Sensor Activation and Radius Adaptation (SARA) in Heterogeneous Sensor Networks

NOVELLA BARTOLINI, TIZIANA CALAMONERI
"Sapienza" University of Rome, Italy
TOM LA PORTA
Pennsylvania State University, USA
and
CHIARA PETRIOLI, SIMONE SILVESTRI
"Sapienza" University of Rome, Italy

In order to prolong the lifetime of a wireless sensor network (WSN) devoted to monitoring an area of interest, a useful means is to exploit network redundancy activating only the sensors that are strictly necessary for coverage and making them work with the minimum necessary sensing radius.

We introduce the first algorithm which reduces coverage redundancy by means of Sensor Activation and sensing Radius Adaptation (`SARA`) in a general applicative scenario with two classes of devices: sensors that can adapt their sensing range (adjustable sensors) and sensors that cannot (fixed sensors). In particular, `SARA` activates only a subset of all the available sensors and reduces the sensing range of the adjustable sensors that have been activated. In doing so, `SARA` also takes account of possible heterogeneous coverage capabilities of sensors belonging to the same class. It specifically addresses device heterogeneity by modeling the coverage problem in the Laguerre geometry through Voronoi-Laguerre diagrams.

`SARA` executes quickly with guaranteed termination, producing a very efficient network configuration. By means of extensive simulations we show that `SARA` obtains impressive improvements with respect to previous solutions, ensuring wider coverage with longer network lifetime in all the considered scenarios.

Categories and Subject Descriptors: C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Distributed networks; network topology*

General Terms: Algorithms, Design, Performance

Additional Key Words and Phrases: Area coverage, wireless sensor networks, heterogeneous devices, variable radii

Author's address: Novella Bartolini, Tiziana Calamoneri, Chiara Petrioli and Simone Silvestri, Department of Computer Science, "Sapienza" University of Rome, Via Salaria 113, 198 Rome, Italy . E-mail: {novella, calamo, petrioli, silvestris}@di.uniroma1.it
Tom La Porta, Pennsylvania State University, University Park, PA, USA. E-mail:tlp@cse.psu.edu
This work has been partially supported by the Italian Ministry of Education and University PRIN project COGENT (COmputational and GamE-theoretic aspects of uncoordinated NeTworks) and by the EC FP7 project GENESI (Green sEnsor NEtworks for Structural monItoring.)

1. INTRODUCTION

As large collections of networked, inexpensive devices, Wireless Sensor Networks (WSNs) are the technology of choice for applications requiring seamless and pervasive coverage of geographic areas, buildings and public or private spaces and structures. Critical applications such as access control and intrusion/hazard detection as well as less critical tasks of which wildlife monitoring and precision agriculture are typical examples, are best served by the infrastructure-less and unobtrusive nature of WSNs.

Since sensor nodes typically have limited battery power, meeting coverage requirements with minimal energy expenditure is a primary issue. For years this problem has been tackled by designing protocol stacks that are energy efficient, implicitly assuming that the culprit of most of the energy consumption of a node is the communication circuitry. As a consequence, solutions that enhance network performance (lifetime, capacity, etc.) have been proposed that are based on methods that reduce communication costs: Data fusion and filtering techniques (for limiting the number of transmissions) [Nakamura et al. 2007], new and advanced forms of energy provisioning [Sharma et al. 2010; Moser et al. 2010; Kansal et al. 2007], clever exploitation of the mobility of network components [Basagni et al. 2008] as well as optimized protocol design [Yick et al. 2008]. However, the level of improvement that energy-efficient techniques for communication can produce is starting to plateau because of the inevitable trade-offs that they impose (e.g., energy conservation versus latency). At the same time, the sensing devices mounted on the wireless node have become more numerous and more sophisticated. Along with the cheap sensors, e.g., those for temperature and humidity, it is now common to endow even small nodes with cameras and active sensors such as radars and sonars, which demand non-negligible energy from the node. Therefore, for providing critical enhancement to network performance, it is no longer possible to focus only on reducing communication costs. Careful consideration must be also given to the sensory component of the node. We also note that, unlike"on-off" sensors, like those for temperature, light, and humidity, more sophisticated sensors consume energy depending on their sensing range. Therefore, similar to communication power control, *sensing coverage control* becomes an important element in the overall WSN performance optimization process. In particular, *sensor activation and radius adaptation*, the ability of selecting which sensor to activate[1] and to what level of coverage, are necessary new ingredients for the design of durable and reliable WSNs.

