An Adaptive Localized Algorithm for Multiple Sensor Area Coverage

Antoine Gallais, Jean Carle

An Adaptive Localized Algorithm for Multiple Sensor Area Coverage

Antoine Gallais and Jean Carle

IRCICA/LIFL, Univ. Lille 1, INRIA Futurs, France
Email: {gallais,carle}@lifl.fr

***Abstract*— Wireless sensor networks are made up of hundreds of devices deployed over a distant or sensitive field to be monitored. Energy consumption is balanced by taking advantage of the redundancy induced by the random deployment of nodes. Some nodes are active while others are in sleep mode, thus using less energy. Such a dynamic topology should not impact the monitoring activity. Area coverage protocols aim at turning off redundant sensor nodes while ensuring full coverage of the area by the remaining active nodes. Providing k-area coverage therefore means that every physical point of the monitored field is sensed by at least k sensor devices. Connectivity of the active nodes subset must also be provided so that monitoring reports can reach the sink stations. Existing solutions hardly address these two issues as a unified one.**

In this paper, we propose a localized algorithm for multiple sensor area coverage able to build connected active nodes sets. We also show that a simple feature of the protocol, called the coverage evaluation scheme, can be enhanced to handle various k-area coverage problem definitions. Experimental results show that our coverage scheme is resistant to collisions of messages as k-area-coverage of the deployment area and connectivity of the active nodes set can still be ensured.

I. Introduction

A sensor network is a set of nodes in which a battery, a sensing module and a wireless communication device are embedded. Densely deployed over hostile or remote environments, their self-organization should provide full monitoring and pertinent data collection so that further heavy computation and analysis tasks could be achieved by better equipped machines (usually called sinks). Once thrown over sensitive areas, the sensor nodes become one-use-only since their batteries cannot easily be replaced or refilled. Energy is therefore implied to be the most critical resource of the system.

In order to increase their lifespan, and that of the constituted network, these objects are densely deployed so that only some of them are really needed for monitoring. Nodes can therefore alternatively be allowed to switch to sleep mode as soon as they are not required for the local monitoring task. The ensuing issue consists in these nodes deciding themselves whether to turn off or not so that the whole area remains fully covered. The area can also be covered once, twice or k times by a set of active nodes. From the point of view of a sink station, the relevance degree of a data sample or the confidence level of a monitoring alert could depend on the number of sensors that were implied in the measures. Thereby, enabling multiple sensor area coverage is a matter of prime importance. A set of sensors is able to cover k times an area if every physical point of the area is covered by at least k sensors. Another definition could be that the active nodes set should be divided in k distinct sets, each covering the area once. Both definitions will be detailed further and we will explain how our solution can handle both definitions.

Then, monitoring reports must obviously reach at least one of the sink stations. This requires that the active nodes set be connected. In this work, we consider connectivity as an important feature of our algorithm and do not rely on any fixed ratio of sensing and communicating radii. We show that, under ideal assumptions, a simple criterion helps preserve the active nodes set connectivity while interesting results can be provided once more realistic assumptions are considered.

We favor localized solutions since their communication overhead is significantly lower (no global view of the network is required) and they can be applied in sensor networks of any size and density. Indeed, each node makes its activity status decision solely based on decisions made by its communication neighbors. In localized solutions, topological changes (due to mobility, activity status modifications, or due to node failures or new incoming nodes) simply imply some modifications in the neighborhood of a node.

Many existing solutions first address the problem of 1-area-coverage and then try to generalize it to k. To the best of our knowledge, none has ever ensured connectivity independently from the ratio of sensing and communicating radii. Moreover, proposed protocols never show to what extent they can be resistant. Indeed, many solutions lie in clustering or distributed protocols in which correct communication is crucial. Meanwhile, no study about the impact of message loss is ever conducted.

We imagine the activity of a randomly deployed sensor network in a rounded fashion. At the beginning of each round, every node must decide its activity status before either monitoring for the remaining of the round or turning into a passive mode until the next decision phase. During this phase, each node selects a timeout, listening to messages from neighboring nodes. It then takes its decision solely based on the messages it has received from active neighbors with shorter timeouts. A variant of this scheme consists in enabling retreats from nodes which have first decided to be active but have later learned about new active covering neighbors. In both variants, no *a priori* neighbor knowledge is needed. This allows our protocol to be extremely resistant to any message loss (due to some collisions or simply to the radio channel randomness [1]). Experimental results show that message losses simply impact the number of active nodes without ever endangering the multiple sensor area coverage.

In this paper, section II exposes various notations and assumptions we have formulated for this work. In section II-D, we then discuss two different formulations of the k-area coverage and explain which one we have decided to address. After reviewing the existing contributions in section III, we describe our solution in section IV, along with some details about aspects such as the timeout computation or the connectivity preservation. We show in section V that our solution can ensure k-coverage with a connected set of active sensors, whose number can be significantly reduced.

