature in which algorithms for a number of studies have been performed regarding routing in the WSN. In the first place, we describe the classification of proposed routing solutions, and then the solutions proposed in the integration of mobile agents in the directed diffusion protocol, before presenting the statement of the problem. In Section 3, we present the description of our communication strategy. The packet structure of each mobile agent for itinerary planning and cooperation in information gathering are described in Sections 4 and 5, respectively. Next, Section 6 sets out the purpose of our application, which is to establish a system to simulate the communication between a set of sensors and a base station. Next, Section 7 is devoted to the evaluation of our proposal through simulation, while Section 8 analyzes the results of our approaches compared to other approaches, according to several criteria. Finally, Section 9 summarizes and concludes this article.

2. Related Work

2.1. Classification of Routing Protocols

Large amounts of data need to be processed because of the deployment of a large amount of smaller and less expensive sensors in a sensor array. Therefore, routing (which is the routing of data from a sender to a destination) is a very important service in these networks. The goal of routing is to establish efficient paths between sensors so that messages can be delivered when needed.

Routing is the method of transporting data to the right destination through a network. The goal of routing is to find the least expensive investment in nominal and standby capacity that routes traffic and ensures its survival before any link or sensor failure.

The most energy-efficient protocols rely on routing and consider very specific characteristics of this type of network. Limited energy resources and computing energy of the sensors requires a new generation of protocols different from those of wired networks [3]. This difference can be explained, on the one hand, by the impossibility of having a global addressing with the routing tables as in the IP networks, given the large number and the density of deployment of the sensors. On the other hand, communication is more important in one direction: several sensors send their data to the sink. In addition, the data transmitted are often highly correlated, which requires a specific operation to optimize bandwidth and energy consumption. Finally, sensors are limited in energy, computing power and storage capacity; this requires efficient resource management.

To overcome these constraints, several routing solutions have been proposed, and can be grouped according to several criteria [3–6], namely the topology of the network, the operations supported [5,6] and the destination of transmitted packets [6]. The adopted network typology divides routing solutions into three main forms: flat topology protocols, hierarchical protocols, and geographic protocols. Moreover, if the supported operations are maintained, we can classify the solutions into multipath-based, query-based, negotiation-based, solutions based on the qualities of the service, and solutions based on the destination of the transmitted packets [3]. The last classification criterion distinguishes the solutions proposed as [5]: the uncast communication, or the destination of the message, is a single sensor, the multicast communication, or the recipients of the message are a defined subset of the network, and broadcast the communication, where the message is intended for the whole network [7].

Among the classifications mentioned above, the first is the most found in the fourth-class enriched literature that deals with quality of service based protocols. Flat topology protocols, which can be query-based, use a naming system to specify the required data that restricts redundant data submissions. The global network is restructured by hierarchical protocols into groups called clusters or sectors. Clusters are each represented by a cluster-head that aggregates and reduces the size of the transmitted data to save energy. Geographic protocols use the geographical position of sensors to guide data transmissions to desired regions of the entire network. The last category, based on the quality of service includes some communication skills in addition to data routing.

2.2. Architecture of Wireless Sensor Networks

As mentioned earlier, the main problem in wireless sensor networks is energy. Indeed, as the energy of the SNs is not rechargeable, they must remain operational as long as possible, and it is, therefore, necessary to save the maximum energy consumed by these nodes. One of the characteristics of sensor networks is the ability to reduce the amount of data flowing through the network to conserve energy by merging data through particular sensors in the network. This is called data aggregation. Aggregation does not only require the transmission of data, but also imposes constraints on the network architecture. The idea is to merge data from multiple SNs by eliminating redundancies and reducing the number of transmissions as much as possible to save energy consumed. In what follows, we present a set of previous works conducted within the same framework.

The traditional client/server (C/S) model is one of the most energy-efficient proposals for managing multi-sensor data fusion in the WSN. Several studies [4,8–11] have been made to optimize the architecture of this model. In this architecture, when information from the environment is detected by a sensor, the node sends it as raw data through the other sensors to perform processing. Unnecessary or redundant information is not eliminated by the transmission of raw information, which requires costly construction in terms of energy.

Some works have proposed compression/decompression (C/D) algorithms for reducing the capacity of information transmitted by sensors. To compress the data in a WSN, the authors of [5,6] proposed a data correlation algorithm. In this search, only one sensor is chosen to send raw data to the receiver while the other nodes send only encoded data. After reception, the data is decoded through the correlations between the compressed and uncompressed data. That is to say, it is difficult to find an uncomplicated coding algorithm appropriate for a SN where a minimum of processing energy is consumed.

Authors in [3,5,7] proposed a data fusion (DF) of a maximum number of sensors. When data is sent by one node to the receiver, the intermediate sensors merge their data with that of the first sensor. Thus, this information is merged into one message instead of several, which saves energy. However, intermediate sensors do not always contain useful information to send, and do not eliminate redundant or irrelevant information. In addition, the authors have not studied the importance of the scalability of this type of network.

Moreover, it has been proved in [2–4,9] that the cluster is a fundamental technique in the WSN. Its purpose is to reduce the processing of the aggregation of the necessary data at the sensor node and to move the load on the sink. The clustering algorithm uses a hierarchical classification and sensors are organized into groups (clusters), which elect a cluster node as a leader. The latter collects the aggregated information from the sensors of its group and transmits them to the sink. In this route, energy consumption is reduced significantly. Unfortunately, the authors do not consider the complexity and the amount of energy required to make such grouped sensor arrays.

In addition to [8,9], the authors of [12–15] also proposed a structured tree and cluster strategy based on data aggregation for the WSN. However, according to [16,17], structured approaches are not really practical for dynamic scenarios because of the excessive communication costs and the centralization of WSN structure management.

An ant colony algorithm for data aggregation in the WSN has been proposed by the authors of [8]. All possible routes from the SN to the receiver will be explored by each ant. The accumulated pheromone builds the data aggregation tree. However, one must take into consideration the great dependence of the construction of the appropriate tree of a large part of the deployment of the nodes, which is generally random and consumes a large amount of energy. However, the Euclidean distance calculated from a base station may not be applicable in the WSN because the communication range of a node is limited.

In [11,18–22], which relates to route planning research, authors rely solely on geographic information. As a result, a variation in the size of the data is provided because of the irregular distribution of SN groups. However, this could lead to an imbalance of data between the sensors.

The balancing factor of the data is compromised between the energy cost and the duration of the task. As an increasing amount of data is collected by the sensor in its memory, the information size increases, which, therefore, takes more time and consumes more energy to return to the receiver. Thus, the balance between the mobile agents at the information size level optimizes the duration of the data collection task on the network and decreases the percentage of data loss.

The main idea of authors' proposal in [23] is the inclusion of the sensor, which minimizes the total cost. Indeed, it distributes in parallel a number of mobile agents that sequentially visit the sensor nodes grouped into tree structures. The purpose of this proposal is to reduce the overall energy expenditure and response time, but its shortcoming is the increase in the size of the WSN, where more branches will be created, which would significantly degrade performance due to interference. Tree-based design directions are illustrated in [24] in which the authors propose a heuristic algorithm that improves the one proposed in [21] and determines the appropriate number of mobile agents to minimize the total cost of aggregation, as well as creates inexpensive routes for all agents and selects the nearest node to form the binary tree. It generates small routes for the MA, but energy consumption is doubled in reverse routes and interference between very large numbers of branches.

