Performance Analysis of CGS, a k-Coverage Algorithm based on One-hop Neighboring Knowledge

Alexandre Pocquet*, Bernard Cousin†, Miklos Molnar‡ , and Patrice Parraud§

*IRISA, MACCLIA laboratory of Military Special School of Saint-Cyr, France

†IRISA, University of Rennes I, France

‡IRISA, INSA Rennes, France

§INSA Rennes, MACCLIA laboratory of Military Special School of Saint-Cyr, France

Abstract—**Controlled Greedy Sleep (CGS) is a recent k-coverage algorithm based on one-hop neighboring knowledge. This paper presents the Controlled Greedy Sleep algorithm and highlights its best use cases thanks to performance analysis. Studies have been performed within a ring topology. This speci£c topology enables CGS validation and the design of an energetic model. They are used to compare CGS performance results with the maximal performances that could be expected. The ring topology allows us to determine analytically the minimum, maximum and relative values of our performances criteria which are: the quality, the cost and the duration of the k-coverage. According to the previous criteria we show in this paper that CGS is before anything designed to guarantee the maximum quality of coverage. We prove the CGS robustness when there are message losses. We show that CGS relative performances are improved when the overdosing and the communication cost increase. This is mostly because sensors just need to send at most three CGS messages per period and only need to know their one-hop neighbors.**

I. INTRODUCTION

Many new applications will use wireless sensors to provide target detection, remote environmental monitoring and localization services [1] [2]. High discrimination and robustness require the coverage of each points of the area under observation by at least k sensors simultaneously. This requirement is called k-coverage.

Wireless sensors which are used for this kind of applications could be unreliable and are limited in computational speed, electrical power (energy) and, communication and measurement ranges. But they are inexpensive, easy to produce and to install. That permits to overdose the space area with sensors, i.e. to deploy a higher number of sensors than strictly required to provide the k-coverage. Thus some sensors can be put in an asleep state to save energy. In this context, the challenge is to select a sequence of sensor subsets which will be minimal and provide an ef£cient k-coverage of the area for the maximum duration. One problem is that the selection of such minimal active sensor subsets is NP-hard to compute [4]. Thus approximate algorithms are proposed. Moreover that requires some knowledge of the network topology: sensors have to exchange some additional control messages. The forwarding of control messages adds to the burden of the sensors, it is why we propose an algorithm which is based only on neighboring exchanges. As messages can be delayed or lost, this paper studies the impact of these phenomena on the k-coverage quality.

Controlled Greedy Sleep (CGS) [6] is a Distributed Priority Algorithm [5] [3] which periodically computes a k-coverage provided by a subset of active sensors. CGS requires few computations and the CGS messages are not forwarded (CGS control message are only exchanged between direct neighbors). The purpose of this paper is to analyze CGS performances according to our criteria which are the quality, the cost, and the duration of the k-coverage. After a general presentation of CGS and a description of the performance criteria of the problem, we introduce a simple ring topology. This speci£c topology enables an analytical study of the problem. All minimal and disjoint active sensor subsets of this topology can be easily computed. We also introduce an energetic model which takes into account the communication cost. This topology and this model is used to ascertain the maximal CGS global performances which is taken as references to normalize our simulation results. Finally, the simulation results exhibit the best CGS use cases and the adequacy of CGS to ensure a full k-coverage when possible.

II. CGS ALGORITHM

A. Algorithm objectives

CGS has two main objectives:

- Provide full k-coverage over a certain set of space points.
- Maximize the duration of the full k-coverage.

For these two objectives, CGS is based on the periodic selection of an active sensor subset. Sensors by themselves decide if they must be active by taking into account their own energy level, the coverage ratio of the space points that they can cover, and the decisions already taken by their neighbor sensors.

So each sensor has just to know data from its one-hop neighbors. This makes CGS easily to distribute and reduce

the number of messages and the computation costs.

Sensors have limited communication and measurement ranges. So usually, one considers that communication or measurement range of sensors is limited by a respective radius. Generally, in coverage computation, it is supposed that these radius are constant in all directions for each sensor. Therefore the coverage capability of the space points could be exactly computed from the geometrical intersections of the disks produced by the radius and centered on each sensor. But this kind of computations could be expensive, consequently the authors of CGS has made the choice that sensors sample their measurement disk by elementary regions.

The following presents de£nitions of the terms that are used in this paper.