II. Notations and assumptions

A. Communication and sensing models

Each sensor has communication capacity. We assume equal communication ranges for each sensor, denoted CR. Therefore two sensors are communication neighbors, or simply neighbors, if and only if the distance between them is at most CR. Consequently, the communication is symmetric: if a node u can communicate with a node v, then v can also communicate with u. The degree of a node is the number of neighbors it has. The density of the network is the average degree of a node in the network. Nodes monitor an area using their various embedded sensing modules. The sensing radius of a node is denoted by SR. Monitored area of a sensor is modeled as a disk of radius SR, centered on the node itself.

B. Notations

Given a set of nodes, noted as V, deployed over an area A. Let $S(u)$ be the set of physical points that a node u can sense (as already stated before, $S(u)$ is modeled as a disk in this paper) :

$$S(u) = \{p \mid d(u, p) \leq SR\}$$

The k-coverage of an area A is so formulated as follow:

$$A_{k-covered} \Leftrightarrow \forall p \in A, |\{v \in V \mid p \in S(v)\}| \geq k$$

For the sake of clarity, a k-covered node u will stand for a node whose monitored area $S(u)$ is k-covered.

$$u_{k-covered} \Leftrightarrow S(u)_{k-covered}$$

C. Assumptions

Before any activity, sensor networks must be deployed. The deployment can be either deterministic or random, depending on the application. We assume nodes to be randomly deployed and to be static. We can therefore ensure sensors to know their respective geographical positions. Sensor positioning problem has already been addressed in literature (see [2]). During this process, two nodes that would obtain identical positions, due to a non precise algorithm or any other anomaly should communicate in order to distinguish one from the other. Precise details of such a coordination are out the scope of this paper. Meanwhile, this allows us to consider that nodes have unique identifiers; their positions. We finally assume that devices are *time synchronized* so that activity decisions can occur in rounds. Synchronization can be achieved by applying some network protocols (see [3] for a survey).

D. Two formulations for k-area-coverage

In wireless sensor networks, monitoring nodes are responsible for either regularly sending data samples or launching an alert once an abnormal event occurs. Both the relevance and trusting levels of these reports may strongly depend on the number of originating sensors. Thereby, enabling multiple sensor area coverage is a matter of prime importance. A set of sensors is able to cover k times an area if every physical point of the area is covered by at least k distinct sensor nodes. A more restrictive definition could be that the active nodes set should be divided in k distinct subsets, each covering the area once. We so propose two definitions for the k-area-coverage problem:

Definition 1: An area is k-covered if every physical point is covered by at least k active sensors.

In the remaining of this paper, we will designate this issue as the flat k-area-coverage problem. Another definition is:

Definition 2: An area is k-covered if there exist k distinct sets of sensors so that each one fully covers the area.

In the remaining of the paper, such sets will be designated as virtual activity layers or shortly activity layers. We will then note this second definition of the k-area-coverage issue as the layered k-area-coverage problem.

Note that solving the layered k-area-coverage problem necessarily implies the flat k-area-coverage issue to be solved also. Meanwhile, ensuring any physical point of the area to be covered by at least k sensors (flat approach) does not imply k-area-coverage by k distinct layers of nodes (layering approach). In other words, the area can be k-covered by a set of nodes from which no k disjoint subsets can be extracted. Still, these two formulations of the k-area-coverage problem can each fit to various classes of applications. We show in section IV-A how both can be easily handled by a node which must evaluate its coverage.

III. Related Work

The problem of simple area-coverage has already been largely studied and a comprehensive literature review of existing solutions for sensor area coverage issue is described in [4]. We now review some of the papers that exist in literature concerning k-area-coverage.

In [5], authors consider networks in which nodes are in sleep mode most of their lifetime. This paper firmly formulates the lower bounds for the number of nodes that should be deployed to ensure k-area-coverage.

Abrams, Goel and Plotkin [6] study the problem of partitioning the sensors into covers so that the number of covers that include an area, summed over all k areas, is maximized. They so address the layered k-area-coverage problem (see def. 2 in sec. II-D). Three approximation algorithms, assuming k is fixed, are described. Randomized algorithm assigns to each sensor one of k covers at random. In distributed greedy algorithm, each sensor sets a timeout and listens to decisions made by neighbors, increasing the counter in the appropriate set for each message announcing decision by a neighbor. When timeout expires, each node selects a set for which the corresponding counter is minimal. Centralized greedy algorithm adds some weights but otherwise runs a similar procedure. In this article, no discussion is made concerning connectivity preservation with different ratios of communication and sensing radii.

Actually, network connectivity is rarely treated in existing works on k-area-coverage. Meanwhile, if $CR \geq 2SR$, k-connectivity is proved to be achieved once the network is k-covered [7]. Therefore, most of reviewed solutions rely on this theorem to focus on area coverage only without addressing the problem of the connectivity preservation.