Researchers [25–30] realized the artificial intelligence community with SNs using the multi-agent system for adaptation to distribute and complex wireless sensor networks. In our previous work [3,4,18,31], we proposed a communication approach using MAS in a WSN plan. Thus, with this solution, we have solved the problem of density by partitioning the network into clusters to some extent. We still have, in [25], proposed a scheme to aggregate a maximum of information. In this work, we have integrated in each SN an agent, in order to process the information and enable each sensor to cooperate with its neighbors.

The authors of [11–17,31–34] proposed mobile agent technology in the WSN for data collection. In these proposals, the management agent is defined as a message containing an application code, a SN list predefined by the receiver and an empty field to place the collected data. It is able to move between all the sensors of the network by moving the captured data in a single message instead of having each sensor bringing this raw information to the receiver separately. The use of a MA allows for more efficient data aggregation on C/S. In these proposals, the management agent visits the SN of the network and progressively merges the sensory data before returning to the receiver. The disadvantage of this type of solution is the difficulty of creating the SN list. Another limitation is the definition of the regions to be treated by the AM, but, as a solution to optimize the route of the MA, the authors [16] have proposed two algorithms of heuristic routes. The Local Closest First algorithm (LCF), the AM starts its route from one sensor and searches for the next destination with the shortest distance to its location. In the Global Closest First algorithm (GCF), the mobile agent begins its route from a sensor and selects the destination closest to the center of the surveillance zone.

The use of a single mobile agent can lead to a very inefficient design due to limited bandwidth and the density of the wireless sensor network. In addition, the use of the latter, which successively visits all the SNs of a large-scale network, may have the following drawbacks:

- Conduction at a very long response time, with many source nodes to visit;
- SN that is found in the MA route extracts energy more quickly compared to other sensors;
- When visiting source nodes, the size of MA increases continuously;
- More power consumption by the mobile agent on his return route to the receiver; and
- The increasing amount of information accumulated by the MA during its migration increases its risk of loss due to noise in the wireless medium.

The main idea of Chen et al. in [19] is to consider multi-agent itinerary planning (MIP) as an iterative version of single agent itinerary planning (SIP). Another algorithm, genetic algorithm based multi-agents itinerary planning (GA-MIP) is proposed in [20]. GA-MIP first proposes a new two-level coding method to solve the MIP problem. The GA-MIP algorithm considers the MIP problem as a single problem instead of using several steps adopted by the previous algorithms. The proposal of [21] considers MIP problem models as a totally-connected graph (TCG). In the TCG, the vertices are the SN of the network and the weight of an edge is derived from the estimates of the jump between the two edge nodes. The authors indicate that all SNs in a particular sub-tree should be considered as a group. In addition, they have a balancing factor, while calculating the weights in the TCG, to form a minimum (BST).

Mpitziopoulos et al. [22] proposed a near-optimal itinerary design (NOID) algorithm to solve the problem of calculating an almost optimal route for an MA. The NOID algorithm was designed based on the rapid adaptation to network modification conditions, the MA route should only include SNs with sufficient energy availability, and the number of AMs depends on the number and size of the network, and the physical location of the SN to visit.

In [23], the main idea of SNOID proposed by Gavalas et al. consists in dividing the area around the base station into concentric areas and start building MA roads with the direction of the interior near Sink. The entirety of SNs in Zone 1 is connected directly to the base station, and these sensors are the starting points of the MA route.

The authors in [35] proposed route planning for PAs. Then, any SSIP algorithm can plan the route of each of these agents. The authors in [36] proposed another data collection system in WSN networks with mobile agents and mobile sink for agricultural applications, in order to plan the route; a dynamic route planning approach is used. An energy-efficient route planning for an MMAEMIP algorithm has been proposed in [37] for iteratively partitioning an area to the directional area where the SN are included in the route; the length of the route is controlled by the sector angle.

In [38], authors developed a mechanism for planning mobile sensors in time-sensitive WSN applications. To plan these sensors, these researchers applied techniques derived from ant colony optimization and genetic algorithms. The static sensor network is derived in small areas where a mobile sensor is assigned to each region. The MA paradigm has proven to be efficient in terms of energy consumption and run time, but the most difficult constraint with this paradigm is route planning for MA. Many solutions have been proposed to solve this problem, but the best multi-agent route algorithms proposed take a long time, or are too complicated in the network.

After analyzing the solutions presented above, one can also deduce that there is still a great deal of work in terms of energy efficiency by paying attention to the relations hip between delivery of the packets and the density of the network. In addition, the solution has to be independent of network deployment.

3. The Proposed Strategy

Our strategy ensures better gathering and management of information in a network; it takes into consideration the energy of each sensor node to make better network presentation. Due to the fact that a great deal of the detected information could be redundant or not useful, a minimization of information gathering could be an effective method to save the energy of the sensor node and extend the lifetime of the network.

About this work, we propose a technique of smart communication that collects information and processes intelligently. Accordingly [1]: transmitting a quantity of data in a single message is less energy-costly than transmitting the same quantity in several short messages. Therefore, because of the density of the network, and high-energy consumption at the data transmission level, instead of having each node send its data to the sink, the network has been broken down into sectors.

Our work is organized as follows: We insert in each node a program to build an agent in the network, which processes the information at the local level and judges their usefulness in order to neglect any unnecessary or redundant information. Moreover, the SN cooperates with neighboring nodes to transmit data.

In the beginning, we grouped the network into sectors, in addition, we proposed a mobile agent at the level of each sector, to concatenate data processed by the SN, to send them in a single message by means of a well-defined strategy.

Once the base station has designated the SN, it sends each sector a MA, to aggregate the data, with a list of well-defined SN. The list of SN is organized according to a route algorithm; the MA circulates between these nodes. For a neighboring sensor to cooperate with its SN, it must respond to two behaviors. The first is that a sensor cooperates with other sensors in the network as much as possible in order to save the transmission of information. Indeed, the reduction of cooperation may result in the loss of certain appropriate information. The sensor consumes energy in its cooperation, but it reduces the probability of losing important information. The second behavior is that a node agrees to cooperate with the network sensors that are in the itinerary to the sink. Indeed, the sensor consumes some energy in its cooperation, but it guarantees the transmission of the packets. During the itinerary of a mobile agent, if the energy of the next SN is not sufficient to transmit the mobile agent, this sensor transmits this information to an intermediate node between itself and the next SN in the list.

Our sector approach based on a multi-agent system is divided into three phases: the first phase is to fix the SN for satisfying the interest according to the direct diffusion, the second is to divide the network into sectors, and the last phase is to determine the mobile agent route in each sector to collect information.

3.1. Detecting Source Nodes

After the base station receives a new task requested by an application, first, the base station broadcasts interest packets to detect the SNs that perform the task, according to the direct diffusion routing protocol. As a result, it generates a message (interest), and then sends it to all its neighboring nodes in a single hop. These neighbors send this message to their neighbors, and so on. The purpose of this message is to discover all the SNs that perform the task, as well as to know and identify all the sensors of our sensor network.

3.2. Partitioning the Network into Sectors

To divide the network into sectors, we used the visiting center local (VCL) algorithm, and we considered the base station as a starting point. A mobile agent has been proposed for each sector to collect information from the SN along a route having a shape similar to that of the sector.

After the receiver receives signals from the SN, it sends MA, to circulate between the sensors of its sectors. These MA group the data processed and captured by the SN to return to the receiver with the data collected.

- *Linking the sink to the VCL sensor*

For each sector, the selected VCL was linked to the base station. As a consequence, several cases can occur: $X_{Sink}>X_{VCL}$ and $Y_{Sink}>Y_{VCL}$, $X_{Sink}<X_{VCL}$ and $Y_{Sink}>Y_{VCL}$, $X_{Sink}>X_{VCL}$ and $Y_{Sink}<Y_{VCL}$, $X_{Sink}<X_{VCL}$ and $Y_{Sink}<Y_{VCL}$; for each case we distinguish between several sub-cases.