B. Vocabulary used

The speci£c terms used in this paper are the following:

- R_m: the sensor measurement radius is the maximum distance between a sensor and a point which is reachable for measurement by the sensor.
- D_m: the sensor measurement disk is a set of points reachable by a sensor for measurement.
- r: region, that is the surface unit used to sample the area surface.
- Op-sensor: operational sensor. Sensor that has enough energy to provide, during a period, the 1-coverage of its regions.
- N_S: the initial number of Op-sensors.
- $N_{Sa_{min}}$: the minimal number of active Op-sensors to provide the full k-coverage. This number could be a mean over an optimal sequence of minimal subsets.
- Cr: for a particular region r, it is the number of Op-sensors that can measure it.
- O_v: overdosing ($O_v > 1$)
  - Global de£nition: $N_S/N_{Sa_{min}}$.
  - Local de£nition (relative to a space point): Cr/k.
- R_c: the communication radius is the maximum distance between two sensors which enables them to communicate directly.
- Neighbor sensors: Op-sensors which can communicate directly.
- Common Region Sensors (CRS): Op-sensors whose measurement disk (D_m) intersects together.
- Neighbor and Common Region Sensors (NCRS): CRS which are also neighbor.

Generally it is required that, the measures taken by active sensors are forwarded by the other active sensors of the network toward some data sinks. To achieve this measurement data transmission, communication graph provided by the active sensors must be connected. As it has been shown in [8], this connectivity is automatically provided if $R_c \geq 2 \cdot R_m$. Furthermore when this inequality is veri£ed, we can deduce that $CRS = NCRS$. Thus, as many other works, with no lack of generality and to not mix k-coverage problems with connectivity problems, to simplify our discussion we assume this radius inequality is veri£ed.

C. The three phases of CGS

CGS works in a distributed and periodic manner. CGS updates its active Op-sensor subset at the beginning of each period. This update period is chronologically divided into three phases which have their own tasks and messages (see Figure 1):

1) Hello phase: Each Op-sensors sends an Hello message which indicates the sensor location and power state to their neighbors (potentials CRS).
2) STD phase : Each Op-sensor computes its Drowsiness Factor (Df) [6] and Shoot Time Delay (STD). Then, they transmit their STD by sending a STD message.
3) Awake phase: Each Op-sensor takes its sleeping decision and if necessary sends an Awake message.

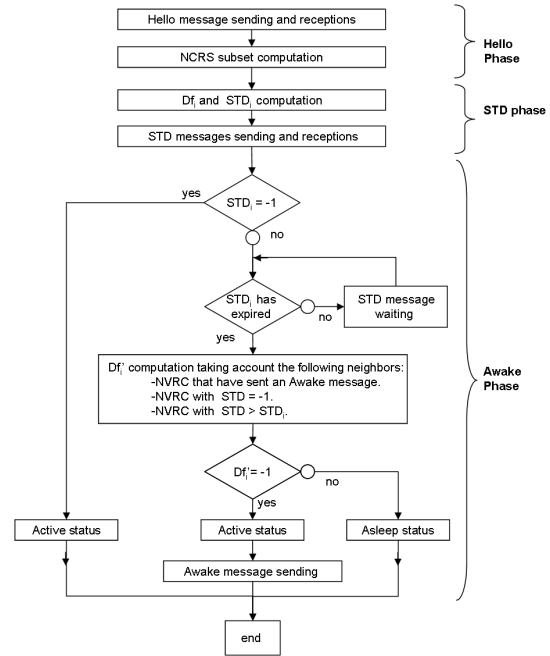More precisely the three CGS phases are described in the following.

Fig. 1. The tree CGS phases running on Op-sensor i.

1) Hello phase: During the Hello phase, Op-sensors receives Hello messages and updates their NCRS list. The sender sensor location found into each Hello message enables the receiving sensor to ascertain whether the sender is one of its NCRS or not. The Hello messages (as all other CGS control messages) are not forwarded by any receiving sensor, because, thanks to CGS, sensors just need to know their one-hop neighbors.

2) STD phase: After the time-out of the Hello phase, each Op-sensor computes its Df and STD values.

An Op-sensor with an E_s remaining energy level computes its Df as the following:

$$Df = \begin{cases} \frac{1}{E_s^\alpha} \sum_{\forall r \in D_m} \phi_r & if \ \phi_r > 0, \ \forall r \\ -1 \ else. \end{cases} \quad (1)$$

α is a positive constant ($\alpha = 2$ typically), and ϕ_r is defined as the following :

$$\phi_r = \begin{cases} \frac{1}{C_r - k} & if \ C_r > k \\ -1 \ else. \end{cases} \quad (2)$$

For a region r, C_r is determined by simple geometric and distance computation thanks to the sensor knowledge of its neighbor locations and its own location.

Once the Df value has been computed, each Op-sensor computes its STD and sends it to its neighbor sensors using an STD message. STD is the activity priority metrics used by CGS. If a sensor has a lower STD value than its neighbor sensors then this sensor has a lower probability to be active during this period.

Typically STD value is computed as the following:

$$STD = \begin{cases} (1 - Df) \cdot T_2 & if \ Df \neq -1 \\ -1 \ else. \end{cases} \quad (3)$$

with T_2 the known duration of the STD phase.