In [8], Gupta, Das and Gu show that the connected coverage problem is NP-hard. In [9], this statement is generalized for connected k-coverage problem. Zhou, Das and Gupta [9] propose three algorithms to solve this issue. The first one is a centralized greedy protocol while the two others are distributed versions of it. The distributed greedy algorithm is very close to the distributed approximation algorithm that is provided in [8]. Let M be the set of nodes that should k-cover the query region. Initially, a random sensor is in M. Then, it must select the best candidate sensor and path to complete both the coverage and the connectivity. This is done by broadcasting a search message to all sensors within $2r$-hops where r is said to be the *link radius* of the network, that is the maximum number of hops that separate two nodes of the network whose sensing areas intersect (then $2r$ might be very large and the broadcast will be much energy-consuming). After a treatment on received response messages, a candidate sensor is added to M. This is repeated until the query region is k-covered. The communication overhead induced by this phase can be very high and authors propose a distributed priority algorithm to reduce it. Meanwhile, in this solution, every node must gather $max(t, r)$-hop (t is a constant and r is the *link radius*) neighborhood information (that include the unique priorities of nodes). Once again, the amount of information that needs to be collected might lead to high energy consumption. Indeed, t is fixed at 2 in the simulation results and so at least 2-hop information is required. Although no discussion is provided on how t can be fixed, authors propose a simple computation of r depending on the network density. Meanwhile, r equals to $2SR/t + 1$ in dense networks ($SR = 4$, $t = 2$), that is 5-hop information could be required. Once again, the induced communication overhead of this solution may be very high and so, resistance once messages get lost may not be sufficient.

Actually, most of contributions that address the k-area-coverage issue do not consider either fully localized approaches nor unreliable wireless communications. Still, it would be very hard to enable acknowledgments for control traffic in wireless sensor networks. Due to the very high communication densities, it would lead to both huge communication overhead and power consumption. We now present our approach for enabling k-area-coverage with connected active nodes sets in wireless sensor networks.

IV. Proposed contribution

In this section, we first detail the mechanisms that allow a sensor node to evaluate the coverage provided by its neighbors. Then, we will explain our protocol along with the timeout computation and the connectivity preservation.

A. Evaluating k-area-coverage

In order to evaluate multiple area coverage, a node must have a required coverage degree, noted as k. It can therefore compare the provided coverage level to k, thus deciding to be fully k-covered or not. Note that such a parameter could be heterogeneous over the set of nodes. Yet, assuming identical required coverage degrees for all sensor devices allows us to better focus on the algorithm itself.

We now show how any node of the network could locally decide to be active or not while the deployment area remains k-covered according to the flat k-area-coverage definition. Given a node u, the basic idea is that u must evaluate the coverage provided by its neighbors. Several coverage evaluation schemes exist and, most of the time, the sensing area is divided following a grid whose every point must be covered by at least k neighbors. Hence, u can evaluate whether or not it is k-covered.

Another way for u to evaluate its k-coverage is to consider virtual activity layers, thus addressing the layered k-area-coverage issue. An activity layer is a set of nodes that have decided to be active. We assume that a number of layer is included in the messages. Therefore, u can sort its neighbors according to this value. Then, u evaluates the coverage provided by each activity layer by using the intersection-based scheme detailed in [7]. It is fully k-covered if and only if at least k virtual activity layers fully cover its area $S(u)$.

B. Protocol description

We propose a localized algorithm that can be applied to time-synchronized networks in which every sensor is aware of its geographical position. Activity of the network is imagined in a rounded fashion. At the beginning of each round, every node must decide its status (i.e. active or passive) before either monitoring for the entire round or turning into a passive mode until the next decision phase. During this phase, a node u selects a timeout while listening to messages from neighboring nodes (the timeout computation is detailed further in sec. IV-C). Once the timeout ends, u takes its activity decision based on known neighboring nodes. It so evaluates its coverage according to the appropriate coverage evaluation scheme (see sec. IV-A).

If completely k-covered according to the flat k-area-coverage issue, u decides to be passive and turns into sleep mode. Otherwise, u remains active and sends a positive acknowledgment. This message contains the values of its communicating and sensing radii along with its position. Any node with a longer timeout that receives this message will therefore add u to its neighbor table. We can so easily ensure k-area-coverage with such a protocol.

Meanwhile, as solving the layered k-coverage issue implies solving the flat problem, we decided to design an adaptive protocol able to solve the layered k-area-coverage problem; Nodes still listen for messages during a given timeout before making their activity decision and choosing an activity layer whose number is included in the messages. Once having evaluated its coverage (see section IV-A), if a node u is not k-covered, it must remain active and it sends an activity message to announce its status. In this case, nodes are said to be in *Positive-only* mode, further noted as PO. It also includes in this message the number of the chosen activity layer. There are various ways of choosing the layer: u can decide to be either active at the uncovered layer with the lowest number or at the layer which provides the less coverage for instance. In this paper, u chooses the uncovered activity layer which has the lowest number.