What follows is as an example the case $X_{Sink} > X_{VCL}$ and $Y_{Sink} > Y_{VCL}$, then the method used is the following in the Figure 1:

✔ To obtain angle α:

$$\mathrm{Tg}(\alpha) = \frac{(Ysink - Yvcl)}{(Xsink - Xvcl)} \quad \text{then :} \quad \alpha = \mathrm{atg}\frac{(Ysink - Yvcl)}{(Xsink - Xvcl)}$$

✔ Obtain the coordinates of the first point of the perpendicular line:

$$\begin{cases} Xvcl - Xdp = radius \times sin\alpha \\ Yvcl - Ydp = radius \times cos\alpha \end{cases} \textbf{Thus}: \begin{cases} Xdp = Xvcl - radius \times sin\alpha \\ Ydp = Yvcl - radius \times sin\alpha \end{cases}$$

✔ Obtain the coordinates of the second point of the perpendicular line:

$$\begin{cases} Xdp' - Xvcl = radius \times sin\alpha \\ Ydp' - Yvcl = radius \times cos\alpha \end{cases} \textbf{Thus}: \begin{cases} Xdp' = Xvcl + radius \times sin\alpha \\ Ydp' = Yvcl + radius \times cos\alpha \end{cases}$$

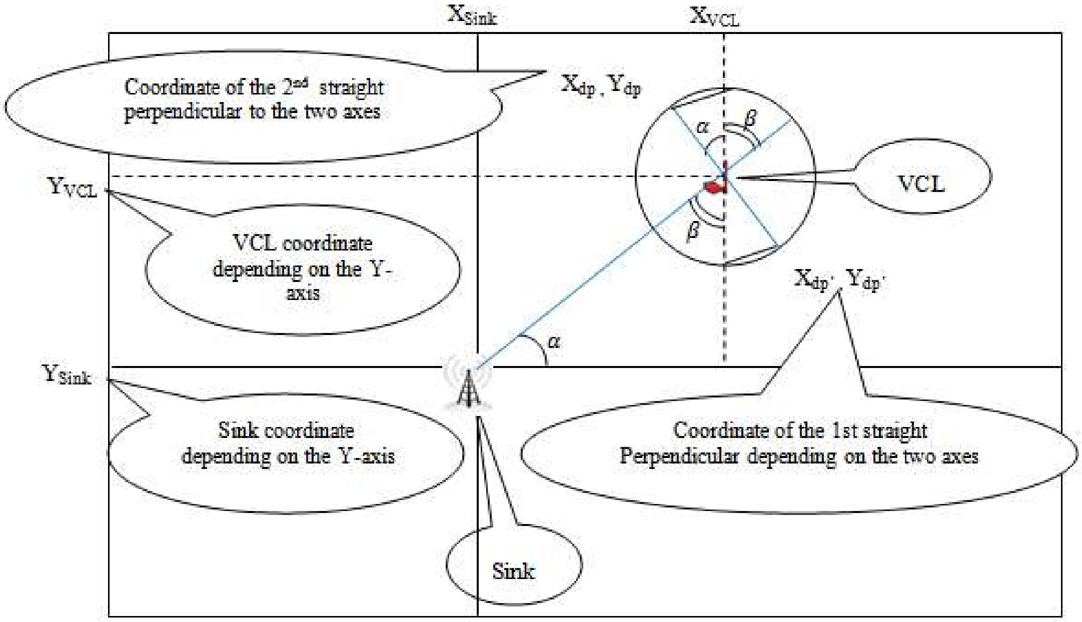

X_{Sink}
X_{VCL}
Coordinate of the 2nd straight perpendicular to the two axes
X_{dp}, Y_{dp}
α
β
Y_{VCL}
VCL
VCL coordinate depending on the Y-axis
β
$X_{dp'}, Y_{dp'}$
α
Y_{Sink}
Sink coordinate depending on the Y-axis
Coordinate of the 1st straight Perpendicular depending on the two axes
Sink

Figure 1. *Linking* the sink with the VCL sensor.

After fixing the coordinates of the two points on the perpendicular line, we can trace it.

- *Connecting an intersection point to the sink*

In each case, to connect a point of intersection of a perpendicular line of the circle whose center is the VCL with the sink, we have several sub-cases. We take an example of the case in Figure 2, $X_{Sink}<X_{VCL}$ and $Y_{Sink}>Y_{VCL}$:

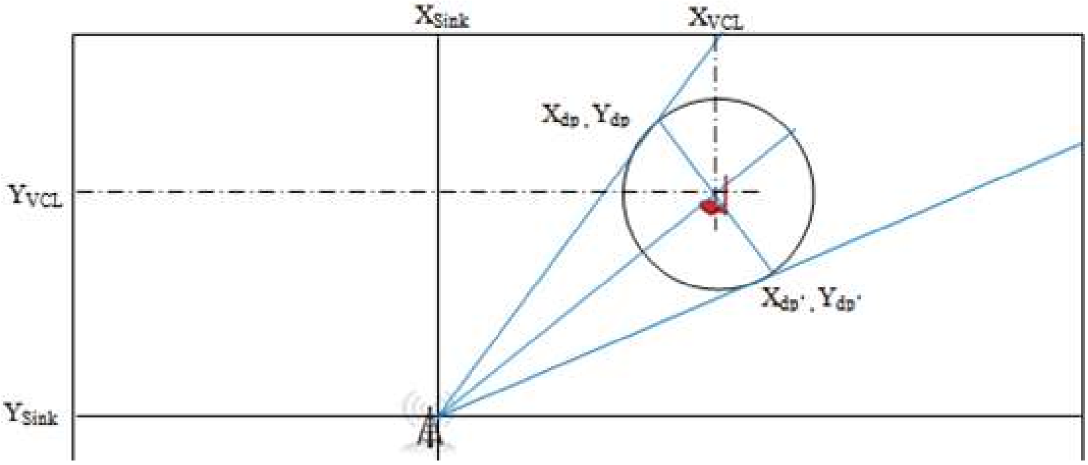


Figure 2. Connecting an intersection point to the sink.

As we know, the equation of a straight line: Y = aX + b then: $\begin{cases} Ysink = aXSink + b \\ Yd = aXd + b \end{cases}$ then:

$a = \frac{YSink - Yvcl}{XSink - Xvcl}$ and $b\ = YSink - aXsink$.

Thus, it is necessary to calculate the coordinates (x, y) of the two points a and b to draw the straight line.

- *Verify the belonging of a sensor in a group*

It is necessary to group the sensor membership to the same sector, after the construction of the sectors. We distinguish different cases and for each case, we distinguish different sub-cases. In the Figure 3, we will try to detail the case of: X_{Sink}>X_{VCL} and Y_{Sink}>Y_{VCL},

✔ First, we calculate the angles α and α' of the group;

✔ Secondly, we calculate the angle for the sensors:α_1, α_2, α_3, α_4,α_5; and

✔ Third, for a sensor to be part of a sector, its angle must be ($\geq\alpha$ and $\leq\alpha'$) | | ($\leq\alpha$ and $\geq\alpha'$).

