Efficient Multi-sink Relocation in Wireless Sensor Networks

L. Friedmann[1] and L. Boukhatem[1]

[1]Laboratoire LRI
Université de Paris Sud XI
15, Avenue du général Georges Clémenceau
91405 Orsay Cedex – France

***Abstract*— The research community in wireless systems hold a keen interest in sensor networks issues. Sensor networks are dense wireless networks where information is gathered by sensor elements spread out in an interest area. Wireless sensors applications cover a large field such as surveillance and security, target tracking, agriculture, health and military purposes. The main deficiency of sensors is their finite source of energy. Therefore, an efficient utilization of this energy resource conditions the network lifetime. In order to enhance the performance of these networks, some research efforts have focused on the mobility of a single or multiple sink nodes. The mobility of sinks introduces a tradeoff between the need for frequent re-routing to optimize the performance and the minimization of the overhead resulting from this topology management. In this paper, we propose a new dynamic approach to extend the lifetime of a sensor network based on both mobility and multiplicity of sinks. According to the evolution of the network, in terms of energy dissipation and distribution, this approach aims to find the optimal position for all the sinks in order to optimize the lifetime of the network and move accordingly these sinks in an intelligent manner. Simulation results show the efficiency of our approach in terms of energy gain.**

Keywords — **Wireless sensor networks, power efficiency, multiple sinks mobility.**

I. INTRODUCTION

A sensor network is composed of a large number of wireless sensors, densely deployed, in the range of a phenomenon to observe, study and/or monitor. A sensor is an electronic device which generally gathers three main capabilities: the ability to measure and collect data relative to the environment surrounding it, the ability to process these collected data, and the ability to exchange it with other devices. The other devices can be sensor nodes or sinks. A sink is a particular node which collects the information resulting from the sensing nodes, process them and/or send them to a data concentration center. Generally, sensor nodes deliver their collected data to the nearest sink.

The main constraint in sensor networks is their limited energy supply. Therefore any program running on this device has to manage carefully the autonomy issue. Indeed, it has been shown that a wireless communication is one of the more expensive operations the sensor has to perform. Hence, all research efforts in this area have focused on energy-aware solutions so that the lifetime of the network is maximized.

Most proposed routing approaches in sensor networks are centered on energy minimization by looking for multi-hop links. In fact, the largely explored multi-hop approach is based on the observation that the transmission power of a wireless communication is proportional to distance squared, or even higher (in the presence of obstacles). Hence, the multi-hop routing consumes less energy than direct communication. Although proposed routing protocols are able to dynamically adapt according to nodes energy, the nodes nearby the sink serving as last-hop relays observe rapid depletion in their energy supply. Therefore, to improve the network lifetime by reducing the total transmission power, the sink is moved towards the last-hop relays which are the most involved in packet transmitting. Such an approach has also the advantage of reducing the average delay observed by data packets.

On the other hand, in a network where not only a single but multiple sinks are present, the correct placement of the sink nodes directly affects the lifetime of such a network. Some research works dealing with the optimal multi-sink positioning problem have been proposed. However, they have not taken into account the network evolution, they only try to place, in an optimal manner, the sinks once and for all. Once the network is deployed, it remains static.

Our aim in this article is to propose a multiple sinks relocation solution for network lifetime optimization. In order to move the sinks towards their optimal positions in an intelligent manner, we developed an efficient solution based on a constrained local search strategy.

The paper is organized as follows. In section II, we present the previous works related to optimal sink positioning and sink movement in large scale sensor networks. Section III describes our multi-sinks movement approach for network's lifetime optimization. Section IV presents and discusses the obtained simulation results. Finally, we conclude the paper in section V.

II. RELATED WORKS

A. Multi-sink optimal positioning

The optimal sink placement issue in a network is an NP-complete problem [3], [5]. Some works have addressed this problem and tried to resolve it by different approaches in a polynomial time (using approximation methods) or not [3], [4], [5].

In [3], the authors tackled this issue as a flow problem. Several propositions in the literature have focused on the flow problem in a network with a single sink and reached efficient solutions based on energy considerations. The paper [3] extended the study to the case of a multi-sink network. To maximize the network lifetime, the authors consider two questions: how to place the sinks in the network and how to route the data towards these sinks? Both of these issues have been formulated using linear programming and resolved with CPLEX software [3]. The produced linear program allows to find out among n nodes which would be the p sinks to reach the maximization of the network lifetime. However, the authors emphasized that their solution does not resolve the NP-completeness of the problem and leave for future works to take charge of this issue by proposing a polynomial time approximation solution.

In [4], the authors use the clustering principle. They assume that the number of sinks to place is known prior to the exploitation phase, and this number represents the number of clusters in the network. Therefore, the problem is translated to find an efficient clustering algorithm. Many clustering algorithms exist in the literature (*k-means clustering*, *self organizing maps*, etc. [4]). Once the clusters formed, sinks placement becomes simple. The easiest solution is to position the sink in the center of mass of the cluster nodes. However, if the routing protocol uses a multi-hop criteria, then it would be wiser to use power aware distance metrics to take into account the consumed energy to reach the sink.