In this paper we present a new solution for the joint problem of dynamically scheduling the activation of different subsets of sensor nodes and of tuning their sensing radii (if their technology allows) for prolonging the network lifetime while ensuring the maximum achievable coverage extension of the given Area of Interest (AoI).

Sensor activation as a research area has received considerable attention in the recent past. In particular, two selective activation algorithms have been proposed that

[1] With *sensor activation* we indicate the turning on of the sensing and communication units of a node. When this happens, the sensor is *awake*. A sensor goes to *sleep* by turning off (or by switching to low power mode) both its sensing and communication units.

have been shown to outperform other solutions for the problem: The Distributed Lifetime Maximization (DLM) scheme [Kasbekar et al. 2009], and the Variable Radii Connected Sensor Cover (VRCSC) [Zou et al. 2009].

In this paper we propose an algorithm called SARA, standing for *Sensor Activation and Radius Adaptation* that, to the best of the authors' knowledge, is the first algorithm working in the general scenario of heterogeneous networks, comprising both fixed and adjustable sensors. Our algorithm follows an original approach to solve the coverage problem, as it makes use of the Voronoi diagrams in the Laguerre geometry to determine the coverage responsibility of each node.

SARA achieves the following desirable properties (theoretically and experimentally proven in the following).

—It ensures maximum sensing coverage at all times, i.e., awake nodes are able to cover the same area that would be covered if all nodes that are still operational were activated with their maximum transmission range.

—It accommodates WSNs composed of heterogeneous nodes, endowed with adjustable or fixed sensing radius.

—It is Pareto optimal, unlike DLM and VRCSC. (This property constitutes a necessary requirement for a sensor activation and radius adaptation policy to be optimal.)

—It is robust with respect to different definitions of coverage requirements and network lifetime.

The performance of SARA has been evaluated by means of simulation experiments on WSNs with heterogeneous nodes. The results of our experiments show that SARA is able to quickly configure the network in a way that ensures low energy consumption and long lifetime. We also conducted a comparative performance evaluation of SARA with DLM and VRCSC, which revealed the superiority of SARA in terms of coverage extension and network lifetime in a wide range of operative settings, including the ones for which those previous solutions were specifically designed.

The paper is organized as follows. Section 2 introduces the problem of radius adaptation and sensor activation. Section 3 motivates the use of the Voronoi-Laguerre measure to address device heterogeneity and provides the notions of computational geometry needed to fully understand the proposed solution. In Sections 4 and 5 we describe SARA and prove its Pareto optimality, convergence and termination. Section 6 briefly describes the algorithms selected as benchmarks: DLM and VRCSC. A thorough performance evaluation of SARA is then provided in Section 7. Finally, Section 8 surveys the literature on related topics, while Section 9 concludes the paper.