We improved our algorithm by introducing retreat messages. In case u had first decided to be active, it can regularly reevaluate its coverage as it receives new positive acknowledgments. Then, if it happens to be k-covered, it can decide to turn into sleep mode by sending a retreat message. Nodes that have not decided yet will be able to remove this node from their neighbor table, thus preventing it from being considered during the coverage evaluation process. This variant is called *Positive and Retreat*, further noted as PR.

Let us now briefly discuss the timeout computation.

C. Timeout computation

In our protocol, when a round starts, every node selects a timeout and evaluates its k-coverage once its timeout expires. As energy is the most critical resource, the timeout of a node could be inversely proportional to the remaining energy. This would allow weak nodes to decide after the others, therefore increasing the probability of being k-covered and so of turning into sleep mode. Meanwhile, due to edge effects and other application-dependent parameters (energy consumption model, etc.), such a timeout computation scheme is hard to achieve and fairness regarding energy consumption may not be ensured. For this work, we so adopted a simple timeout computation, based on a random number generator. We assume, for simplicity, that any two neighboring nodes would select different random numbers. However, we will show that this requirement is not mandatory at all. Two nodes selecting equal timeouts will simply take their decisions at the same time, therefore ignoring each other. We will show in section V to what extent it can impact the performances of our algorithm.

We now explain how connectivity of the set of active nodes is ensured, independently from any fixed ratio of sensing and communication radii.

D. Connectivity: solution and discussion

Existing protocols normally assume that the communicating range is at least twice the sensing range (e.g. [10]) and therefore the connectivity is ensured for sensors with partially overlapping sensing areas. This has been generalized to k-coverage and k-connectivity but does not hold anymore once $CR < 2SR$. In such a case, a simple connectivity test is added in our activity decision process. The knowledge of positions and transmission ranges of active neighbors is sufficient to learn their connectivity graph, and Dijkstra's shortest path algorithm can be applied to test their connectivity.

We will further observe that, when messages get lost, solving the k-area-coverage problem may help preserve the overall connectivity of the active nodes set. As k distinct layers are built, redundancy is increased thus implying a higher guarantee for the connectivity of the active nodes set once non ideal assumptions are considered.

V. Experimental results

Experimental results were obtained from randomly generated connected networks. Nodes are deployed over a $6 * 6$ rectangle area. Communication and sensing ranges (respectively CR and SR) are equal to 1. Please note that, unlike some existing propositions (see section III, the independence between our coverage evaluation scheme and our connectivity criterion helps us handle independent communicating and sensing radii. Nevertheless, fixing these parameters allows us to focus on k-coverage and other features of our protocol.

We are now giving the results of our simulations. Results were obtained from high number of simulated topologies so that the confidence intervals are sufficiently low, as observed on the graphs. As already explained earlier (see section IV-D), we only focus on the layering approach.

A. General results

We evaluate our protocol by, first, measuring the coverage it provides. Every time we will talk of full k-area-coverage, it will mean that every layer $i \leq k$ fully covers the area (see def. 2 in sec. II-D). In order to evaluate connectivity, we simulated several random topologies, applied our algorithm and finally computed how many sets of active nodes were connected. Finally, we especially looked at the percentage of active nodes that were required by our variants, PO and PR, previously defined in section IV-B.

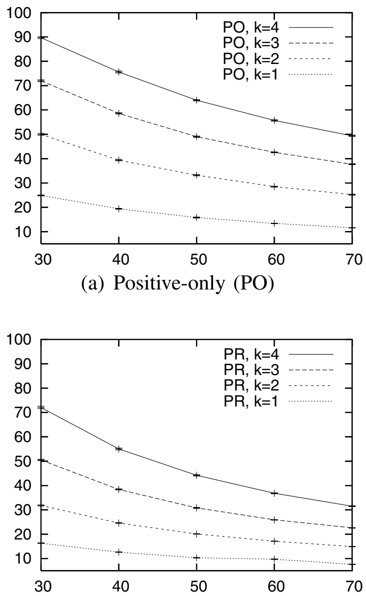
In this first subsection, we give some results about networks that were simulated in ideal mode. By ideal mode, we mean that no message failure was ever simulated and that nodes had unique timeouts, thus being aware of every other active node with shorter timeout in the neighborhood.

Under ideal assumptions, both approaches guarantee full k-area-coverage and connectivity of the set of active nodes. The percentage of active nodes helps us evaluate which variant might be the more efficient. Fig. 1 shows the average percentage of active nodes that are required for each density and coverage degree.