In [5], the authors have shown that if the transmission power and the capacity of nodes are fixed, the maximization of the network lifetime is comparable to a maximum flow issue. The used metric to estimate the optimality of the network is the maximization of the data production rate while guaranteeing the network subsistence. Even impracticable in several cases, the authors assumed, in this approach, that a central algorithm is available to provide information on nodes localization. Their aim was to develop an analysis tool to be aware of sinks position effect on the flows in the network and thus on its lifetime. This Base Station Positioning (BSP) problem, as called by the authors, remains NP-complete. To find an approximated solution, the authors have investigated three approximation algorithms: a greedy algorithm and two local search algorithms (Random-restart hill-climbing algorithm and Metropolis -Simulated Annealing- algorithm). It obviously appears that the sinks position highly affects the efficiency. One possible solution for sink positioning is to choose the best out of a linear number of random samples. However, the authors results show that in many cases the local search provides better results. In spite of the inability of the local search algorithm for guaranteeing the solution quality, it seems that this algorithm solution is close to the optimum. Disregarding its good solutions, determinism and low complexity, the greedy algorithm was still below the performance of the local search method.

In conclusion, the optimal multi-sink positioning problem in a sensor network has already been addressed in the literature. However, major propositions as [3], [5], which formulated this problem as a maximum flow issue, have restricted the solutions space since the sinks are chosen among the nodes. This constraint, which chooses p sinks among n nodes, is very strong and will be released in our paper.

Besides, all these works focused on the initial positioning of sinks and considered that the network remains static since the optimal position is discovered. None of these works mentioned to optimize the network efficiency throughout its lifetime. None has put forward the idea of combining the optimal placement and the ability of moving sinks in order to manage, in real time, the network all over its lifetime. All these proposals mainly based their solutions on nodes position and not nodes energy consumption. Even if in [4], it appears that the authors tried to find the sink placement within a cluster using an energy-aware algorithm but the clusters pattern is definitely based on nodes position and not their energy supply.

B. Sink movement

Very few research works have focused on the relocation of a unique sink in a sensor network [6], [7], [8], [9], [10].

The authors of [7] propose a potential sink moving in order to reduce power dissipation in the nodes nearby the sink serving as last-hop relays. The following questions are addressed:“When?”, “Where”, and “How” to move the sink without affecting data traffic. Sink repositioning problem is NP hard in nature, and a heuristic approach is used to resolve it. The used solution is based on the substitution of the last-hop nodes serving the sink (substitution operated by the routing algorithm) and the traffic delivered by these nodes. If the distance separating a last-hop node with the sink is longer than a threshold value, then the formula *energy transmission* $\times$ *traffic* is evaluated. If the obtained value is greater than a fixed threshold, the sink estimates the effect of the movement and performs it if justified.

To improve the network lifetime, the sink is moved towards the last-hop relays which are the most involved in packet transmitting. To moderate the relocation, a point G is defined as the equidistant position from these relay nodes in term of *distance* $\times$ *traffic*. In order to avoid affecting the less traffic dominating nodes, the final position of the sink is fixed between its actual position and the point G. This position is determined using a dichotomy approach based on an evaluation formula. Finally, the sink is moved using a straight line movement.

Before achieving the movement, the sink evaluates the gain of such a reposition. To estimate the quality of a position the used metric evaluates the whole energy transmitted by the last-hop relaying nodes. The repositioning is accepted if the energy gain exceeds a fixed threshold. The threshold computation is based on the overhead generated by the sink movement. The obtained simulation results show that the sink repositioning involves a decrease in packet energy consumption, an increase in the average node lifetime, and a reduction in the transmission delay.

In [6], [7], [8], [9], [10], all the proposals focus on the relocation of a single sink in a cluster. The solutions can be extended to a network including several sinks by organizing it in clusters. However, we can observe that relocation of each sink is limited to its cluster and the approaches do not have a global view of the entire network.

Therefore, since this direction was not explored by now, we aimed to propose a solution which manages the movement of several sinks in a more global strategy, without restricting the sinks in clusters and by taking advantage of the works related to the optimal multi-sink positioning.

III. OUR APPROACH: MULTI-SINK MOVEMENT

Multi-sink repositioning consists in finding in real-time, during the network lifetime, the best sinks positions within the network. Giving the sinks a movement facility allows us to move the sinks in an intelligent way towards the positions which optimize the network lifetime. Our approach does not attempt to optimize the lifetime of each cluster by relocating each sink in its cluster but tries to view the network as an entire and sole entity.