2. PROBLEM FORMULATION

In this paper we consider heterogeneous WSNs, where the nodes are endowed with several kinds of sensing technologies. In particular, we focus on the use of two classes of sensors, namely, those with adjustable sensing radius and those with fixed radius. The capability to adjust the sensing range is typical of devices based on active sensing technologies, such as those equipped with radars and sonars. The power consumption of this kind of sensor depends on the extent of the sensing radius. For this type of sensors setting the sensing range to the minimum necessary for coverage decreases energy consumption. Although not all commercial active devices allow sensing radius adaptation, some sensors with adjustable sensing radii are already commercially available [OSIRIS photoelectric sensors 2010; Kompis and Aliwell 2010]. By contrast, for sensors based on passive sensing technologies (e.g., those equipped with piezoelectric sensors or thermometers) the monitoring activity typically consists in taking single point measures. For these devices the sensing radius is typically fixed as it is considered the range in which the value of the sensed property can be approximated with the point measure with limited error. An exception is the case of low power CMOS cameras, based on a passive sensing approach, where the depth of field can be adjusted to guarantee a given quality of monitoring at certain distances.

We consider a set $\mathcal{S} = \mathcal{S}_{\texttt{adjustable}} \cup \mathcal{S}_{\texttt{fixed}}$ of $|\mathcal{S}| = N$ sensors, where $\mathcal{S}_{\texttt{adjustable}}$ contains the nodes with adjustable sensing radius (hereby shortly called *adjustable sensors*) and $\mathcal{S}_{\texttt{fixed}}$ those with a fixed radius (shortly called *fixed sensors*) . If a node s_i belongs to the set $\mathcal{S}_{\texttt{adjustable}}$ its sensing radius r_i can be set to any value from 0 to $r_i^{\texttt{max}}$. For a node $s_j \in \mathcal{S}_{\texttt{fixed}}$ the sensing radius r_j is either 0, meaning that the sensing unit is sleeping, or $r_j^{\texttt{fixed}}$, when the sensing unit is awake. The sensors of the two sets can also have heterogeneous transmission radii $r_i^{\texttt{tx}}$, $i = 1, \dots, N$. We assume that the transmission radii are such that any two sensors with intersecting or tangential sensing circles are connected to each other. Therefore, complete coverage implies also that the WSN is connected, and no sensor should be kept awake if it is not necessary for coverage.

An exact model of the relationship between the energy consumed by a node for sensing and the extent of its sensing radius cannot be given as it is dependent on the sensing technology and electronic circuitry for detection. For the purpose of our work, we refer to a general approximate model also used in [Pattem et al. 2003; Zou et al. 2009] according to which if sensor s_i has sensing radius r_i the energy consumption per time unit is given by

$$E_{\texttt{sensing}}(r_i) = a \cdot r_i^c + b. \tag{1}$$

The parameters a and b are device specific constants. The parameter c is related to the sensing technology in use and typically varies in the range $[2, 4]$ in case of sensors adopting an active sensing technology.

The energy consumption due to communications is also dependent on the specific type of device being considered. It is typically an increasing function of the transmission radius, which takes into account all the energy consuming activities related to radio communications, namely transmissions, receptions and idle listening to the radio channel. In this paper we consider the energy cost model of Telos nodes [Polastre et al. 2005].

As usually done in the literature, we also assume that the AoI is a convex region.

The problem addressed in this paper is the following: *Given a WSN each sensor $s_i \in S$ has to decide whether to stay awake or not at any given time and, if awake, how to set its sensing radius r_i at that time. The objective is guaranteeing maximum achievable sensing coverage while prolonging the network lifetime as much as possible.*

Here we define the network lifetime as the time during which the network is able to guarantee a coverage extension higher than a given percentage p of the AoI, while maximizing coverage. For instance, if $p = 100\%$ the network lifetime is the time at which the first coverage hole appears. If $p = x\%$ the network lifetime is the first time at which less than $x\%$ of the AoI is covered [2].