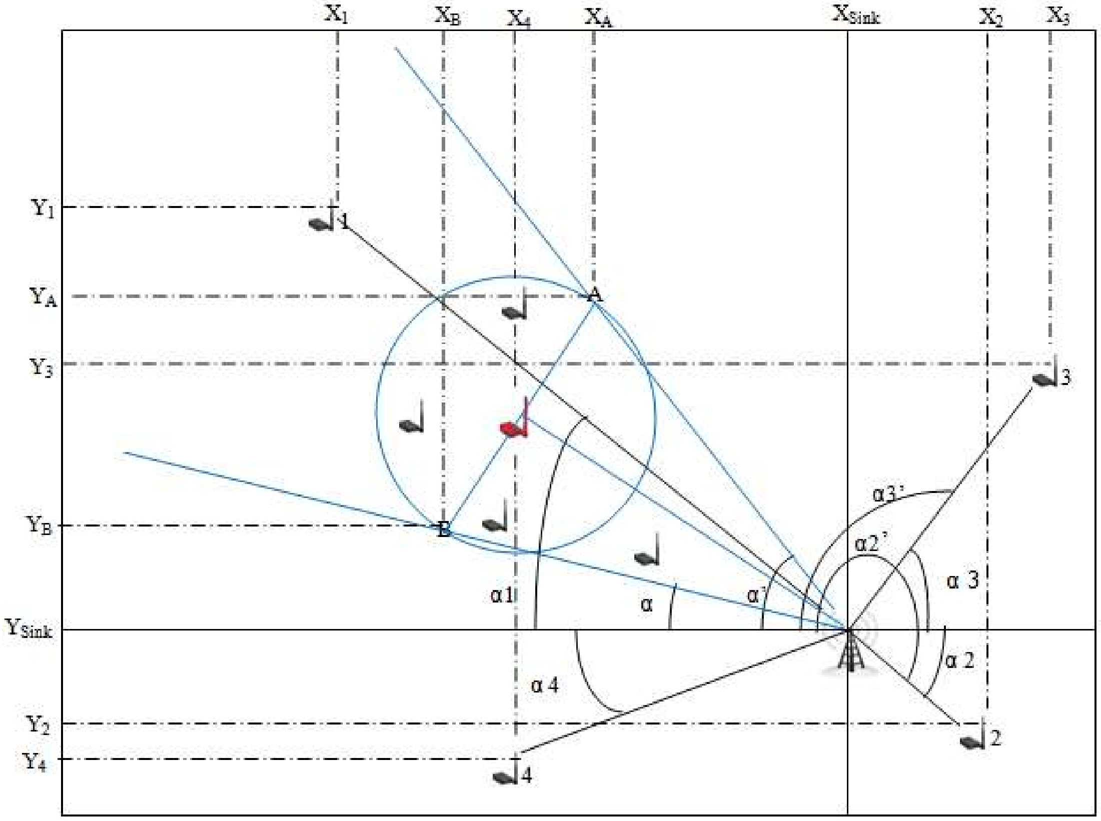


Figure 3. Verify the belonging of a sensor in a group.

How to calculate the angle:

$$\text{Angle } \alpha: \ \text{atg}\ (\frac{|\text{YSink} - \text{YB}|}{|\text{XSink} - \text{XB}|})$$

$$\text{Angle } \alpha': \ \text{atg}(\frac{|\text{YSink} - \text{YA}|}{|\text{XSink} - \text{XA}|})$$

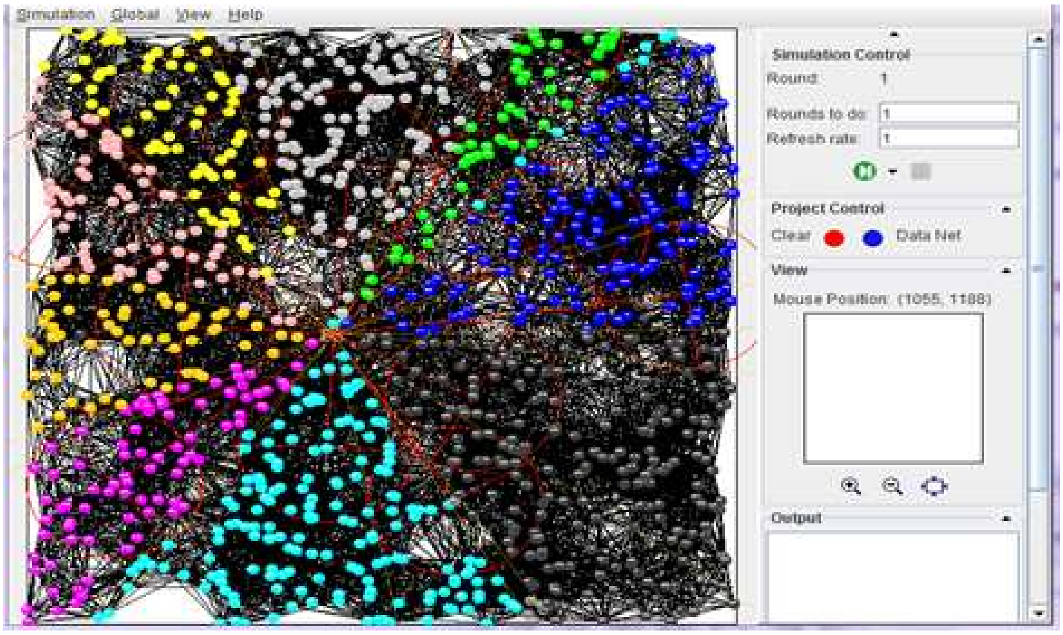
$$\text{Angle } \alpha1: \ \text{atg}\left(\frac{|\text{YSink} - \text{Y1}|}{|\text{XSink} - \text{X1}|}\right). \text{Then}: \ \alpha1 \ = \ \alpha1$$

$$\text{Angle } \alpha2: \ \text{atg}\left(\frac{|\text{YSink} - \text{Y2}|}{|\text{XSink} - \text{X2}|}\right). \text{Then}: \ \alpha2' \ = \ \pi \ + \ \alpha2$$

$$\text{Angle } \alpha 3: \ \operatorname{atg}\left(\frac{|\text{YSink} - \text{Y3}|}{\text{XSink} - \text{X3}}\right). \text{ Then}: \ \alpha 3' = \pi - \alpha 3$$

$$\text{Angle } \alpha 4: \ \operatorname{atg}\left(\frac{|\text{YSink} - \text{Y4}|}{\text{XSink} - \text{X4}}\right). \text{ Then}: \ \alpha 4 = -\alpha 4$$

As shown in Figure 4, the sink partitions the network into sectors.



Simulation Control

Round: 1

Rounds to do: 1

Refresh rate: 1

View

Mouse Position: (1055, 1188)

Figure 4. The sink partitions the network into sectors.

3.3. Plan the MA Itinerary

At this stage, the base station will send the mobile agents, so that they circulate between the sensors of its sectors. The role of these mobile agents is to return to the sink with the information that is collected and processed by the SN. Hence, to determine the route of a mobile agent, we have created an algorithm that allows us to determine an optimal and consistent route.

The algorithm that we have created allows a mobile agent to move from one SN from the sink to another SN closer to it in order to minimize the path of that agent. The base station determines the list of SN that will be visited based on the distance between it and the SN.

4. Mobile Agent

4.1. Mobile Agent Form

It is well known that the direct diffusion protocol detects the SN that contains data, then for each group or sector, the base station sends a mobile agent. These latter are packets of data that can be used to reduce communication costs in sectors in a significant way, especially on low-bandwidth links, by moving the processing function to a sensor rather than bringing the data to the station basic. The data contained in a mobile agent package is illustrated in Figure 5.