Remark: if a sensor has one of its regions with $C_r \leq k$ then the Df and STD values are equal to -1. Theses negative values compel the sensor to be active during the current period. The reception of a STD message which contains a negative value is sufficient for the receiving sensors to know that the sending sensor will stay active. This early knowledge sharing enables these sensors to reduce their communication cost: the sender of the STD message (with negative STD value) stops here is decision process (it stays active) and does not send any other message (Awake message). Moreover that knowledge makes the decision of the Awake phase easier for all the receiving sensors.

3) Awake phase: Sensors must decide if they stay active or not when their STD timer have expired.

A sensor will be allowed to go to sleep only if the full k-coverage of its regions may be provided by the activation of some of its neighbor sensors. That is sensors which have sent to the local sensor:

- a STD message with a STD value equal to -1, or
- an Awake Message, or
- a STD message with a higher STD value than the STD value.

One way to determine the sensor future state is to recompute the $Df\prime$ value (see Figure 1) by only taking into account

the previous sensors. If $Df\prime$ value is different from -1 then the sensor can go to sleep, otherwise it must stay active for this period and send an Awake message just after its STD expiration.

D. About CGS complexity

We evaluate the complexity of the CGS algorithm in number of messages and in number of operations. According to our definitions and our general assumptions (in particular if $R_c \approx 2 \cdot R_m$ and, if all sensors are operational and, if the sensors are evenly spread) each sensor may have $\frac{\pi \cdot R_c^2}{\pi \cdot R_m^2} \cdot C_r = 4 \cdot C_r$ neighbors.

As we have seen previously, each Op-sensor may send 2 or 3 messages every period. This overload is pretty low if the CGS period (typically several seconds) is rather long versus the duration of a local transmission (typically some microseconds or less). It is not useful to evaluate the overall number of CGS messages exchanged over all the network every period because CGS doesn't require sensors to forward CGS messages. However wireless network is a multiple access network. Thus the network load around (but local to) one sensor is of interest. Obviously there is between $2 \cdot 4 \cdot C_r$ and $3 \cdot 4 \cdot C_r$ messages sent per period because there are $4 \cdot C_r$ neighboring sensors around each local sensor.

To evaluate the operation complexity of CGS algorithm, we assume that test or addition costs are negligible compared to multiplication or division costs, and that the most demanding part of the algorithm is the Df computation because it requires a sensor to compute the distance between each of its region and all of its neighbors.

To compute its Df a sensor must compute the C_r of each region around itself. Thus if each sensor shares its D_m disk into N_R regions, then the Df computation would require $O(N_R)$ operations. Each C_r computation requires a computation for each neighbor (some geometric distance calculations between the neighbors and the region). As a sensor has C_r neighbors, consequently, every period, the Df computation would require $O(C_r \cdot N_R)$ operations. We can notice that this complexity decreases with the communication radius (whose C_r depends) and the measurement radius and accuracy level (whose region size and N_R depends). This behavior is in favor of CGS algorithm.

E. Performances Criteria

We use three metrics, they are in accordance with the two algorithm objectives.

1) Coverage quality for a period: The coverage quality Q is an approximation of the ratio of k-covered surface, it supposes regions of equal size and weight:

$$Q = \frac{1}{N_R} \cdot \sum_{i=k}^{N_S} N_{R_i} \quad (4)$$

N_R is the total number of regions and N_{R_i} is the number of i-covered regions. As $N_R = \sum_{i=0}^{N_S} N_{R_i}$, we have $0 \leq Q \leq 1$.

2) Global performance: Global performance $P(Q)$ is the duration during which a minimal quality of Q is provided by the k-coverage algorithm. Typically for CGS, we want to evaluate $P(1)$, i.e. the duration during which the k-coverage is assured over all the regions of the area under observation.

Normalization: $P_n(Q) = \frac{P(Q)}{P_{max}(Q)}$

We know that $P(1)$ will be limited by the selection of non optimal subsets of active Op-sensors. So a good performance criteria on each period is the coverage cost.

3) Coverage cost: The coverage cost C is simply the number of active Op-sensors during a period for a certain k-coverage quality. The coverage cost must be compared with the minimal mean cost coverage Nsa_{min} of sensor subsets that provide a maximal global performance $P_{max}(Q)$. We can compute the relative coverage cost $C_{rel} = \frac{C}{Nsa_{min}}$.

Nsa_{min} and $P_{max}(Q)$ computation can be very complex, because they are optimal values. Exhaustive searching methods are too long for large network. That's why we choose an original and very simple topology which assures the existence of an analytical expression for Nsa_{min} and $P_{max}(Q)$.