The evaluation of a configuration is based on an energy-aware routing algorithm which was developed in [6], [8]. Since the evaluation is based on the routing algorithm, we are confident that there is a real correlation between the consumed power during routing and the evaluation score. Using this mathematical tool for configuration evaluation, we adopt an operational research theory approach to find an approximated solution. Once the optimal sinks positions are determined, the last step is to complete a smart movement of the sinks. Indeed, the straight line movement towards the final position can be very expensive in terms of energy cost.

A. Routing

As in [6], we adopt a centralized approach which allows to use a source routing methodology (see figure 1). Before realizing the route discovery, a first phase consists in providing each link in the network a specific weight. This weight depends on the energy of the destination node, in order to relay the information by the nodes having the higher remaining energy, and the distance between nodes, in order to prefer short distance transmissions.

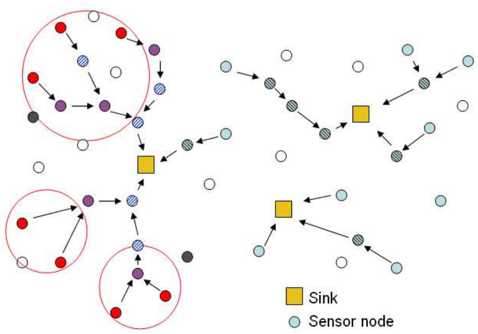

Fig. 1. Routing in a sensor network

Once this weight computed, we obtain a graph on which Dijkstra algorithm can be applied to find the shortest path between a sink and each node in the network. However, in our case the network contains more than one sink. So the aim now is to find the shortest path towards the nearest sink. We will proceed by exploring all the shortest paths towards all the sinks, and we will conserve for routing needs the one leading to the nearest sink. At this point, it is worth to notice that a more efficient searching method can be found, nevertheless our approach has the benefit to easily maintain, for each node, the n nearest sinks and their corresponding paths, which are necessary for the configuration evaluation (see section III-E.2).

The algorithm complexity is acceptable since we achieve p Dijkstra in $O(n)$. The next problem to tackle is how to compute the weight of the links.

B. Links weight

In [6], the cost function of a link connecting two nodes i and j is defined as the sum of eight metrics balanced with their respective coefficient. Despite the relevance of all these metrics, and for simplicity sake, we have chosen the two most important ones:

- The distance between the nodes i and j defines the consumed energy for the communication. Since the required energy is proportional to distance squared, the weight will show the same behavior.
- The remaining energy in the destination node j which advantages the highest energy relaying nodes. The lowest is the destination node energy and the highest is the link weight.

The link weight is defined as the following:

$$W(i,j) = CF_0 \times dist(i,j)^{expCF_0} + CF_1 \times \frac{1}{energy(j)} \quad (1)$$

where,

- $W(i,j)$ is the weight of a link between nodes i and j

- $dist(i,j)$ is the energy consumed by the communication between nodes i and j

- $energy(j)$ is the remaining energy in node j

- CF_0, CF_1, and $expCF_0$ are coefficients for equation balancing.

When a packet arrives to a sink, the energy model is updated by decreasing the energy of the whole nodes which contributed to the packet relaying. In our implementation, only the transmission energy consumption has been considered.

For routing concern, we used the routing protocol defined in [6]. The routing protocol is divided in several periods. The major periods are data transferring phases. Regularly, a routing phase is activated in order to detect if a better configuration of the sinks is possible and take a potential decision for sinks movement.

C. A solution evaluation

The evaluation of a configuration have to reveal its efficiency as regards to the network lifetime. A priori, the best placement will gradually get the sink closer to the nodes that generate the highest traffic but without moving it too far from other future data sources so as to lose connectivity with them. Hence, we define three categories of nodes listed bellow in priority order:

- S_1, set of nodes which are in the *sensing* state and transmitting data,
- S_0, set of nodes which are in the *sensing* state but are not currently transmitting data,
- N, set of all the other nodes.

The equation comes naturally as follows:

$$\forall i \in S_1, \sum TC(i, K(i,1)) \times NP_{nbc}(i)^{x_1} \times C_1$$
$$+\forall i \in S_0, \sum TC(i, K(i,1)) \times C_2 \quad (2)$$
$$+\forall i \in N, \sum TC(i, K(i,1)) \times C_3$$

where

- $K(i, j)$ is the j^{th} nearest sink from node i,

- $TC(i, s)$ is the transmission cost from node i to sink s. It is derived from the routing table,

- $NP_{nbc}(i)$ is the rate of node i. It represents the number of generated packets during the last nbc cycles,

- C_k and x_1 are coefficients for function balance.

Using the appropriate coefficients C_k, this equation first takes into account the sensing nodes which transmit data, on a data ratio basis, then sensing nodes which currently do not transmit data, and finally the rest of network nodes.

Using this mathematical tool, we can now qualify a network configuration. After all, the optimal sinks position is the configuration which minimizes this cost function.

D. Optimum search

The optimal multi-sinks positioning problem in a network is NP-complete. The only possibility to be certain about the optimality of a solution is to perform an exhaustive exploration of the solutions space. However, the exhaustive exploration is not conceivable in large scale problems. In our case of study, we face an infinite space of solutions since a sink position is defined by a couple of real numbers. Even if we admit such an approach in an integer numbers space, in a 100×100 points space, we will have 10^{12} solutions for a 3 sinks placement.