Positive-only (PO)															
Coverage degree	$k=2$					$k=3$					$k=4$				
Density	30	40	50	60	70	30	40	50	60	70	30	40	50	60	70
$timeout \in \{0..31\}$	50.8	40.8	34.3	29.2	25.8	73.8	59.7	50.3	43.5	38.7	91.7	77.9	66	57.6	51
$timeout \in \{0..7\}$	55.4	45.7	39.1	34	31	80.6	67.2	57.2	51.1	46.3	97.1	86.7	76	68.1	62.1
Positive and Retreat (PR)															
Coverage degree	$k=2$					$k=3$					$k=4$				
Density	30	40	50	60	70	30	40	50	60	70	30	40	50	60	70
$timeout \in \{0..31\}$	20.6	15.6	12.4	10.6	8.9	34	23.9	19	15.8	13.6	54.3	34.3	26	21.1	18.1
$timeout \in \{0..7\}$	20.7	15.3	12.5	10.6	9	33.5	24.2	18.8	15.8	13.4	54.4	33.8	26	21.4	18.2

TABLE I

ACTIVE NODES (%) FOR BOTH PO AND PR VARIANTS. TIMEOUTS ARE RANDOMLY CHOOSEN IN DIFFERENT INTERVALS.



(a) Positive-only (PO)

(b) Positive and retreat (PR)

Fig. 1. Active nodes (%) required by both variants versus network density using ideal mode.

As expected, the higher the density, the lower the active nodes percentage. We can also notice that, compared to PO, PR efficiently decreases the number of active nodes (8% at density 70 versus 12% for PO when $k=1$ and only 32% when $k=4$ versus 50% for PO). PR so ensures k-area-coverage with a connected set that involves few active nodes. We so consider PR as a very well adapted solution for the connected k-area-coverage problem. Let us now observe to what extent it can remain efficient.

B. Non unique timeouts

So far, the timeouts used by the nodes were unique. As such an assumption may sound unrealistic, hardly feasible with reasonable latency or requiring a too much complex distributed computation task, we have observed the impact of non unique timeouts on our protocol and especially on our PR variant. Each node randomly selects a timeout in a given virtual window. Two nodes with identical timeouts take their decisions at the same time, thus ignoring each other. The size of the timeout window is noted as TOW_{min}. We simulated timeout windows of $[0, TOW_{min}-1]$, with TOW_{min} in $\{8, 32\}$. If we consider PO, there is no reason why k-coverage or connectivity would not be ensured. Indeed, two nodes ignoring each other simply have less knowledge when making their activity decision. No wrong decision can be made but nodes have a higher probability to remain active. This is confirmed by results that are summarized in Tab. I. With our PO variant, the tighter the window is, the more nodes decide to be active.

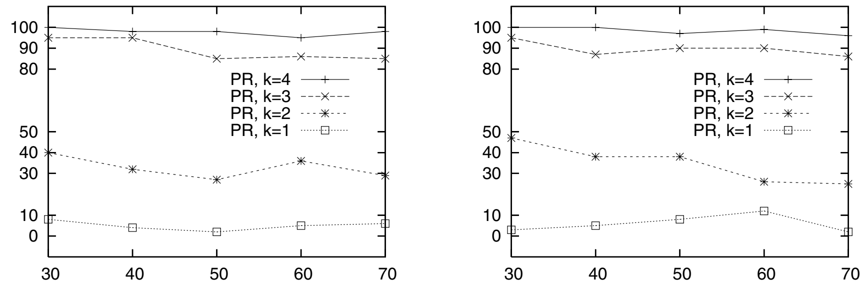
Concerning PR, it still generates less active nodes than PO. As for PO, the percentage of monitoring nodes increases as the timeout window gets tighter. Both variants so look like having similar behaviors. Meanwhile, we can guess that PR should fail to preserve the k-area-coverage and connectivity of the active nodes set. Indeed, if two nodes decide to retreat at the same time, they will ignore each other. If one of them was crucial to the coverage of the other, then a coverage hole can occur. We decided to compute the coverage achieved by each virtual activity layer. Table II shows the portion of the deployment area that was covered by each given layer.

Network density	30	40	50	60	70
$timeout \in \{0..7\}$					
1st layer	99	98.5	98.3	98.2	97.8
2nd layer	98.2	98.1	97.1	97.1	96.9
3rd layer	97.3	96.6	96.5	96	95.8
$timeout \in \{0..31\}$					
1st layer	99	98.3	98.4	98.1	98
2nd layer	98.2	97.4	97.1	97.1	97.3
3rd layer	97.4	96.9	96.3	96.2	95.9

TABLE II

COVERAGE OF EACH VIRTUAL LAYER (%) ACHIEVED BY PR. TIMEOUTS ARE NOT UNIQUE AND REQUIRED COVERAGE DEGREE IS 3.