III. SIMPLE RING TOPOLOGY TO EVALUATE CGS

A. Basics

We choose to place uniformly the sensors and the regions on a ring (see Figure 2). This topology is very interesting because its symmetry enables us to £nd easily the minimal disjoint subsets of sensors (with $Q = 1$). Each subset can be deduced from the initial one by the rotation which permits to go from a sensor on the ring to one of its consecutive sensor. So all subsets have the same cost which is in fact Nsa_{min}. One simple condition is $O_v \in \mathbb{N}^*$, this allows to make $\lfloor O_v \rfloor$ disjoint minimal subsets of sensors. So in this topology, measurement and communication radius (R_m and R_c) can be easily computed from N_S, Nsa_{min} and k values, and we are assured that these chosen parameters can provide the maximum k-coverage quality ($Q = 1$).
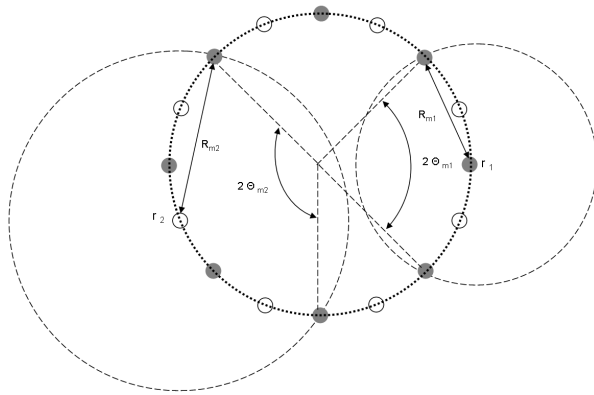

Fig. 2. Measurement Radius (R_m) computation in a ring topology with $k = 3$, $r = 1$, $N_S = 16$, $O_v = 2$, and $N_R = 320$

B. Minimal measurement and communication radius

Let's us assume the following:

- θ_S: the angle between two consecutive sensors on the ring ($\theta_s = 2\Pi/N_S$).
- θ_r: the angle between two consecutive regions on the ring ($\theta_r = 2\Pi/N_R$).
- θ_{R_m}: the equivalent angle to R_m.
- θ_{R_c}: the equivalent angle to R_c

To determine the minimal R_m and R_c radius, we must take into account regions which are located £rst, exactly between two inactive sensors (see r_1 in Figure 2) second exactly between two active sensors (see r_2 in Figure 2). If k is even r_1 region (resp. r_2) needs $O_v \cdot k$ (resp. $O_v \cdot k + 1$) Op-sensors that can measure them and, respectively for odd values of k.

That's mean that it must have $O_v \cdot k + 1$ Op-sensors in the disk D_m of each Op-sensor. This is equivalent to an angle of $2 \cdot \theta_{R_m}$. So we have $O_v \cdot k + 1 = \frac{\theta_{R_m}}{\Pi} \cdot N_s$.

We can compute the minimal measurement radius:

$$R_m = 2 \cdot r \cdot \sin(\frac{\theta_{R_m}}{2}) = 2 \cdot r \cdot \sin(\frac{\Pi \cdot (O_v \cdot k + 1)}{2 \cdot N_s}) \quad (5)$$

For the minimal communication radius, we have to choose an equivalent angle radius such that $\theta_{R_c} \geq 2 \cdot \theta_{R_m}$ and so $R_c \geq 2 \cdot r \cdot \sin(\theta_{R_m})$. That means in this topology that initially an Op-sensor should have at least $2 \cdot O_v \cdot k$ neighboring sensors.

C. Maximum global performance computation

The simple ring topology enables us to analytically produce the maximum Global Performance $P_{max}(Q)$. $P_{max}(Q)$ may be reached by successive sensor minimal subsets where for each successive period $Q = 1$ and $C_{rel} = 1$.

Op-sensors need to send messages at each CGS period. These message sendings consume energy. So the shorter the period durations are, the more numerous the periods are. Consequently, all being equal, shortest periods consume more energy. That is why, our model has to take into account the number of periods, their duration and the energetic cost of message transmission.

So if we want to reduce the energy consumed by the message transmissions, we have to limit the number of periods and so increase their duration. The maximum full k-coverage duration is reached when each sensor has just to stay active during one period (we assume that initially all sensors have the same energy level). In this case the network we be able to provide $\lceil O_v \rceil$ periods of full k-coverage. We propose to consider this case as a reference for this paper.

However there is a drawback when very long period duration is considered. The k-coverage quality is reduced for the

remaining period time when any active sensor fails during the period.

The following formal analysis propose to determine the maximal global performance ($P_{max}(1)$) on the ring topology by taking into account these parameters:

- The initial energy level of each sensors.
- The wished number of active periods of each sensor (N).
- The maximal total number of periods ($N \cdot O_v$) with the full k-coverage ($Q = 1$). This number value is equal to $N \cdot O_v$ because there are O_v disjoint minimal subsets of active sensors.
- The energy spent in communications and measurements for each sensor.
- Each period duration (T_N).

Basically, we can say that $P_{max}(1) = N \cdot O_v \cdot T(N)$, so the main task is to determine $T(N)$.

Our hypothesizes are the following:

- CGS scheduling duration is negligible over T_N.
- Energy consumed during CGS scheduling is negligible: it is lower than the energy consumed by the communications and the measurements.
- The parameters of all the sensors have the same initial values (sensor locations excepted).