As stated before in section II-A, to overcome this complexity, meta-heuristic approaches are used to derive upper bounds based on a neighboring exploration. In our work, we choose the *local search Random-restarts hill-climbing* algorithm which was previously used in sensor networks context and where it showed good results.

1) A solution neighborhood and relocations: The aim of the local search approach is as follows: from an initial solution x_0, a finite series of solutions x_i is generated with a systematic change of neighborhood. x_{i+1} is derived from x_i such that for all i, $f(x_i) > f(x_{i+1})$. f is the evaluation function of the solution. In other terms, only the solutions which enhance the solution are accepted.

Generally, the neighborhood of a solution is derived by the mean of an elementary transformation. A transformation is every operation which allows to change a solution x into a solution $x^{'}$ of S. S denotes the set of solutions.

We defined three levels of transformations to derive the neighborhood of a solution:

- One sink movement: this transformation consists in one sink relocation in respect to the initial position. This movement is performed towards eight directions: North(N), South(S), East(E), West(W), N-E, N-W, S-E, S-W. In our experimentations, the extension of the movement is fixed to 0.1, 0.3, 1, 3 and 10 which leads to 40 movements for each sink.
- Two sinks movement: this transformation consists in a two sinks simultaneous movement to widen the neighborhood of a solution and avoid certain deadlock situations. We limited the transformations to the fourth cardinal points. We obtain then 16 possible movements for the couple of sinks. Three movement steps were chosen for the experimentations (0.3, 3, and 10) leading to 24 movements per sink.
- Three sinks movement: three sinks are relocated simultaneously.

To illustrate the relevance of the two sinks movement, figure 2 shows a theoretical case proved by simulation where a single sink movement do not lead to a better solution. The movements are rejected because they degrade the solution. However, the simultaneous movement of two sinks allows in this case to get a better solution which will lead to the optimal solution.

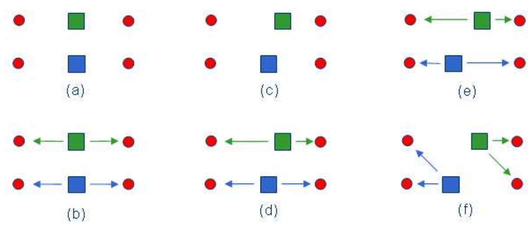

Fig. 2. a,b): Initial network and associated routing; c,d): one sink movement and associated routing; e,f): two sinks movement and associated routing

More complex movements can be defined, but this will require a longer exploration of a larger neighborhood. Hence, a compromise has to be made between the quality of the solution and the devoted time to reach it.

However, the local search approach has its limits in the sense that the obtained optimal solution disregards current sinks positions. Hence, we can obtain for example a solution which inverts the positions of two sinks leading to a wasteful relocation and a calamitous power consumption. To avoid such situations, we implemented a solution based on both permutations generation and a minimization of the total distance traveled by the sinks.

E. Sinks movement

Once the optimal sinks positions found, the relocation problem of each sink to its final position still remains. As shown in figure 3, a linear movement can lead to costly configurations. In the illustrated example, even if the sinks move towards their optimal positions, the movement of the left sink has impaired the situation. It would have been better if the left sink has waited that the right sink gets closer to its optimal position to begin the movement.

In fact, to derive the optimal position the local search approach does not consider the sink velocity, the distance to travel, and the dynamics of the network. To perform an *intelligent* movement, we propose an approach based on a local search in a constrained space.

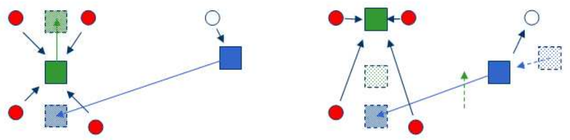

Fig. 3. Initial and optimal positions, routing and linear movement of sinks

1) constrained local search: The aim of the constrained local search is to limit the sinks movement while maintaining their direction to the optimal positions. We define a *liberty space* based on the current and the optimal locations of the sink. The next move will take place in this constrained space.

Let's d be the *liberty distance*. We also define G, the point located at a distance d from the current position of the sink on the line formed by the current position Cp and the optimal position of the sink. The liberty space is then defined as the set of points located at a distance smaller than d from the point G and the current position (see figure 4). Graphically, this point is the intersection between two discs with G and Cp as centers and d as radius.