We can observe that, with the tightest window ($TOW_{min}=8$), the coverage falls only to less than 96%. If PR can not completely ensure k-coverage once timeouts are not unique anymore, Tab. II yet shows that the coverage loss is not very important. Moreover, it is distributed over the layers. Basically, few distinct values of timeouts



(a) $timeout \in \{0, .., 7\}$ (b) $timeout \in \{0, .., 31\}$

Fig. 2. Connected Active nodes sets (%) built by PR versus network density. Timeouts are not unique but no message failure can ever occur.

are available. Therefore, nodes hardly consider each other decisions. This is true for both positive and retreat messages. This strongly decreases the probability of non receiving a retreat message from a node whose positive acknowledgment has been received earlier. Moreover, as the required coverage degree k increases, more nodes must become active, thus reducing the amount of potential retreating nodes. More redundancy is induced and helps improving the performances of PR. Indeed, Fig. 2 shows the average number of connected sets of active nodes that are generated by PR. Each figure stands for a given timeout window while each curve stands for a given coverage degree, k.

Very few connected sets of active nodes are built by PR when k equals to 1 (always lower than 1 out of 10 whatever the values of k and TOW_{min}). However, the higher k is, the more connected topologies there are since the redundancy of active nodes is increased. Indeed, Fig. 2 shows, that for any number of TOW_{min} values, the number of connected active nodes sets is always close to or over 90% as soon as the coverage degree k is at least 3.

Finally, even if timeouts are not unique, our variant PR, that would be the more prone to fail (because of potentially ignored retreat messages), still ensures high level of coverage and builds a relatively high number of connected sets. Moreover, Tab. I shows that the number of active nodes that are involved is not very high. For $k = 3$, it is always lower than 34%, reaching percentages lower than one fifth of the nodes once the density is greater or equal to 50.

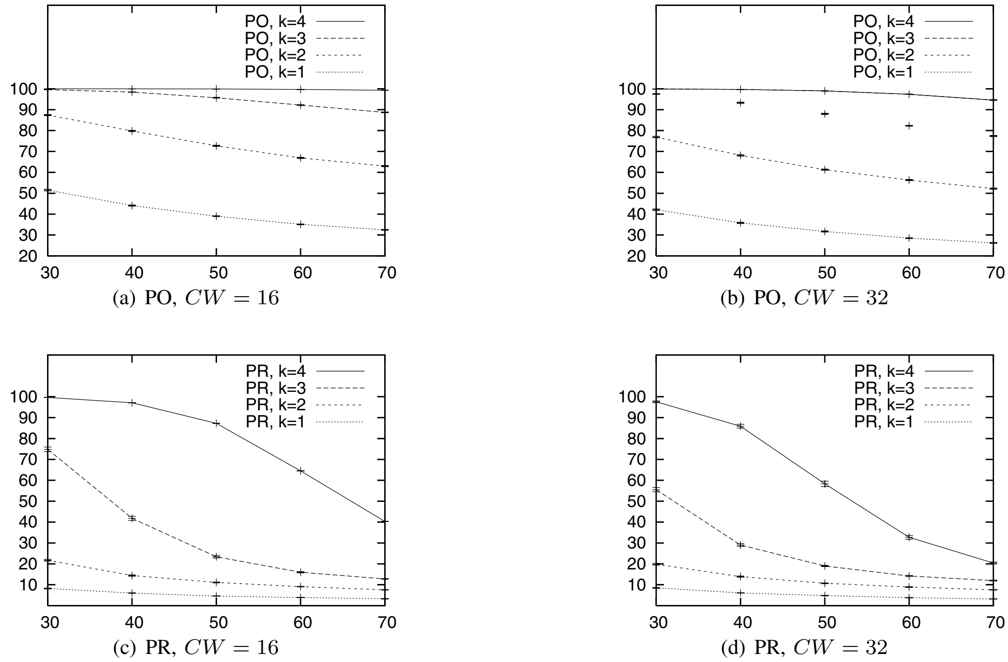
C. Overcoming message losses

Until now, every activity message sent by a node was received by all of its neighbors. Meanwhile, in real environments, radio channel randomness or message collisions forced us to take a new look at this assumption. In this section, we so consider that communication between two nodes can fail.

In order to model the message failures, we basically had two options. The first one was to consider a probabilistic radio propagation model. If PO would still build connected active nodes sets that k-cover the area, nodes using PR could be unable to receive some of the retreats from neighbors from which they have previously received the positive acknowledgments. Meanwhile, substituting such model to our unit-disk graph would have implied another consideration. Indeed, connectivity is then hard to characterize. Some schemes could have been applied but it would have implied either simplified radio links considerations (only take every link whose probability is above a given threshold and compute the connectivity of the resulting subgraph) or heavy computations (e.g computing the probability of every path between every pair of nodes and then evaluating the overall probability of connectivity for the set). This would require a much longer dissertation and we have so opted for models in which we still use unit disk graphs.