Global energy balance:

- E_0: initial energy level.
- E_R: remaining energy level.
- E_C: energy consumed in communications and computations.
- E_M: energy consumed in measurements.

$$E_R = E_O - (E_M + E_C) \qquad (6)$$

Energy balance of an Op-sensor for a period:

- $\frac{dE_M}{dt_a}$: energy consumed in measurements per time unit.
- e_{ca}: energy consumed in communications and computations during an active period.
- e_{cs}: energy consumed in communications and computations during a sleeping period.
- e_m: energy consumed to send a message to a maximum distance of R_c.

With our previous hypothesis we can say that:

- $e_{ca} \approx 3.e_m$ because an awake sensor has sent at most 3 messages (Hello, STD messages plus an Awake message when $Df \neq -1$).
- $e_{cs} \approx 2.e_m$ because an asleep sensor has sent 2 messages (Hello and STD messages).

After $N \cdot O_v$ periods, we have for each sensor the following energy balance:

- $E_M = \frac{dE_M}{dt_a} \cdot N \cdot T_N$
- $E_C = ((N-1) \cdot O_v + i) \cdot e_{cs} + N.e_{ca}$, where i is the rank of the subset during the last O_v periods ($1 \leq i \leq O_v$).

To compute $P_{max}(1)$, we have to take into account the case where sensors have consumed all their energy ($i = O_v$ and $E_R = 0$). In this case we can say the following:

$$E_0 = E_M + E_C = \frac{dE_M}{dt_a} \cdot N \cdot T_N + N \cdot (O_v.e_{cs} + e_{ca}) \quad (7)$$

Let's us assume the following ratios:

- $\alpha = \frac{E_C}{E_M}$. α enables us to compare the energy quantity consumed in communications and computations (which a lost energy) against the energy consumed in measurements (which is a well utilized energy).
- $\beta = \frac{e_{cs}}{e_{ca}}$ the ratio of the communication costs, the communications cost of the asleep state over the awake state. Typically it is equal to $2/3$.

We can say that: $E_0 = E_M + E_C = \frac{E_c}{(1/\alpha + 1)}$

Thus: $\alpha = \frac{1}{(E_0/E_C - 1)}$

And $E_C(N) = N \cdot e_{ca}(\beta.(O_v - 1) + 1) = N \cdot E_C(1)$

Then:

$$\alpha_N = \frac{1}{\frac{E_0}{E_C(N)} - 1} = \frac{1}{\frac{E_0}{N \cdot E_C(1)} - 1}, \text{ with } E_C(1) = \frac{E_0}{1/\alpha_1 + 1}$$

Consequently:

$$\alpha_N = \frac{\alpha_1 \cdot N}{1 - \alpha_1 \cdot (N - 1)} \qquad (8)$$

with $1 \leq N < E_0/E_C(1)$

We can now compute the period duration T_N. An interesting α_1 value is 0. That means all energy is consumed in measurements and that $N \cdot T_N$ will be constant ($\forall N \in \mathbb{N}^*$). We name this constant the reference time (T_{ref}).

Also, if $\alpha_1 = 0$ then $\alpha_N = 0 \; \forall N \in \mathbb{N}^*$ (see equation 8). In this case we can say that: $P_{max}(1) = O_v \cdot T_{ref}$.

Since:

$$E_0 = E_M \cdot (\alpha_N + 1) = \frac{dE_M}{dt_a} \cdot N \cdot T_N \cdot (\alpha_N + 1) \qquad (9)$$

If $N = 1$ and $\alpha_1 = 0$ then $E_0 = E_M = \frac{dE_M}{dt_a} \cdot T_{ref}$. If we assume that $T_{ref} = 1$ (for normalization) we can say that:

$$\frac{dE_M}{dt_a} = E_0 \qquad (10)$$

Thus: $E_M = \frac{dE_M}{dt_a} \cdot N \cdot T_N = \frac{E_0}{1 + \alpha_N}$

Then:

$$T_N = \frac{1}{N.(1 + \alpha_N)} \qquad (11)$$

Consequently:

$$P_{max}(1) = O_v \cdot N \cdot T_N = \frac{O_v}{1 + \alpha_N} \qquad (12)$$

Because $0 \leqslant \alpha_1 \leqslant \alpha_N$ and according to our references,

we can say that: $\frac{P(1)}{O_v} \le \frac{P_{max}(1)}{O_v} \le 1$.

And for $P(1)$ normalization we can propose the following:

$$P_n(1) = \frac{P(1)}{P_{max}(1)} = \frac{P(1)}{O_v} \cdot (1 + \alpha_N).$$

IV. CGS PERFORMANCES

We have performed the simulations with various O_v, N_S, $N_{Sa_{\min}}$ and k values but all simulations have the same number of periods ($N = 30$).