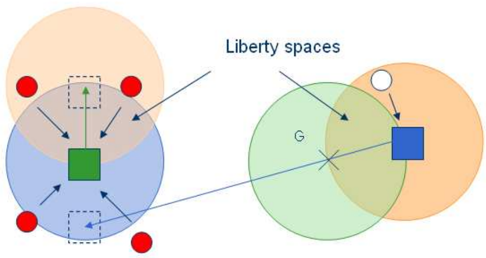

Fig. 4. Liberty spaces delimitation

To find the new target of a sink defined as the *constrained optimum*, a local search is activated in the constrained space using the same principle used to find the optimal position. All the neighbors of the solution which do not belong to this restricted space are disregarded. Therefore, the constrained local search will generate constrained optimums which help to perform efficient and power-aware movements towards the optimum position.

However, as illustrated in figure 5, the restriction made on the sinks movement can result in deadlock configurations while the optimum solution exists. In our example, once the first constrained optimal position is reached by the right sink, the sink is blocked. Using the evaluation function (2) of the local search algorithm, none of the positions in its constrained space can enhance the solution. We come to the conclusion that a new evaluation function has to be defined.

2) New evaluation function: As shown in the previous example, the lack of the evaluation function is that it considers only the transmission cost of each node towards its closer sink. A more efficient evaluation function would consider the transmission cost between each node and each link using digressive coefficients according to sinks proximity order. Even

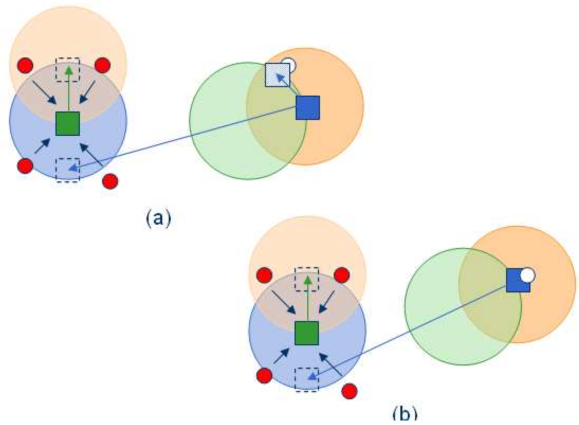

Fig. 5. a)Intelligent movement in the constraint space; b) The sink in the right side is blocked

conceivable, such a function would be substantial to compute. Therefore, we restrict the number of sinks to the three nearest sinks from each node, considering that this solution will allow to get out from most deadlock situations without an excessive calculation load. In the light of these observations, we defined the following new evaluation function:

$$\begin{array}{l} \forall i \in S_1, \sum \begin{array}{l} TC(i,K(i,1)) \times NP_{nbc}(i)^{x_1} \times C_1 + \\ TC(i,K(i,2)) \times C_{1bis} + \\ TC(i,K(i,3)) \times C_{1ter} \end{array} \\ + \\ \forall i \in S_0, \sum TC(i,K(i,1)) \times C_2 \\ + \\ \forall i \in N, \sum TC(i,K(i,1)) \times C_3 \end{array}$$

where C_{1bis} and C_{1ter} are coefficients corresponding respectively to the second and third nearest sink.

As shown in figure 6, this new formula allows to exit from the deadlock situation of the previous example. In general, this new formula allows to derive constrained optimums which tend to slowly approach the non constrained optimums in an intelligent manner.

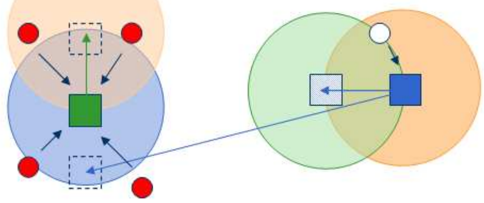

Fig. 6. Intelligent movement in the constraint space using the new evaluation function

Finally, it is worth to notice that the distance separating the current position and the next constrained optimum depends on both sink velocity and routing phases frequency. It is then strongly correlated with the routing protocol. More specifically, at the beginning of the next routing phase, the intermediate position would be reached. Both the basic local search and the constrained search algorithms are activated. If only few modifications have occurred on the network, we will find an optimal position very close to the previous one, to which we will get closer. Conversely, if the network changes, we will find a new position and we will guarantee a good reactivity.

Concerning the data management during the sinks relocation, a movement plan is drawn up to avoid packet loss. Before the sink relocation, the algorithm checks if the last-hop relays can reach the sink during the movement. If not, we implemented in our simulations a solution which performs a complete re-routing. Other solutions are possible [10] such as the increase of the last-hop relays transmission power or the designation of new relaying nodes during the movement.

IV. Simulation results

A. Simulation model

The implementation of the sensor network as well as the solutions we proposed were realized by simulation with Opnet Modeler 10.0 software. A first series of simulations was dedicated to validate the model. The aim was to make sure that the network behaves according to the theoretical model and operates on some easily verifiable configurations.

In the experiments, the sensor network consists of nodes and sinks randomly placed in a 100x100 square meter area. Except for testing specific capabilities of our approach, the sinks and node positions are determined randomly within the area boundaries. Each node is assumed to have an initial energy of 5 joules and is considered non-functional if its energy level reaches 0. We have carried out many experiments with various numbers of sinks and nodes to evaluate the performance of our smart movement. The experiments we present in the next sections are representative of the whole series of tests we achieved.