The second option was to consider message collisions. In such a case, losses are straightly impacted by the network density. The more nodes there are in a communication zone, the more collisions there should be. Furthermore, from the point of view of a receiving node, it is a random process. Indeed, except if some backoff information is exchanged, it is impossible to determine on which link collisions are the more prone to occur. Yet, collisions could be avoided by one of the many already standardized mechanisms (RTS/CTS messages in 802.11 standard for example or acknowledging every message in order to allow retransmissions once the reception has failed, etc.). Meanwhile, the kind of control messages that we use should not be acknowledged. The so induced communication overhead would be a too much important source of energy consumption for wireless sensor nodes as density can be as high as 70 nodes per communication area with most of them acknowledging every reception. Therefore, regarding the context of highly dense wireless sensor networks, deployed in sensitive areas, considering message collisions is a pertinent choice.

Collisions can be considered as random losses and overcoming this kind of losses necessarily means that the protocol is robust. We considered a MAC layer in which no reemission ever happens. Once a node has made its activity decision, its activity message is sent after a random backoff. This backoff is computed within a contention window, whose size is noted as CW. A node u can not receive messages sent during the same time slot. Furthermore, since most of existing radio technologies do not allow simultaneous message reception and sending, we also assume that no message can ever be received by u during its own time slot.



(a) PO, $CW = 16$ (b) PO, $CW = 32$

(c) PR, $CW = 16$ (d) PR, $CW = 32$

Fig. 3. Active nodes (%) required by both variants versus network density. Timeouts are unique while messages can collide.

1) Active nodes and k-coverage maintenance: We simulated networks of various densities, with various coverage degrees and two different sizes for the contention window: 16 and 32. Fig. 3 shows the average percentage of active nodes. As in the ideal case (unique timeouts and perfect message receptions, see section V-A), PR basically generates less active nodes than PO. If we compare these results to the one summarized in Fig. 1, we can observe an overhead due to the loss of some messages. Nodes have less information and so more decide to be active. For instance, PO generated relatively low percentages of active nodes in the ideal mode (see Fig. 1). It now induces more than 60% with a contention window of size 16 as soon as the required coverage degree is greater than 1. Roughly, the overhead in terms of active nodes is about 20%. Moreover, when the required coverage degree is high (4 for instance), we can observe that always more than 95% of active nodes are needed. Meanwhile, as already explained before, less information does not prevent PO from ensuring full k-coverage of the area nor from preserving the connectivity of the active nodes set.

Let us more focus on PR. We can see on Fig.3(c) and Fig.3(d) that, thanks to the retreat messages, the number of active nodes slightly differs from the results obtained before. If too many nodes are selected, retreat messages help some of them to reconsider their activity status, thus reducing the impact of message losses. Meanwhile, since some retreats can be lost due to collisions, the direct consequence is that PR can not guarantee full k-coverage anymore.

Fig. 4 shows the coverage of each virtual activity layer. As density increases, PR can not avoid the more and more important loss of coverage. However, the coverage of each layer is still maintained above 95% which might be sufficient in most of potential monitoring applications. We can finally notice that, for a given density, the coverage provided by a layer does not depend on its number. For instance, when CW equals to 32, at density 40, the first layer provides less coverage than the second while this is reversed at density 60. This could look surprising since nodes that are not fully k-covered first decide to be active at the layer with the lowest number. Meanwhile, as they reconsider their status upon later receptions of positive acknowlegments, they can decide to retreat, whatever their current virtual activity layer is. Therefore, the loss of coverage is not necessarily fairly distributed over the virtual activity layers and does not depend on its number.

2) Preserving connectivity: As already observed in V-B, connectivity can be difficult to ensure, especially for PR (PO always preserves both k-coverage and connectivity since it

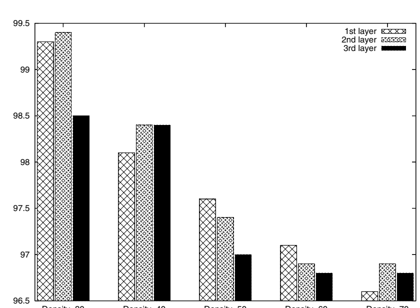


Fig. 4. Coverage of each virtual activity layer (%) achieved by PR. Timeouts are unique while messages can collide ($CW = 32$). Required coverage degree is 3.

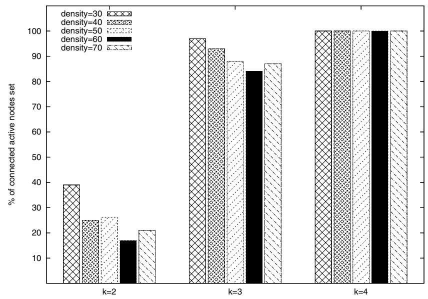


Fig. 5. Connected active nodes sets (%) built by PR. Timeouts are unique while messages can collide ($CW = 32$).

always has an uncomplete but yet correct knowledge of the neighborhood). This was the case with non unique timeouts and is still once messages get lost. Fig. 5 gives the average number of connected active nodes sets that are built by PR for three coverage degrees (2, 3 and 4) and five densities. Roughly, the more dense the network is, the less connected topologies there are, which is logical as more collisions occur. Meanwhile, as the coverage degree is increased, more and more active nodes sets are connected. For instance, when k at least equals to 3 and for all densities, PR builds more than 8 connected sets out of 10 with only 20% of active nodes. In lower densities, PR can reach more than 90% of connected active nodes sets. Once again, such an improvement in terms of connectivity preservation is due to the higher redundancy of active nodes thanks to a higher required coverage degree k.