3. PRELIMINARIES ON VORONOI LAGUERRE DIAGRAMS AND ON THEIR USE TO DETERMINE AND REDUCE COVERAGE REDUNDANCY

Prior works on sensor networks very often rely on the use of Voronoi diagrams to model coverage, such as in [Wang et al. 2006] for mobile sensors, in [Ammari and Das 2008] for energy aware routing, or in [Zou et al. 2009] for selective activation. Voronoi diagrams can be used to model the coverage problem only in the case of sensors endowed with equal sensing radii as discussed in [Bartolini et al. 2009]. In order to address the problem of coverage in the presence of heterogeneous devices, namely devices with different sensing ranges and different capability to adapt their setting, we introduce the notion of Voronoi diagrams in *Laguerre geometry.* We also discuss how these diagrams can be exploited to decrease coverage redundancy (and thus the energy consumption due to sensing) while preserving network coverage and connectivity.

In a Voronoi diagram, we call *Vor line* the axis generated by two sensors which is equidistant from them and perpendicular to their connecting segment. This line divides the plane into two halves. In the case of sensors with the same sensing radius the Vor line properly delimits the responsibility regions of the two sensors as it is the symmetry axis between the two. If the sensors have heterogeneous radii, the Vor line may not determine the responsibility region correctly, as depicted in Figure 1. Indeed, according to a Voronoi-based partition of coverage responsibilities, the sensor positioned in C_1 has the responsibility to sense anything to the left of the Vor line, and the sensor positioned in C_2 should sense anything to the right. In particular, the grey areas in the figure would incorrectly be assigned to the sensor in C_1, whereas they are covered only by the sensor in C_2. The line which correctly delimits the responsibility regions of the two sensors is the one that is equidistant from C_1 and C_2 in Laguerre geometry. In Figure 1 this line is called *VorLag.*

Formally, given a circle $\mathscr{C}$ with center $\mathrm{C} = (x_\mathrm{C}, y_\mathrm{C})$ and radius r_C, and a point $\mathrm{P} = (x_\mathrm{P}, y_\mathrm{P})$ in the plane $\Re^2$, the Laguerre distance $d_L(\mathscr{C}, \mathrm{P})$ between $\mathscr{C}$ and P is defined as follows:

$$d_L^2(\mathscr{C}, \mathrm{P}) = d_E^2(\mathrm{C}, \mathrm{P}) - r_\mathrm{C}^2, \tag{2}$$

where $d_E(\mathrm{C}, \mathrm{P})$ is the Euclidean distance between the points C and P. In Laguerre geometry, given two circles with distinct centers and possibly different radii, the locus of the points equally distant from them is a line, called *VorLag line*, that is perpendicular to the segment connecting the centers. If the two circles intersect each other, their VorLag line crosses their intersection points, as in Figure 1 (a) [Imai et al. 1985].

Given N circles $\mathscr{C}_i$ with centers $\mathrm{C}_i = (x_i, y_i)$ and radii r_i, $i = 1, \ldots, N$, the

[2] Definitions of lifetime based on the percentage of alive nodes [Bough and Santi 2002] can be adopted as well. Although more commonly used in the literature, these different notions of lifetime are less suitable than our when the applicative task is coverage of an AoI.

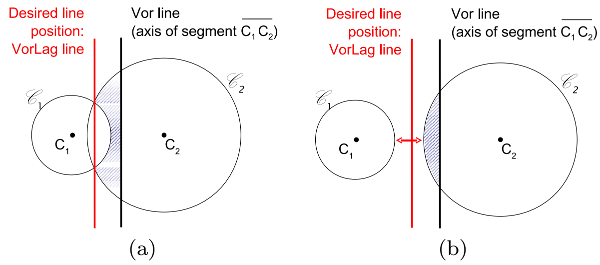


Fig. 1. Different positions of the line equidistant from C_1 and C_2 according to the Euclidean (Vor) and to the Laguerre (VorLag) distance in the case of intersecting (a) and non-intersecting (b) circles.