Both attributes are used to identify a mobile agent packet(sec_ID and MA_Seq_N). Whenever the sink sends a new MA packet, it increments the MA_Seq_N. The Src_List list specifies the SN of the sector to be visited by the mobile agent. This list will be filled in by the sink. Nex_Src_ID specifically determines the succession of SN identifiers that must be visited by the MA. If Nex_Src_ID is equal to the sink, it means that it is the last node visited by the mobile agent. The Int_List list specifies the intermediate nodes (the help nodes) between the SN. Data_Cooperation contains the information that is processed by the SN of the itinerary (useful and non-redundant information). It is from the second

SN of the itinerary that the MA begins to process the redundancy of information collected at the sensor and so on.

Sec_ID	MA_Seq_N	Nex_Src_ID	Src_List	Int_List
Data_Cooperation				

Figure 5. Mobile agent form.

4.2. Planning the Mobile Agent Itinerary

To establish the MA route, we have created an algorithm that makes it possible to find an optimal and efficient itinerary, in order to manipulate the list of SN in a coherent itinerary. The algorithm we have created provides a MA to pass from a further sensor to a sensor closer to the sink. As we said, the route starts at the first sensor that is furthest away and ends with the nearest sensor to the sink.

The degree of correlation of the detected information among the sensors is closely related to the distance between the nodes, so that the nearby sensors generate unnecessary detected information. The MA aggregates the individual data detected during the visit of each SN. We evaluate the size of the information collected by the MA using the method used by [21]. According to this method, a sequence of detected information can be combined with a merge factor ρ. Let Ni be the amount of accumulated data accumulated after the AM has collected the result Ni (source node i) and Ri the size of the locally processed detected data that will be accumulated by the mobile agent at Ni. Note that data aggregation only starts from the second source, and then we have: $N_1 = R_1$; $N_i = N_1 + \sum_{k=2}^{i} \rho$. R_k(i>=2), as explained in Algorithms 1 and 2.

Algorithm 1. Chaining source nodes at the sink level.

```
Begin
If Sink receives SN identifications by DD then
Begin
// The Sink fills Src_List;
maxim:= List[1]; // List: contains the SN after applying the DD protocol
For i = 2 to N Do
    If dist(Sink, maxim) <dist(Sink, List[i]) Then //the farthest distance with the base station
     maxim:= List[i]
EndFor;
MA@Src_List[1]:= maxim; //maxim: the farthest distance with the Sink
MA@Src _Firs:= maxim;
List:= List – {maxim};
For i = 1 to N Do
minim:= List[i];
For j = 2 To N Do
    If dist(MA@Src_List[i], minim) >dist(MA@Src_List[i], List[j]) Then
    Minim:=L[j]
EndFor
MA@Src_List[i+1]:= minim;
List:= List – {minim};
N:=N−1
EndFor
End;
```

Algorithm 2. Pass the MA among the SNs.

```
Begin
While MA arrives at a sensor$_i$ but not Sink do
If sensor$_i$ has novel information then
            If MA is empty then
            N:=R$_1$; //          Place the information to the MA package
            Nex_Src_ID:= read the novel destination from Src_List;
            Pass the MA to Nex_Src;
Else
               If sensor$_i$ information already exists in the MA Packet then
               // Ignore the information and move on to the next;
               Nex_Src_ID:= read the novel destination from Src_List;
               Pass the MA to Nex_Src;
               Else
                N:=N+ $\sum\rho$. Ri;// Place the information to the MA package
Else
Nex_Src_ID:= read the novel destination from Src_List;
Pass the MA to Nex_Src;
End.
```

5. Gathering Information Cooperation

The cooperation of SNs that do not have enough energy (to transmit the MA) with their neighbors during information collecting sessions are detailed in the figure below. The source node C, which does not have enough energy, begins to seek cooperation near its neighbors of the itinerary when the previous source node B receives a mobile agent as shown in step (1) of Figure 6. The cooperation request is a short message, programmed for a single hop, and neighbors will be programmed not to reply to this message. Thus, node C waits during a fixed delay, if neighboring nodes do not have enough energy; the sensor node considers that these neighboring nodes are not cooperating sensors. In this case, the message is programmed for two hops, and so on, until a cooperating node is obtained (the source node or a simple neighbor node knowing that the priority of choice goes to the source node). Thus, the source node makes the decision to cooperate or not with a neighbor according to a specific energy threshold; which is in the itinerary of the list. After the appropriate decision is made, the source node C sends its processed information (useful and non-redundant) to the cooperating neighbor node C ". Then, the previous source node B sends the MA to the cooperating neighbor node C ", as shown in step (2) of Figure 6.

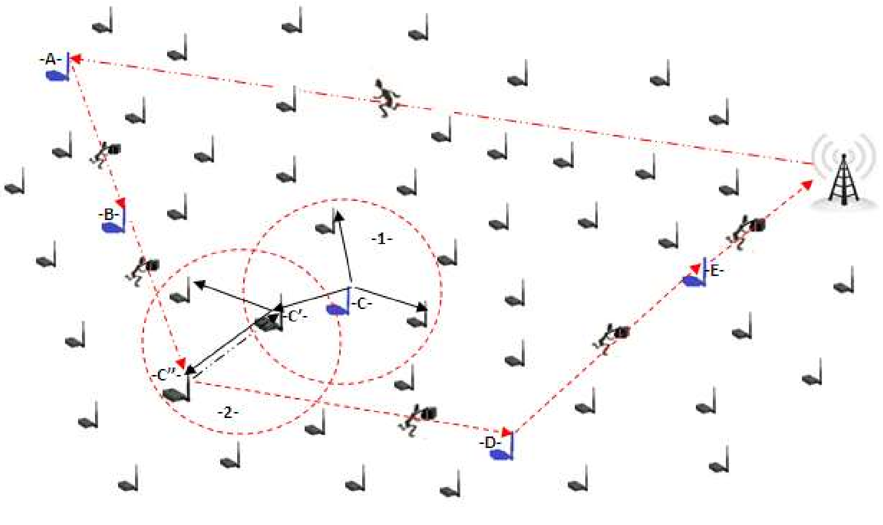


Figure 6. Gathering information cooperation.

The other following source nodes in Src_List list of MA iterate the same previous steps until they reach the base station; these steps are explained in Algorithm 3.

Algorithm 3. Pass the MA among the SNs.

```
Begin
Pass the MA to MA@Firs_Src;
While MA comes at SN Do
If MA@Nex_Src_IDdoes not have enough energy Then
For i = 1 to sensor@Nbr_neighbor Do // neighboring number of SN
sensor diffuse ReqCoop
EndFor
            If deadlines <= D and neighbor sensor cooperating responds Then
            // the waiting time to send the MA to the next
                  If sensor receives ReqCoop Then
                           If E>S Then //if the energy remaining> at a predefined threshold
                           neighbor sensor cooperating responds
                           EndIf
                     EndIf
             N:=N+∑ρ.Ri; //Add information to the package
             Else
             For i = 1 to sensor@Nbr_neighbor Do //neighboring number of sensors neighbor
             Sensor diffuse ReqCoop
                  If neighbor sensor = source node
                   N:=N+ ∑ρ.Ri
                  EndIf
             EndFor
             EndIf
EndIf
Fill in the MA@Int_List //by the intermediate sensor between the two SN
MA@Nex_Sensor = read the novel destination from MA@Int_List;
Pass the MA at MA@Nex_Sensor;
End.
```

The rest of the energy of a battery is represented by E, so when a sensor contains enough energy, the latter participates in a large number of cooperation sessions.