VI. Conclusion

In this article, we have proposed to solve the k-area-coverage problem with a fully localized and adaptive approach. We have summarized the two existing formulations for this issue and have finally decided to address the layered k-area-coverage issue. Yet, we have given the process that every node should run in order to answer the flat formulation. We also introduced the notion of virtual activity layer and proposed a simple scheme to help nodes evaluate their k-area-coverage. We have shown that our two variants, PO and PR, were able to build k distinct layers consisting of low percentages of active sensor devices. Under ideal assumptions, both our variants could ensure full k-area-coverage with a connected set of active nodes. Then, further analysis on the basis of our assumptions was provided and the impact of more realistic ones was observed. Considering non unique timeouts and message collisions, PO was still able to ensure full k-area-coverage via a connected set while our variant PR provided good results with assumptions reshaped in the worst case, maintaining a low number of active nodes while suffering from minor coverage losses at each virtual activity layer. Moreover, as the coverage degree k increases, the overall connectivity of the active nodes set is most of the time preserved by PR.

Our current work therefore consists in improving this variant by enhancing the retreat process. Indeed, in this paper, as soon as a node is k-covered, it can decide to retreat, provided that it had first decided to be active (or else it would be passive already). Regular activity layer modifications could also be possible upon later information received from active nodes. Still, a trade-off between real gain in terms of either number of active nodes or reached coverage degree should be studied.

Acknowledgment

This work was partially supported by a grant from CPER Nord-Pas-de-Calais/FEDER TAC COM'DOM, CNRS National platform RECAP and European FP6 IST IP Project WASP.

References

[1] L. Quin and T. Kunz, "On-demand routing in MANETs: The impact of a realistic physical layer model," in *Proceedings of the International Conference on Ad-Hoc Networks and Wireless (ADHOC-NOW)*, Montreal, QC, Canada, 2003, pp. 37–48.

[2] J. Bachrach and C. Taylor, "Localization in sensor networks," *Handbook of Sensor Networks (I. Stojmenovic, ed.), Wiley*, pp. 277–310, 2005.

[3] K. Romer, P. Blum, and L. Meier, "Time synchronization and calibration in wireless sensor networks," *Handbook of Sensor Networks: Algorithms and Architectures (I. Stojmenović, ed.), Wiley*, pp. 199–238, 2005.

[4] D. Simplot-Ryl, I. Stojmenović, and J. Wu, "Energy-efficient backbone construction, broadcasting, and area coverage in sensor networks," *Handbook of Sensor Networks: Algorithms and Architectures (I. Stojmenović, ed.), Wiley*, pp. 343–380, 2005.

[5] S. Kumar, T. H. Lai, and J. Balogh, "On k-coverage in a mostly sleeping sensor network," in *Proceedings of ACM Mobile Computing and networking (Mobicom)*, Philadelphia, PA, USA, 2004, pp. 144–158.

[6] Z. Abrams, A. Goel, and S. Plotkin, "Set k-cover algorithms for energy efficient monitoring in wireless sensor networks," in *Proceedings of International Symposium on Information Processing in Sensor Networks (IPSN)*, Berkeley, CA, USA, 2004, pp. 424–432.

[7] X. Wang, G. Xing, Y. Zhang, C. Lu, R. Pless, and C. Gill, "Integrated coverage and connectivity configuration in wireless sensor networks," in *Proceedings of ACM Conference on Embedded Networked Sensor Systems (SenSys)*, Los Angeles, CA, USA, 2003, pp. 28–39.

[8] H. Gupta, S. Das, and Q. Gu, "Connected sensor cover: Self-organization of sensor networks for efficient query execution," in *Proceedings of ACM Mobile Ad Hoc networking and Computing (MobiHoc)*, Annapolis, MD, USA, 2003, pp. 189–200.

[9] Z. Zhou, S. Das, and H. Gupta, "Connected k-coverage problem in sensor networks," in *Proceedings of 13th International Conference on Computer Communications and Networks (ICCCN)*, Chicago, IL, USA, 2004.

[10] H. Zhang and J. C. Hou, "Maintaining sensing coverage and connectivity in large sensor networks," *Wireless Ad Hoc and Sensor Networks: an International Journal*, vol. 1, pp. 89–123, 2005.