B. Performances Metrics

We used the following metrics to evaluate the performance of our multi-sink repositioning approach and to compare it with the motionless approach:

- *Time for first node to die*: This metric gives an indication of network lifetime.
- *Number of delivered packets and lost packets*: This metric gives a good measure of the efficiency of each approach (with and without sink movement). A good approach will lost fewer packets.
- *Average delay per packet*: Defined as the average time a packet takes from a sensor node to the gateway.
- *Average energy consumed per packet*: This metric represents the average energy consumed in transmitting and receiving a data packet. An approach that minimizes the energy consumed per packet will, in general, yields better energy savings.

C. Environment validation

We first verified that the local search algorithm finds a better configuration than the best configuration of a hundred random configurations. The results are quite similar to those of the local search algorithm presented in [5]. Then we verified that the implemented routing algorithm was energy-aware, i.e. the routing table changes according to the sensor energy level and is not only based on the distance between sensors. The results of an experiment, shown in Figure 7, illustrate this principle. In this example, when the energy of the node C is equal to 5 Joules, the routing algorithm chooses C to route the information from A to the sink. Whereas, the node B is chosen as relay when the energy of C is 1.5 Joules.

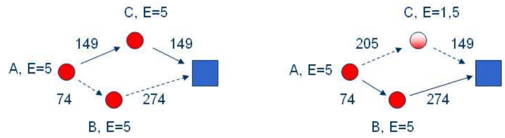

Fig. 7. Energy, routing cost and energy-aware routing

We have also checked that the optimal location is really influenced by the throughput. The experiments prove that the optimal location tends to be closer to the nodes which provide the most packets, as shown in figure 7.

Fig. 8. Optimal location according to the number of produced packets

Finally, we checked that the optimal location of each sink is selected so as to minimize the sum of the distances crossed by the sinks and thus, reduce the delay necessary to reach theses positions. This was done by testing each possible permutation of the sinks.

D. Performance results

In this section, we present the main performance results derived from our simulation model.

1) Moveable sinks versus stationary sinks: First, the two approaches: with and without sinks movement are compared. In the experiments, the network consists of 16 randomly placed nodes in a 100x100 square meter area. At the initial simulation time, in both approaches, five sinks are placed at their optimal positions. Then, in one case, the sinks can move and in the other, they remain static. In this first set of experiments, data packets generation is performed by groups of nodes. Each group is composed of up to 5 nodes. The identity of these groups is changed every thousand seconds. This relative stability of packet production zones allows the sinks to get closer to these zones. The results depicted in figures 9, figure 10 and figure 11 show that our approach preserves energy and increases the network lifetime. Figure 9 shows that the energy decreases slowly when the sinks are mobile. Figure 10 shows that we deliver more packets with less energy.

Figure 11 illustrates all the principles of our approach. When the sinks move towards the data generating zones, and since these zones are relatively stable, we observe a decrease in the consumed energy. The curve decreases when the zones of production change. The energy spent is then inevitably more important since the sinks are not in an optimal configuration anymore.

Others results we obtained, not illustrated here, show that the packet average delay is largely improved. The shape of the curve of the average delay per packet is very similar to that of the average energy consumed per packet presented in figure 11. This comes from the simple fact that the sinks, while approaching the transmitting sensors, decrease the distance separating them from these active sensors, and consequently decrease the communication delay.

2) Optimal locations versus random locations at the Initial time: Others experiments, not illustrated here, show that the effect of the sinks location at the initial simulation time has no real impact on performance. The longer the simulation time is, the smaller is the influence. This means that the sinks could be deployed randomly without really impacting the performance.