Voronoi-Laguerre polygons $V(\mathscr{C}_i)$ for the circles $\mathscr{C}_i$ are defined as

$$V(\mathscr{C}_i) = \{\mathrm{P} \in \Re^2 | d_L^2(\mathscr{C}_i, \mathrm{P}) \leq d_L^2(\mathscr{C}_j, \mathrm{P}),\ j = 1, \ldots, N, \text{ and } j \neq i\}.$$

A Voronoi-Laguerre polygon is always convex. A tessellation of the plane into Voronoi-Laguerre polygons is called a *Voronoi-Laguerre diagram*. Obviously, if $r_i = r_j$ for all $i, j = 1, \ldots, N$, the Voronoi-Laguerre diagram is an ordinary Voronoi diagram. Notice that it may happen that the Voronoi-Laguerre polygon $V(\mathscr{C}_i)$ does not contain any point of the plane. This happens when the half-planes generated by the VorLag lines formed by $\mathscr{C}_i$ and its nearby circles have no overlap. In this case, $V(\mathscr{C}_i)$ is called a *null polygon*. The occurrence of null polygons is specific of Voronoi-Laguerre diagrams and reflects a situation of complete redundancy that is not captured by traditional Voronoi diagrams for which the generated polygons are always not null.

In the following, the sensor s_i whose sensing circle $\mathscr{C}_i$ generates the polygon $V(\mathscr{C}_i)$ is called the *generator* of $V(\mathscr{C}_i)$; the vertices of the same polygon are hereby shortly referred to as *Voronoi-Laguerre vertices*.

Two sensors are *Voronoi-Laguerre neighbors* if their polygons have one edge in common. Given a sensor $s_i \in \mathcal{S}$, the set of its Voronoi-Laguerre neighbors is hereafter referred to as $\mathcal{N}_\mathcal{S}(s_i)$. Furthermore, we refer to $\mathcal{N}_\mathcal{S}^\emptyset(s_i)$ as the set of sensors with null polygons which have a sensing overlap with the sensor s_i:

$$\mathcal{N}_\mathcal{S}^\emptyset(s_i) = \{s_j \in \mathcal{S} : d_E(s_i, s_j) \leq (r_i + r_j) \wedge V(\mathscr{C}_j) = \emptyset\}.$$

The reason why Voronoi Laguerre diagrams perfectly model the coverage problem in the case of heterogeneous sensors is their capability to partition the area of interest into polygonal regions which in fact represent the responsibility regions of the deployed sensors. Indeed, a fundamental property of the Voronoi diagrams in Laguerre geometry is the following:

THEOREM 3.1. *([Bartolini et al. 2009]) Let us consider N circles $\mathscr{C}_i$, with centers $C_i = (x_i, y_i)$ and radii r_i, $i = 1, \ldots, N$, and let $V(\mathscr{C}_i)$ be the Voronoi-Laguerre polygon of the circle $\mathscr{C}_i$. For all $k, j = 1, 2, \ldots, N$, $V(\mathscr{C}_k) \cap \mathscr{C}_j \subseteq \mathscr{C}_k$.*

Less formally, if a point P of the area of interest is covered by at least one sensor, it is certainly covered also by the sensor s_i that generates the Voronoi-Laguerre polygon $V(\mathscr{C}_i)$ that includes P.

3.1 Characterization of coverage redundancy

We define as *redundant* any sensor $s_i \in \mathcal{S}$ such that the sensing circle $\mathscr{C}_i$ is completely covered by other sensors, namely $\mathscr{C}_i \subseteq \cup_{s_j \in \mathcal{S}, j \neq i} \mathscr{C}_j$. The following corollaries 3.1, 3.2 and 3.3 of Theorem 3.1 show the criteria to decide whether s_i is redundant.