A. Coverage quality

In this section we have selected the results of some representative simulations with $N_s = 99$, $O_v = 3$, $k = 4$. We introduced some message loosing to evaluate the CGS robustness. We compare the coverage quality provided by CGS and the Random Scheduling (RS) algorithm. The advantage of the RS algorithm is, it does not need to send any message. Thus message loss has no impact on RS algorithm.

We have performed the simulations with various loss probability p_l, and various sleeping probability p_s. A good value for p_s should be $p_s = 1 - 1/O_v \approx 0.667$.

In Figure 3, the coverage quality curves of CGS and Random Scheduling evolve similarly along the time because in the two cases coverage quality can be chronologically divided in two behavior phases:

1) £rst a constant coverage quality is assured.
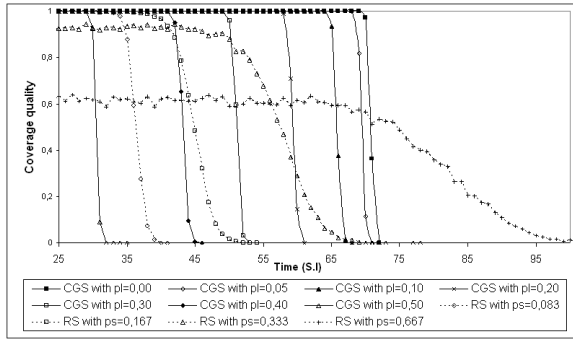2) then the coverage quality is decreasing to zero (the number of Op-sensors is deceasing).

Fig. 3. CGS coverage quality as a function of messages loss probability (p_l) vs Random Scheduling coverage quality as a function of sleeping probability (p_s) with $N_s = 99$, $O_v = 3$ and $k = 4$.

When the message loosing probability increases (from 0% to 50%) the length of the £rst behavior phase of CGS decreases from 70 to 30 (see $P(1)$ in table I) but the Q level is constant ($Q \approx 1$). That's why we consider the duration of the CGS £rst behavior phase as $P(1)$. The length of CGS second behavior phase (which corresponds to $P(0 < Q < 1)$)

p_l	0.00	0.05	0.1	0.2	0.30	0.40	0.50
$P(Q = 1)$	70	69	65	58	50	41	30
$P(Q > 0)$	73	72	69	62	54	47	33
$P(0 < Q < 1)$	3	3	4	4	4	6	3
$\overline{Q}$	0.99	0.98	0.97	0.98	0.97	0.95	0.97
σ_Q	0.07	0.11	0.14	0.11	0.14	0.20	0.16

TABLE I
CGS COVERAGE QUALITY Q AND GLOBAL PERFORMANCES AS A FUNCTION OF MESSAGES LOSS PROBABILITY WITH $N = 30$, $Ov = 3$, AND $k = 4$.

p_s	0.083	0.167	0.333	0.667
$P(Q = 1)$	27	0	0	0
$P(Q > 0)$	42	55	72	109
$P(0 < Q_s < 1)$	15	55	72	109
$\overline{Q}$	0.90	0.99	0.74	0.47
σ_Q	0.28	0.005	0.33	0.23

TABLE II
RANDOM SCHEDULING GLOBAL PERFORMANCES ($P(Q)$) AS A FUNCTION OF SLEEPING PROBABILITY p_s VALUES WITH $N = 30$, $Ov = 3$, AND $k = 4$.

is shorter than the length of CGS £rst behavior phase (the ratio is $6/47$ for $p_l = 0.40$). On all simulations, during £rst and second behavior phases, CGS provided a mean k-coverage quality ($\overline{Q}$) between 0.95 and 0.99 with standard deviation (σ_Q) between 0.07 and 0.20.

RS behaves differently. Indeed when the sensors sleeping probability increases (from 0.083 to 0.667) the two behavior phases increase. Total durations for $P(Q > 0)$ increase from 42 to 109. (see Figure 3) They could be higher than CGS ones. However, mean coverage quality goes down to 0.47. The only case where RS provides a full k-coverage (but for a limited time interval) is for the lowest p_s value 0.083. One could expect from RS a better behavior: full k-coverage may be reached as soon as the p_s value is lower than $2/3$.

These results show that CGS has been £rst designed to ensure full k-coverage ($Q = 1$) as in [7]. These results also show CGS robustness because when messages are lost, CGS manages to maintain $Q = 1$. Message losses disturb the sensors because they could not accurately build the NCRS list and take appropriate scheduling decisions. So the selection of the active sensor subset is disturbed by the message loss and that increases the coverage cost.

Now we know that CGS provides, £rst of all, full k-coverage, thus the evaluation of its coverage cost (when $Q = 1$) is very signi£cant.

B. Coverage cost

In this paper we only give the results of simulations with 100 sensors. Others simulations have been made with 300 and 999 sensors, they con£rm the stability of our observations.

We propose to study the CGS cost coverage within the periods where:
- $Q = 1$.