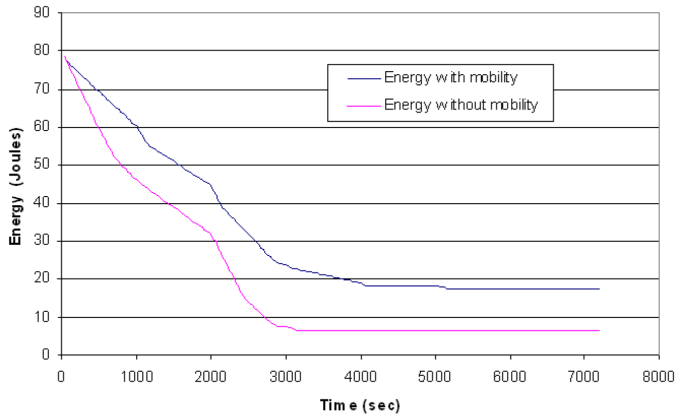

Fig. 9. Energy dissipation with and without mobility

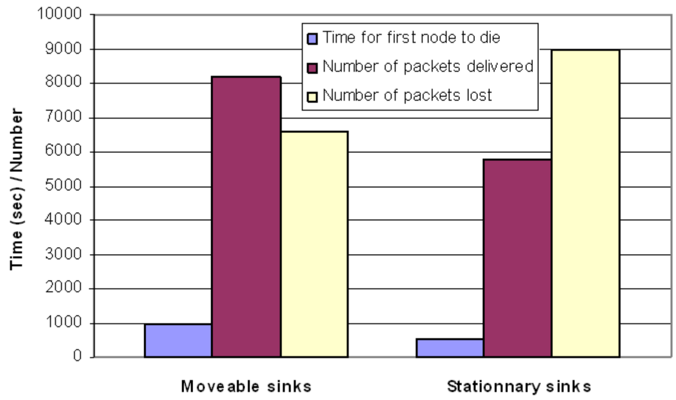

Fig. 10. Nodes lifetime and packet transfer results

3) Straight line movement versus smart movement: A major part of our work was dedicated to the smart location of the sinks by defining a restricted area of freedom to move the sink in a progressive way towards its optimal position while avoiding the unfavorable positions as well as possible. This test aims at highlighting the profit of our approach. For this purpose, we took again the same configuration but in one case, we disabled the smart movement, and in the other, the sinks move in a straight line towards their optimal position at a maximum speed. The results, not presented in this paper, show that the smart repositioning consumes less energy. We also noticed that with the same energy consumption, the network with smart movement delivered 8162 messages versus 7516 for the network with straight line movement, this leading to a non negligible benefit of 8.5%.

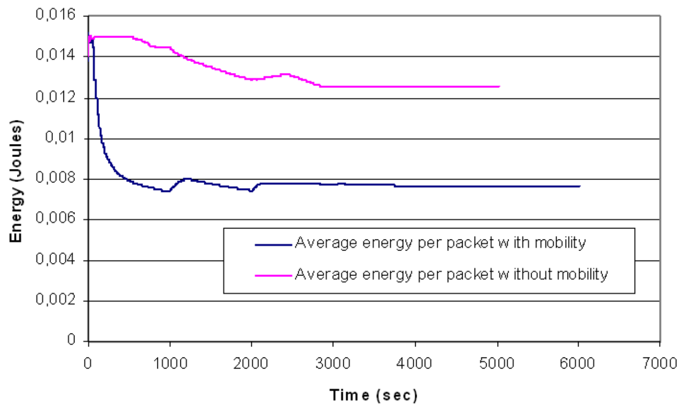

Fig. 11. The average energy consumed per packet

4) Limitation of our approach: chaotic data production: As discussed before, the efficiency of sinks repositioning within the network is highly dependent on the stability of data production areas. If these zones change unceasingly, the sinks may not have enough time to approach their optimal locations. Then the benefit of our technique is likely to be low. It is even very likely that the performance would be worse than for motionless sinks. In this second set of experiments, the data production zones change more quickly (every hundred seconds) and are more dispersed. The sinks which have to move towards the optimal locations will be, when the data generation areas change, in very unfavorable locations and thus affect the overall network performance. These tendencies are illustrated in figure 12. We can note that the network with moveable sinks is less performing than the one with static sinks.

Figure 13 depicts the average energy consumed per packet. Between times 0 and 100 seconds, the sinks approach the optimal location which causes a reduction in the average energy consumed per packet. Then, when the nodes which produce packets change, the sinks come to be in a very unfavorable location and the average energy consumed per packet increases strongly. These results confirm that sinks repositioning can involve, in some cases, a loss of performance. However, it is interesting to notice that the optimal location of the sinks also depends on the coefficients values in the equation used for configuration evaluation. When tuning these coefficients, we observed a decrease in the loss of performance. One can for example make the optimum less sensitive to the heavily loaded