COROLLARY 3.1. *If a sensor s_i does not cover any point of its Voronoi-Laguerre polygon $V(\mathscr{C}_i)$ or it has a null polygon, then its sensing circle $\mathscr{C}_i$ is completely covered by other sensors in $\mathcal{S}$. Therefore s_i is redundant.*

PROOF. We want to prove that if the point $P \in \mathscr{C}_i$ there exists a sensor $s_j \in \mathcal{S}$, with $s_j \neq s_i$, such that $P \in \mathscr{C}_j$, i.e. P is also covered by s_j. Since by hypothesis $V(\mathscr{C}_i) \cap \mathscr{C}_i = \emptyset$, $\mathscr{C}_i$ contains only points that are external to its polygon. Because the Voronoi-Laguerre diagram constitutes a partition of the AoI, it exists $s_j \in \mathcal{S}$ such that $P \in V(\mathscr{C}_j)$. As $P \in V(\mathscr{C}_j) \cap \mathscr{C}_i$, Theorem 3.1 ensures that $P \in \mathscr{C}_j$. □

Corollary 3.1 affirms that if s_i does not cover its polygon, it can be put to sleep without affecting coverage.

COROLLARY 3.2. *Given a sensor s_i which covers only a portion of its polygon $V(\mathscr{C}_i)$, let ℓ be a circular segment on the intersection between the boundary of $\mathscr{C}_i$ and the polygon $V(\mathscr{C}_i)$. All the points on ℓ which are not on edges of $V(\mathscr{C}_i)$ are covered only by s_i.*

PROOF. By hypothesis, there are some points in $V(\mathscr{C}_i)$ that are not covered by $\mathscr{C}_i$. Therefore, due to Theorem 3.1, these points are not covered by any sensor. Consider any circular segment ℓ on the boundary of $\mathscr{C}_i$ and inside $V(\mathscr{C}_i)$ (see Figure 2 in which ℓ is the arc $\widehat{DF}$) and a point P on ℓ but not on the edges of $V(\mathscr{C}_i)$. We want to show that s_i is the only sensor which covers P. Since P is not on the edges of the polygon, it is possible to find a value of ϵ arbitrarily small, such that the ϵ-surrounding of P is internal to $V(\mathscr{C}_i)$. The intersection of this ϵ-surrounding with the region $V(\mathscr{C}_i) \setminus \mathscr{C}_i$ (that in Figure 2 is delimited by the segments $\overline{EF}$, $\overline{DE}$ and by the arc $\widehat{DF}$) is obviously uncovered.

We now proceed by contradiction. Let us assume that there is another sensor $s_j \in \mathcal{S}$ such that P is also covered by s_j. Since, by construction, any ϵ-surrounding of P contains an uncovered region, the circle $\mathscr{C}_i$ can cover P only with its boundary. Furthermore, since s_j cannot cover points of $V(\mathscr{C}_i) \setminus \mathscr{C}_i$, then $\mathscr{C}_j$ must be tangential to $\mathscr{C}_i$ in P, and must have a lower sensing radius $r_j < r_i$. This implies that P would be crossed by the Voronoi-Laguerre edge formed by s_i and s_j, and the portion of $V(\mathscr{C}_i)$ on the opposite side of this edge with respect to $\mathscr{C}_i$ could not belong to $V(\mathscr{C}_i)$, contradicting our construction. □

Corollary 3.2 states that if s_i only partially covers its polygon, it cannot reduce its sensing radius without affecting coverage.

COROLLARY 3.3. *Let us consider a sensor s_i, with sensing circle $\mathscr{C}_i$ and Voronoi-Laguerre polygon $V(\mathscr{C}_i)$. Let P be a point such that $P \in V(\mathscr{C}_i) \cap \mathscr{C}_i$. If P is covered also by a sensor $s_k \in \mathcal{S}$ other than s_i, then $s_k \in \mathcal{N}_{\mathcal{S}}(s_i) \cup \mathcal{N}^{\emptyset}_{\mathcal{S}}(s_i)$. In other words, any point of $V(\mathscr{C}_i)$ that is covered by more than one sensor, is certainly covered at least by the generating sensor s_i and by either one of its Voronoi-Laguerre neighbors or a sensor with null polygon.*

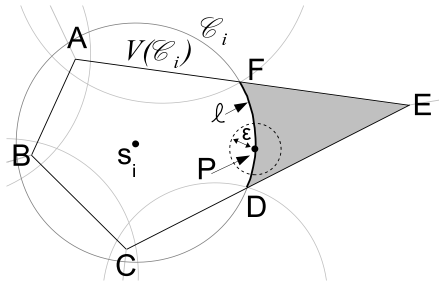


Fig. 2. Voronoi-Laguerre polygon partially covered by its generating sensor.