6. Configuration of the Simulation

Thus, we chose to use the SINALGO simulator (Sinalgo is for free, published under a BSD license), for our simulation platform, it is a simulator that enables creating, simulating and validating distributed algorithms [24]. We have set up a platform to simulate or achieve communication architecture in the midst a set of sensor nodes and a sink in the form of a WSN. It is installed via the method of partitioning the network into groups as sectors and the use of a multi-agent system, which is considered as a method to save the energy of a wireless sensor network. We will, therefore, realize an efficient application in terms of energy consumption, the task duration, and the packet drop ratio. In order to design our simulator, we have adopted the following assumptions:

- The energy of a sensor is limited;
- The base station has an infinite supply of energy;
- The sink and SNs are stationary;
- At the beginning, each sensor has an initial energy;
- After network activation, each sensor is an agent;
- The manager considers the number of sensors, the radius and the position of the sink;

- We consider the amount of sensors is limited;
- Sensor positions are known;
- The deployment is random; and
- Two sensors do not occupy the same coordinates.

All the simulations of our work realized a surface of 1000 m × 1000 m with a random diffusion of the sensors for 1000 s. The radio range and the rate of each sensor are limited to 87 m and 1 Mbps, respectively, as proposed in [12]. Transmit and receive power parameters, which directly influence the radio range, were chosen from the platform defined in the sunspot system [17]. The local processing time is set at 40 ms. Table 1 represents the simulation parameters:

Table 1. Basic simulation parameters.

Simulation Parameter	Values
Network size	1000 m × 1000 m
Sensor distribution	Random
Radio range	87 m
Debit	1 Mbps
Size of data collected per sensor	variable at 0.5 Ko to 4 Ko
Data captured interval	10 s
Simulation time	1000 s
Local processing time	40 ms
Treatment Code Size	0.4 Ko
Fusion Factor (ρ)	01

7. Evaluation of the Proposal

To significantly diminish communication costs, mobile agents and multi-agent systems have been used, which has a significant impact on the WSN. Several works have been done, as explained in Section 2, and prove that the use of multi-agent systems in the WSN is more efficient. As in our previous work [31], we have demonstrated that the integration of stationary agents in sensors is more efficient, as well as the partitioning of a network into sectors by providing a client/server network or a clustered network. Still, several works that have been done prove that the use of mobile agents in the WSN is more efficient. Among these works in Section 2, local closest first (LCF) and global closest first (GCF) are the simplest methods for MA itinerary planning. LCF searches for the next sensory information SN that has the shortest distance to the current sensor, although GCF looks for the closest sensor to the sink as its next SN from the base station.

Our article suggest to utilize a smart method based on multi agent systems for the collect of data in a WSN and processed them intelligently. The proposed strategy is called: Intelligent Itinerary based on a Wireless Sensor Network (IIWSN).

8. Results and Analysis

Thus, to prove the performance of our proposal IIWSN in the WSN, we evaluate it to the LCF and GCF methods, based on two criteria. First, we verify the impact of the number of SN on the energy performance criterion; this parameter is considered the most important criterion. Since energy consumption is the parameter that defines the life of the WSN, we set the number of SN from 05 to 50 in steps of 05, and obtain a set of results for each case. We have shown the results thus obtained in Figure 7, which illustrates the impact of the number of SN on the energy, to obtain sensory data of all the source nodes.

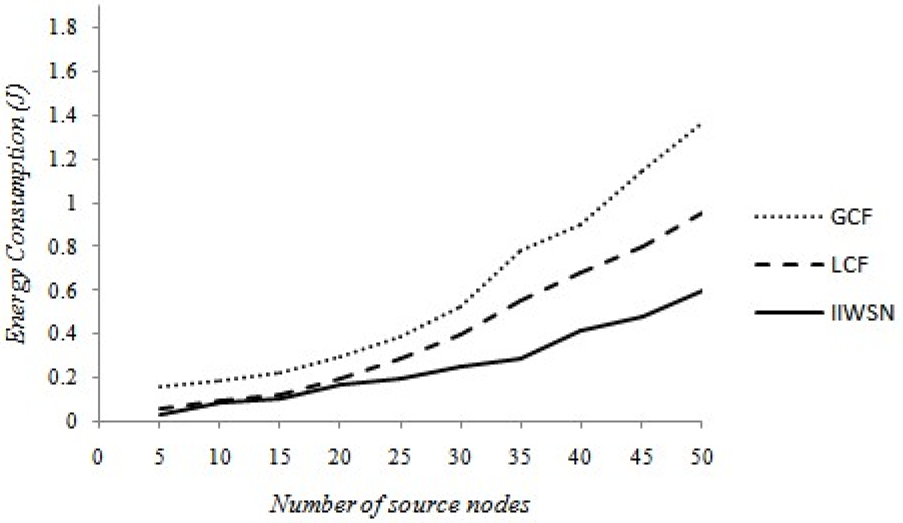


Figure 7. Comparison of energy consumption.

We demonstrate the comparison of the performances of the three approaches in terms of energy in the following figure. Foremost, as we know, when the number of SN is increased, more energy is needed to carry out the data transmission of a communication for the three approaches. We note that our IIWSN approach is always more improved than the GCF approach, not considering the number of SN, and better than the LCF approach even when we increased the number of SNs. We can conclude that the difference between our approach and the other two approaches is becoming more and more important, and this difference increases continuously with the increase in the number of source nodes.

At first, there is not a significant difference between IIWSN and LCF, but there is a small difference between our algorithm and GCF. From 20 SNs, our approach minimizes energy consumption by more than 3% at LCF and 12% at GCF. Nevertheless, at 60 SNs, the consumption of our approach minimizes energy consumption by more than 29.5% and 42% than LCF and GCF, respectively. By comparing the result, the explanation of IIWSN technique presents better energy efficiency.

Moreover, and in an additional test, we illustrate the comparison of the performances of the three approaches in terms of the duration of a task. For local closest first and global closest first techniques, the duration of the task is equivalent to the average end-to-end reporting delay, from the moment the mobile agent is sent by the sink until this agent returns to the receiver. On the other hand, the duration includes the processing time of the information at the SN to eliminate redundancies. Therefore, since more than a few MA are functioning in parallel, and with several experiments, we calculated the duration of a task of a mobile agent to return to the sink, and the duration of a task is the average time of an agent mobile where his itinerary starts from the sink until it returns to it. However, the duration includes the information processing time at the sensors, added to the co-operating time of the SN that do not have enough energy to transmit the MA with its neighbor sensors, and the longer the time taken by the mobile agent. The results obtained are shown in Figure 8:

Figure 8 explains the difference of the performance of the three approaches in terms of task duration. First of all, and this is quite normal, when the number of source nodes is increased, added to the time required for mobile agents to perform the tasks of each of the three approaches. By observing the results, we can see with added SNs to visit, the size of a mobile agent grows and a large number of transmissions will be completed. At first, almost both LCF and GCF techniques are similar, other than from 10 SNs, GCF consumes additional time than LCF. Compared to energy performance, regardless of the number of SNs, we prove that our technique is always lower than the LCF and GCF techniques in terms of task duration. It may be further remarked that the difference between our proposition and the other two approaches becomes more and more important, and this difference increases continuously with the increase of the number of source nodes. In addition, we notice that maximum value between different values is less than 10 ms. As a result the accuracy can be influenced for very time-sensitive applications requiring less than 10 ms. Even more, having more SNs to visit can explain the difference between these methods, as many route verifications will be done to transmit to the MA.