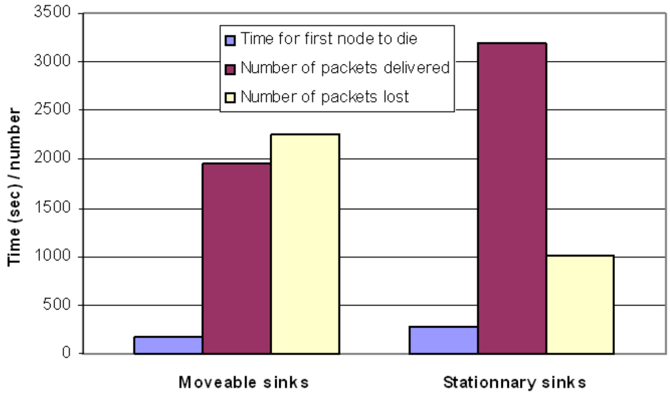

Fig. 12. Comparison of the two approaches

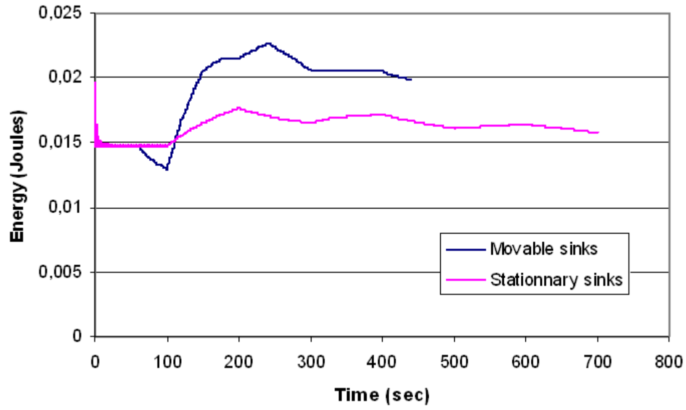

Fig. 13. Average energy consumed per packet for the two approaches

nodes, by increasing the coefficient $C2$. The sinks will then tend to move towards the highly data generation zones while keeping a central position in the network. In this scenario, we notice that the performances are better when the $C2$ parameter is increased.

Finally, we can conclude that the efficiency of our approach may vary. Generally, the more the data production zones are stable, the more the efficiency is important. Moreover, in such a situation, the best performance is obtained when choosing high values for the coefficients which take into account the flow in the network. On the other hand, when the zones of packet production change quickly and especially when their geographical locations are distant, the performance may be disastrous. In this case, motionless sinks approach may give better results. Therefore, a balance has to be found and the equations and their coefficients have to be adapted to the scenario.

V. Conclusions and Future Works

In this paper, we proposed a new approach for multi-sinks repositioning in a sensor network. This approach is based on previous works related to two fields: multiple sinks positioning and the unique sink relocation. Our approach has the advantage of considering a global level network view when relocating the sinks to their optimal positions. The definition of these optimal positions is achieved by the mean of a local search algorithm which derives an optimal sinks distribution. The optimal positions are largely affected by the areas of heavy traffic and thus are likely subject to continuous variations. Moving the sinks towards the areas of interest (in terms of information production), allows to obtain a power saving provided that a stability of these areas exists. In such a configuration, simulation results show a significative enhancement in the network lifetime. Conversely, in a configuration where data production is chaotic or changing fast relatively to sinks velocity, the performance can be very weak. Finally, in order to reach good performance of our approach, the equation and the coefficients of the evaluation function have to be adapted and tuned according to the chosen scenario.

Our future plan includes extending the approach to allow for an event mobility-aware method. In other words, since simulation results show that the performance of our approach is highly dependant on data generation areas stability, we are currently studying different mobility profiles of the observed event. We mainly focus on tracking strategies which we expect to be very helpful in some sensor applications where the observed phenomenon motion can be predicted and tracked.

VI. Acknowledgement

The authors would like to acknowledge M. Younis for his constructive comments and for providing his simulator which was very helpful for our implementation model.

References

[1] I. F. Akyildiz and al, "A survey on sensor networks", *IEEE Communication Magazine*, vol. 40, no 8, pp. 102-114, August 2002.

[2] K. Akkaya and M. Younis, "A survey On Routing Protocols for Wireless Sensor Networks", *Elsevier Ad Hoc Network Journal (to appear)*, 2004.

[3] H. Kim and al, "Optimal Multi-sink Positioning and Energy-efficient Routing in Wireless Sensor Networks", *Lecture Notes in Computer Science*, vol. 3391 no. pp.264-274, Jan. 2005.

[4] E. Iker Oyman and Cem ErsoyF, "Multiple Sink Network Design Problem in Large Scale Wireless Sensor Networks", *in the Proceeding of the International Conference on Communication*, ICC 2004.

[5] A. Bogdanov, E. Maneva and S. Riesenfeld, "Power-aware Base Station Positioning for Sensor Networks", *in the Proceeding of the 23rd IEEE INFOCOM*, 2004.

[6] M. Younis, M. Youssef and K. Arisha, "Energy-aware Routing in Cluster-Based Sensor Networks", *in the Proceedings of the 10th IEEE/ACM International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, (MASCOTS2002), Fort Worth, Texas, October 2002. no. 5.408.237, April 1995.

[7] M. Younis, M. Bangad and K. Akkaya, "Base-Station Repositioning For Optimized Performance of Sensor Networks", *in the Proceedings of the IEEE VTC 2003 - Wireless Ad hoc, Sensor, and Wearable Networks*, Orlando, Florida, October 2003.

[8] M. Younis, M. Youssef and K. Arisha, "Energy-aware Management for Cluster-Based Sensor Networks", *Computer Networks: The International Journal of Computer and Telecommunications Networking*, Volume 43, Issue 5, p 649-668, December 2003.

[9] K. Akkaya and M. Younis, "Energy-aware Routing to a mobile gateway in wireless sensor networks", *in the Proceedings of the IEEE Globecom, Wireless Ad Hoc and Sensor Networks Workshop*, Dallas, TX, November, 2004.

[10] K. Akkaya and M. Younis, "Sink Repositioning For Enhanced Performance", *in the Elsevier Computer Networks Journal (to appear)*, November 2004.