- All sensors are operational.
- There are no message loss and no synchronization problem.
- There are no error in sensor locations.
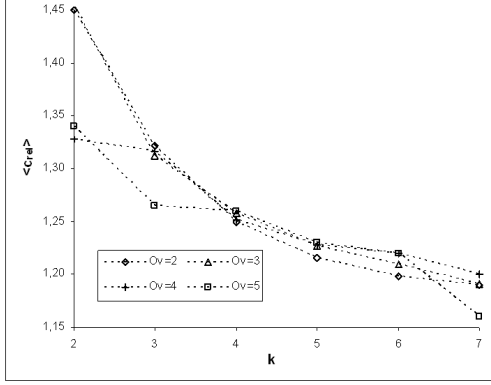- All sensors have Df and STD values which are different to -1 (the last periods are avoided).

Fig. 4. CGS relative coverage cost ($< c_{rel} >$) as a function of overdosing (O_v) and k.

As we can see on Figure 4, CGS does not have the same mean coverage costs for different topologies. The relative mean coverage cost decreases from 1.45 to 1.16 when k increases from 2 to 7 and with O_v values between 2 and 5.

One can note that for $k > 3$, k seems to be the most discriminative parameter for the mean cost coverage. Indeed mean coverage cost values are very closed when $k > 3$ because the differences between the values are lower than 0.05. The most homogeneous results are obtained for $k = 4$ (the mean coverage cost is 1.25 and the standard derivation is 0.006). That's why results shown on previous and next subsections are based on a $k = 4$ value.

Figure 4 shows us that when k value increases from 2 to 7 mean relative coverage cost decreases from 1.40 to 1.20. We explain this by the fact that the number of neighbor sensors increases as k and O_v increase (number of neighbors $\approx 2 \cdot O_v \cdot k$). So the larger number of neighbors a sensor has, the best the mean cost coverage is.

k being constant, the higher O_v is, the lower mean cost coverage is. One may think that cost coverage simply depends on the accurate knowledge of the topology. That could be described by the ratio C_r/N_S. But simulations with $N_S = 300$ and $N_S = 900$ with the same k and O_v values (and consequently with the same C_r value) tell us that N_S do not affect the mean cost coverage.

So let's us compare the global analytical and experimental performances.

C. Comparison of analytical and experimental global performances

The experimental relative cost coverage which is superior or equal to 1 when $Q = 1$ reduces the number of periods $N\prime_p$ (with $N\prime_p \leq O_v \cdot N$). It is also reduced the experimental global performance $P\prime(1)$ if we consider the same period duration T_N.

So $P\prime(1) = N\prime_p \cdot T_N$

and $P\prime_n(1) = \frac{N\prime_p}{O_v \cdot N} = \frac{N\prime_p \cdot T_N}{O_v} \cdot (1 + \alpha_N)$

We simulated CGS with various α_1 values. Table III shows us that $\alpha(N = 30)$ increases from 0 to 6.92 for α_1 values between 0 and 0.03 because of the increasing of the communication cost. Maximum period duration $T(N)$ decreases from $0.033 = 1/N$ to 0.004. Mean relative coverage is constant and close to 1.25.

As expected, $P_{max}(1)$ and $P(1)$ decrease when α_1 increases (see Figure 5). Consequently, for $\alpha_1 = 0$, $C_{rel} = 1.25$, we have $P_n(1) \approx 0.77$, which is closed to the expected value of $\frac{P_{max}(1)}{O_v \cdot C_{rel}} = 0.80$.

CGS is advised for high α_1 values. Indeed the higher α_1 value is (thus the higher communication costs are), the better $P_n(1)$ is (because coverage costs are less determinant in $P(1)$ than communication costs). $P_n(1)$ increases from 0.77 to 0.92 when α_1 increases from 0 to 0.03. We can say that the higher the communication costs are, the more interesting the use of CGS seems to be.

α_1	0.000	0.001	0.005	0.010	0.020	0.030
$\alpha(30)$	0.00	0.03	0.18	0.42	1.42	6.92
$T(30)$	0.033	0.032	0.028	0.023	0.013	0.004
$< C_{rel} >$	1.25	1.26	1.26	1.26	1.26	1.25

TABLE III
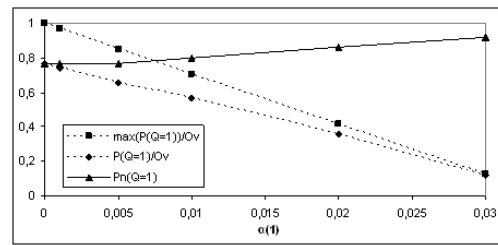CGS PARAMETERS EVOLUTION AS A FUNCTION OF α_1 VALUE WITH $N = 30$, $O_v = 3$, AND $k = 4$.

Fig. 5. CGS global relative and absolute performances with $N = 30$, $O_v = 3$ and $N_S = 99$.