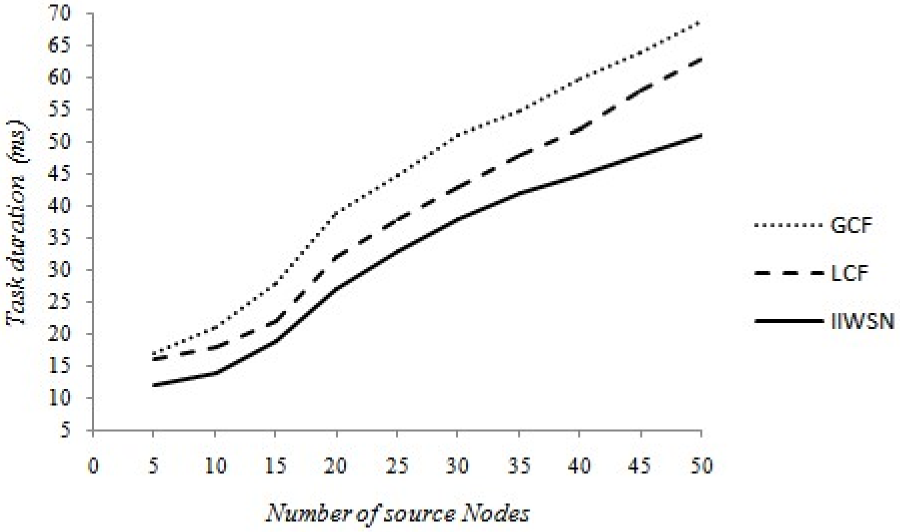


Figure 8. Comparison of task duration.

On the other hand, and in a different essay, we have changed the size of the data captured at each SN as of 0.5 Ko to 0.4 Ko. The results found are shown in Figure 9:

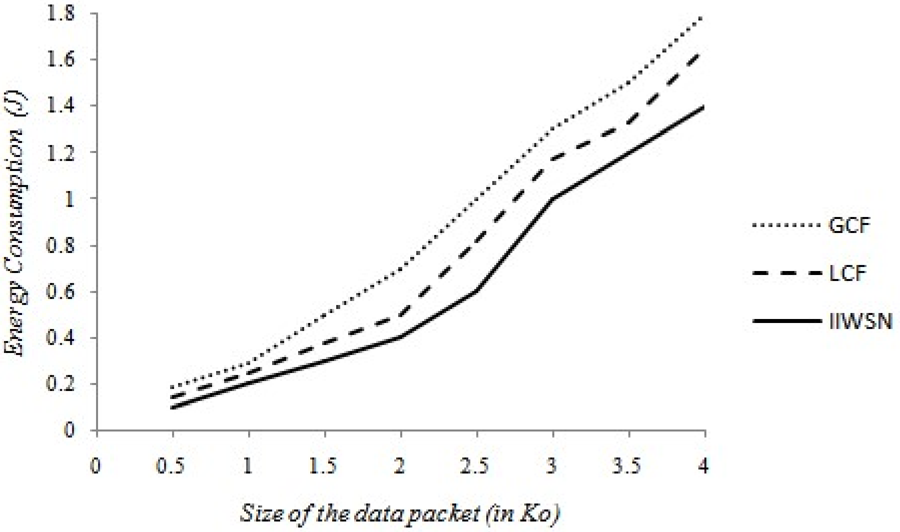


Figure 9. Energy consumption based on network size.

The analysis in the previous figure highlights diver's favorable case: firstly, with the increase of the size of the packets, it is normal that the energy consumption increases continually to execute the tasks of every one of the three techniques. We examine that energy consumption with our approach is also always lower than that of the other approaches, whatever the size of the packets. On the other hand, when packet size is a bit, energy consumption using our approach is less important compared to other approaches. On the other hand, for medium size packets 2 Ko, our approach is lower compared to the other approaches by 15% and 22% than that of LCF, and GCF, respectively. We can still notice, from 2 Ko, the difference between our approach and the other two other approaches becomes more and more important, and this difference increases continuously with the increase in the size of the packets. However, for large packets, our approach is advantageous, at 4 Ko, IIWSN minimizes energy consumption to more than 23% and 29% compared to that of LCF and GCF, respectively. By comparison, the solution of our approach presents better energy efficiency.

9. Conclusions

In an environment where the SNs are close to each other, and where there is considerable redundancy in the sensed data, the sensor nodes generate a large amount of transmissions over the wireless canal, causing not only a loss of the wireless band pass-through, but also the consumption of a great deal of battery energy. Instead of traditional mode approaches like C/S and DF, where the sensor nodes send the raw detected data to the sink, we based our approach on a paradigm of MA and agents at each sensor to process the information (useful and non-redundant).

In this work, we have proposed a strategy that collects and processes information. As we demonstrated in our previous work, sending multiple data in multiple short messages is more than sending them in a single message. Thus, owing to the highest network density and energy consumption at the communication level, instead of each sensor sending its data to the sink, the network has been broken down into a sector.

Our method is based on two types of agents, especially fixed agents and mobile agents. Fixed or stationary agents consist of including an agent into each sensor, so each agent processes locally sensed information from its related SN and estimates its importance. The SN has been grouped into sectors, and an MA has been proposed at each group to concatenate the data processed by the SN, to send them in a single message by a well-defined strategy. This agent will circulate in the sector sensors in a precise itinerary, to collect the processed and captured data by the agents (node). Next, each SN cooperates with its neighboring sensors of the same sector to transmit the mobile agent according to a method based on several important parameters to decide the relevance of cooperation.

Our technique limits the communication of non-useful information, and consequently decreases the amount of traffic and energy consumed. This means a gain in the amount of information and the overhead needed to send them. First, we have proved in previous work, by successive simulations that the division of a WSN into sectors presents better performances comparing to the approaches like C/S and DF, where we used the strategy of the agent.

In this work, we have proven through successive simulations that our approach shows better performance compared to LCF and GCF approaches, in terms of power consumption and the packet delivery rate in the dense wireless sensor network. The reason for these results is that in our approach the MA guaranteed the collection of information in its itinerary. Source nodes that do not have enough energy are able to cooperate with their neighbors to transmit the to the mobile agent in a more efficient and safer itinerary. As future work, either we attempt to minimize the loss of packets, by partitioning the network in another way, or by adding other parameters to the cooperation formula of the mobile agent with the sensor nodes.

Author Contributions: The authors contributed equally to this work.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Gregory, J.; Kaiser, W.J. Wireless integrated network sensors. *Commun. ACM* **2000**, *43*, 5158.
2. Sardouk, A.; Boulahia, M.; Gaïti, L. Agent-Cooperation Based Communication Architecture for Wireless Sensor Network. In Proceedings of the IFIP/IEEE Wireless Days/Adhoc and Wireless Sensor Networks, Dubai, United Arab Emirates, 24–27 November 2008.
3. Bendjima, M.; Feham, M. Optimal itinerary planning for mobile multiple agents in WSN. *Int. J. Adv. Comput. Sci. Appl.* **2012**, *3*, 11.
4. Bendjima, M.; Feham, M. Architecture of an MAS-Based Intelligent Communication in a WSN. *Int. J. Distrib. Sens. Netw.* **2015**, *2015*. [CrossRef]