V. CONCLUSION

Simulations results are coherent with our formal analysis. They have been applied on a ring topology and they give rise to the best use cases of CGS (with the higher P_n values).

Our results from coverage quality study prove that CGS was £rst designed to guaranty the k-coverage. They prove also CGS robustness when there are message loosing, because the highest coverage quality is maintained but for a shorter duration. So CGS algorithm guarantees (when possible) to £nd in a distributed manner a good approximation of the optimal solution of the k-coverage problem. The found solutions are some of the best whatever the knowledge of the sensors about their neighbor decisions or neighbor existences.

Moreover the coverage cost study reveals that CGS scheduling ef£ciency (with a full k-coverage) increases as k and the overdosing (O_v) increase. This is due to the increasing of the number of one-hop neighbors and the increasing of number of sensors that can measure a region (respectively $2 \cdot k \cdot O_v$ and $k \cdot O_v$ in a ring topology).

Then we are convinced that the more the sensors have neighbors (and the more they know their existences or their decisions) the less the relative cost coverage is; And since k-coverage is provided, the more the relative global performance is.

Global relative performance put in evidence that the more the energy spent in communications is rather in measure ones for a sensor, the more the CGS relative global performances are. We explain this by the fact that in this case the relative coverage cost (typically closed to 1.25 in our results) is less signi£cant on relative global performances.

That's the reasons why we advise to use CGS when the £rst objective is the full k-coverage and in a topology with high k and overdosing values and/or with high communication costs.

VI. OPEN PROBLEMS

Additional performance analysis could be done by taking into account the imprecisions of the sensor locations and the potential lack of synchronization between sensors.

An interesting but future study is whether it should be better to optimize the number of sensors in the network or the sensor radius (communication and measurement radius). Let give a short glance about the tradeoff in this optimization problem. In one hand, the performance of the algorithm is function of the overdosing (that is good). But, the overdosing is related to the number of sensors in the network. And the network energy is the sum of the energy of all the sensors in the network. Thus the network energy increases as the overdosing increases (which is bad from an ef£ciency point of view). In the other hand, if the measurement radius increases more regions are covered by one sensor (which is good). But if the communication radius increases the collision rate and (sometimes) the energy consumed by the communication will increase. Thus communications will become less ef£cient (which is bad).

A such study can be easily done thanks to our ring topology. In this topology measurement and communication radius can be easily deduced from the k-coverage and the overdosing.

Nevertheless to take these points into consideration, our energy model have to be enhanced. For instance the energy consumed by a message transmission could be function of the distance between the sender and destinations, or in another way function of the overdosing.

The increasing of measurement and communication radius could be considered by increasing the cost due to communications (α_1 and so α_N). For instance if we choose a physical model which compute the energy consumed when a message is sent to a certain distance (e_m).

REFERENCES

[1] C.F. HUANG and Y.C. TSENG, *The Coverage Problem in a Wireless Sensor Network*, ACM Int. Conf. on Wireless Sensor Networks and applications, September 19, 2003, San Diego, California, USA.
[2] S. KUMAR, T.H. LAI and J. BALOGH, *On K-Coverage in a Mostly Sleeping Sensor Network*, Annual Int. Conf. on Mobile Computing and Networking (MobiCom'04), Sept. 26-Oct. 1,2004, Philadelphia, Pennsylvania, USA.
[3] H. GUPTA, Z. ZHOU, S.R. DAS and Q. GU, *Connected Sensor Cover: Self-Organisation of Sensor Networks for Ef£cient Query Execution*, ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc), Annapolis, MD, 2003.
[4] X.Y. LI, P.J. WAN O. FRIEDER, *Coverage in Wireless Ad-hoc Sensor Networks*, Department. of Computer Sciences, Illinois Institute of Technology. IEEE Communications Magazine, 2002.
[5] Z. ZHOU and S. DAS and H. GUPTA, *Connected K-Coverage Problem in Sensor Networks*, Proceedings 13th International Conference on Computer Communications and Networks (ICCCN 2004), 2004.
[6] G. SIMON and M. MOLNAR and L. GÖNCZY and B.COUSIN, *Dependable k-Coverage Algoritms for Sensor Networks*, Instrumentation and Measurement Technology Conference (IMTC'07), 2007.
[7] G. SIMON and M. MOLNAR and L. GÖNCZY and B. COUSIN, *Robust k-coverage algorithms for sensor networks*, IEEE Transactions on Instrumentation and Measurement, August 2008.
[8] X. WANG and G. XING and Y. ZHANG and C. LU and R. textscPless and C. GILL, *Integrated Coverage and Connectivity Con£guration in Wireless Sensor Networks*, ACM SenSys conference, November 5-7, 2003, Los Angeles, California, USA.
[9] I.F. AKYILDIZ and W. SU and Y. SANKARASUBRAMANIAM and E. CAYIRCI, *A Survey on Sensor Networks*, IEEE Communications Magazine, August, 2002.