5. Cecilio, J.; Costa, J.; Furtado, P. *Survey on Data Routing in Wireless Sensor Networks*; Springer: Berlin/Heidelberg, Germany, 2010.
6. Aissani, M. Optimisation du Routage dans les Reseaux de Capteurs Pour les Applications Temps-Reel. Thèse de Doctorat, Université des Sciences et de la Technologie Houari Boumediene && Université Paris-Est, Algiers, Algeria, 13 March 2011.
7. Chen, M.; Kwon, T.; Yuan, Y.; Leung, V.C.M. Mobile Agent Based Wireless Sensor Networks. *J. Comput.* **2016**, *1*, 14–21. [CrossRef]
8. Liao, W.H.; Kao, Y.; Fan, C.M. Data aggregation in wireless sensor networks using ant colony algorithm. *J. Netw. Comput. Appl.* **2008**, *31*, 387–401. [CrossRef]
9. Mineno, H.; Mizuno, T. Adaptive data aggregation scheme in clustered wireless sensor networks. *Comput. Commun.* **2015**, *31*, 3579–3585.
10. Lang, T.; Zhao, Q.; Adireddy, S. Sensor networks with mobile agents Multidisciplinary. In Proceedings of the IEEE Military Communications Conference, Boston, MA, USA, 13–16 October 2003.
11. Mihailescu, M.; Nguyen, H.; Webb, M.R. Enhancing wireless communications with software defined networking. In Proceedings of the 2015 IEEE Military Communications and Information Systems Conference (MilCIS), Canberra, ACT, Australia, 10–12 November 2015; pp. 1–6.
12. Sohraby, K.; Minoli, D. *Wireless Sensor Networks, Technology, Protocols, and Applications*; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2007.
13. Qi, H.; Xu, Y.; Wang, X. Mobile-Agent-Based Collaborative Signal and Information Processing in Sensor Networks. *Proc. IEEE* **2003**, *91*, 1172–1183.
14. Wu, Q.; Rao, N.S.V.; Barhen, J. On computing mobile agent itineraries for data fusion in distributed sensor networks. *IEEE Trans. Knowl. Data Eng.* **2004**, *16*, 740–753. [CrossRef]
15. Vigna, G. Mobile Agents: Ten Reasons for Failure. In Proceedings of the IEEE International Conference on Mobile Data Management, Berkeley, CA, USA, 19–22 January 2004; pp. 298–299.
16. Qi, H.; Wang, F. Optimal Itinerary Analysis for Mobile Agents in Ad Hoc Wireless Sensor Networks. *Proc. IEEE* **2001**, *18*, 147–153.
17. Zhang, Y.; Zhou, Z.; Zhao, D.; Barhamgi, M.; Rahman, T. Graph-based mechanism for scheduling mobile sensors in time-sensitive WSN applications. *IEEE Access* **2017**, *5*, 1559–1569. [CrossRef]
18. Bendjima, M.; Feham, M. Intelligent Wireless Sensor Network Management Based on a Multi-Agent System. *Int. J. Comput. Sci. Telecommun.* **2012**, *3*, 9–14.
19. Chen, M.; Gonzlez, S.; Zhang, Y.; Leung, V.C. Multi-agent itinerary planning for sensor networks. In Proceedings of the International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness, Las Palmas, Spain, 23–25 November 2009.
20. Liu, J.; Cheung, P.; Guibas, L.; Zhao, F. A Dual-Space Approach to Tracking and Sensor Management in Wireless Sensor Networks. In Proceedings of the 1st Workshop on Sensor Networks and Applications (WSNA'02), Atlanta, GA, USA, 28 September 2002; pp. 131–139.
21. Gavalas, D.; Mpitziopoulos, A.; Pantziou, G.; Konstantopoulos, C. An approach for near-optimal distributed data fusion in wireless sensor networks. *Springer Wirel. Netw.* **2009**, *16*, 1407–1425. [CrossRef]
22. Mpitziopoulos, A.; Gavalas, D.; Konstantopoulos, C.; Pantziou, G. Deriving efficient mobile agent itinerarys in wireless sensor networks with NOID algorithm. In Proceedings of the 18th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, Athens, Greece, 3–7 September 2007; pp. 1–5.
23. Gavalas, D.; Pantziou, G.; Konstantopoulos, C.; Mamalis, B. New Techniques for Incremental Data Fusion in Distributed Sensor Networks. In Proceedings of the 11th Panhellenic Conference on Informatics (PCI'2007), Patras, Greece, 18–20 May 2007; pp. 599–608.
24. Chen, M.; Gonzalez, S.; Leung, V. Directional source grouping for multi-agent itinerary planning in wireless sensor networks. In Proceedings of the 2010 International Conference on Information and Communication Technology Convergence (ICTC), Jeju, Korea, 17–19 November 2010; pp. 207–212.
25. Bendjima, M.; Feham, M. Multi Mobile Agent Itinerary for Wireless Sensor Networks. *Int. J. Emerg. Trends Technol. Comput. Sci.* **2012**, *1*, 6–11.
26. Eschenauer, L.; Gligor, V.D. A Key-Management Scheme for Distributed Sensor Networks. In Proceedings of the 9th ACM Conference on Computer and Communications Security (CCS'02), Washington, DC, USA, 18–22 November 2002; pp. 41–47.

27. Cai, W.; Chen, M.; Hara, T.; Shu, L. GA-MIP: Genetic algorithm based multiple mobile agents itinerary planning in wireless sensor network. In Proceedings of the 5th Annual ICST Wireless Internet Conference (WICON), Singapore, 1–3 March 2010.
28. Shokri, R.; Poturalski, R.; Ravot, M.; Papadimitratos, G. *A Low-Cost Secure Neighbor Verification Protocol for Wireless Sensor Networks*; Laboratory for Computer Communications and Applications EPFL: Lausanne, Switzerland, 2008.
29. Bivens, A.; Gupta, R.; Mclean, I.; Szymanski, B.; White, J. Scalability and Performance of an Agent-Based Network Management Middleware. *Int. J. Net. Manag.* **2004**, *14*, 131–146. [CrossRef]
30. Schurgers, C.; Tsiatsis, V.; Ganeriwal, S.; Srivastava, M. Topology Management for Sensor Networks: Exploiting Latency and Density. In Proceedings of the 2nd ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc'02), EPFL, Lausanne, Switzerland, 9–11 June 2002; pp. 135–145.
31. Bendjima, M.; Feham, M. Wormhole Attack Detection for Reliable WSN Communication. In Proceedings of the 2nd International Conference on Automation, Control, Engineering and Computer Science (ACECS'15), Sousse, Tunisia, 22–24 March 2015.
32. Chou, J.; Petrovic, D.; Ramchandran, K. Tracking and exploiting correlations in dense sensor networks. *Conf. Signals Syst. Comput.* **2012**, *1*, 39–43.
33. Chou, J.; Petrovic, D.; Ramchandran, K. A distributed and adaptive signal processing approach to reducing energy consumption in sensor networks. In Proceedings of the Joint Conference of the IEEE Computer and Communications Societies, San Francisco, CA, USA, 30 March–3 April 2003; pp. 1054–1062.
34. Patil, S.; Das, S.R.; Nasipuri, A. Serial Data Fusion Using Space-filling Curves in Wireless Sensor Networks. In Proceedings of the 2004 First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks, Santa Clara, CA, USA, 4–7 October 2004; pp. 182–190.
35. Srdjan, C.; Levente, B.; Huraux, J.P. SECTOR: Secure Tracking of Node Encounters in Multi-hop Wireless Networks. In Proceedings of the ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN), Washington, DC, USA, 31 October 2003.
36. Dong, M.; Ota, K.; Yang, L.T.; Chang, S.; Zhu, H.; Zhou, Z. Mobile agent-based energy-aware and user-centric data collection in wireless sensor networks. *Comput. Netw.* **2014**, *74* (Part B), 58–70. [CrossRef]
37. Wang, J.; Zhang, Y.; Cheng, Z.; Zhu, X. Emip: Energy-efficient itinerary planning for multiple mobile agents in wireless sensor network. *Telecommun. Syst.* **2016**, *62*, 93–100. [CrossRef]
38. Heidemann, J.; Silva, F.; Intanagonwiwat, C.; Govindan, R.; Estrin, D.; Ganesan, D. Building Efficient Wireless Sensor Networks with Low-Level Naming. In Proceedings of the 18th ACM Symposium on Operating Systems Principle (SOSP'01), Banff, AB, Canada, 21–24 October 2001; pp. 146–159.