PROOF. Let $\mathcal{D}$ be the Voronoi-Laguerre diagram generated by $\mathcal{S}$ and $\mathcal{D}'$ be the diagram generated by $\mathcal{S}' = \mathcal{S} \setminus \{s_i\}$. In the diagram $\mathcal{D}$, $P \in V(\mathscr{C}_i)$. By contrast, in the diagram $\mathcal{D}'$, the sensor s_i is not present.

Since by the hypothesis, P is covered by a sensor in $\mathcal{S}'$, thanks to Theorem 3.1 we can affirm that P is also covered by the generating sensor s_k of the polygon, such that $P \in V'(\mathscr{C}_k)$ defined in $\mathcal{D}'$. Obviously, $V'(\mathscr{C}_k) \neq V(\mathscr{C}_k)$. Let us assume, for sake of contradiction, that $s_k \notin N_{\mathcal{S}}(s_i) \cup \mathcal{N}^{\emptyset}_{\mathcal{S}}(s_i)$. If the sensor s_k is not a Voronoi-Laguerre neighbor of s_i and it has not a null polygon in $\mathcal{D}$, its polygon in $\mathcal{D}'$ would be the same as in $\mathcal{D}$, because it would be delimited by edges formed by sensors other than s_i. Therefore it would be $V'(\mathscr{C}_k) = V(\mathscr{C}_k)$, which is a contradiction. $\square$

Corollary 3.3 states that in order to decide whether s_i can reduce its radius or be put to sleep it is sufficient to evaluate the coverage of the sensors in $\mathcal{N}_{\mathcal{S}}(s_i) \cup \mathcal{N}^{\emptyset}_{\mathcal{S}}(s_i)$.

3.2 Reducing the redundancy of sensors with adjustable sensing radius

The Corollaries 3.1, 3.2 and 3.3 let us determine whether an adjustable sensor s_i can reduce its sensing radius or go to sleep. In particular: (1) if the sensor s_i does not cover any point of its polygon, s_i can be put to sleep (in consequence of Corollary 3.1); (2) if s_i covers its polygon only partially, s_i must stay awake and work with its current radius (in consequence of Corollary 3.2); (3) if s_i covers its polygon completely, it may reduce its sensing radius of an extent that can be determined on the basis of the coverage of its neighbors (in consequence of Corollary 3.3).

We now address the third situation more in detail. Let $f(V(\mathscr{C}_i))$ be the *farthest vertex* of the polygon $V(\mathscr{C}_i)$ from the generating sensor s_i. If s_i covers its polygon completely, it can shrink its sensing radius to the distance between s_i and $f(V(\mathscr{C}_i))$, without affecting coverage.

As an example of sensing radius reduction, let us consider the sensor s_1 in Figure 3. In Figure 3(a) the farthest vertex of $V(\mathscr{C}_1)$ is at a distance from s_1 which is smaller than its radius. Because of Theorem 3.1 we can assert that all the points that are internal to $\mathscr{C}_1$ but do not belong to $V(\mathscr{C}_1)$ are covered by the sensors generating the Voronoi-Laguerre polygon to which they belong. Therefore s_1 redundantly covers the region within its circle that is external to its polygon and it can reduce its radius to cover no farther than $f(V(\mathscr{C}_1))$, maintaining full coverage of its responsibility region. Such a reduction of the sensing radius of s_1 is shown in Figure 3(b). Changing the sensing radius of s_1 requires the Voronoi

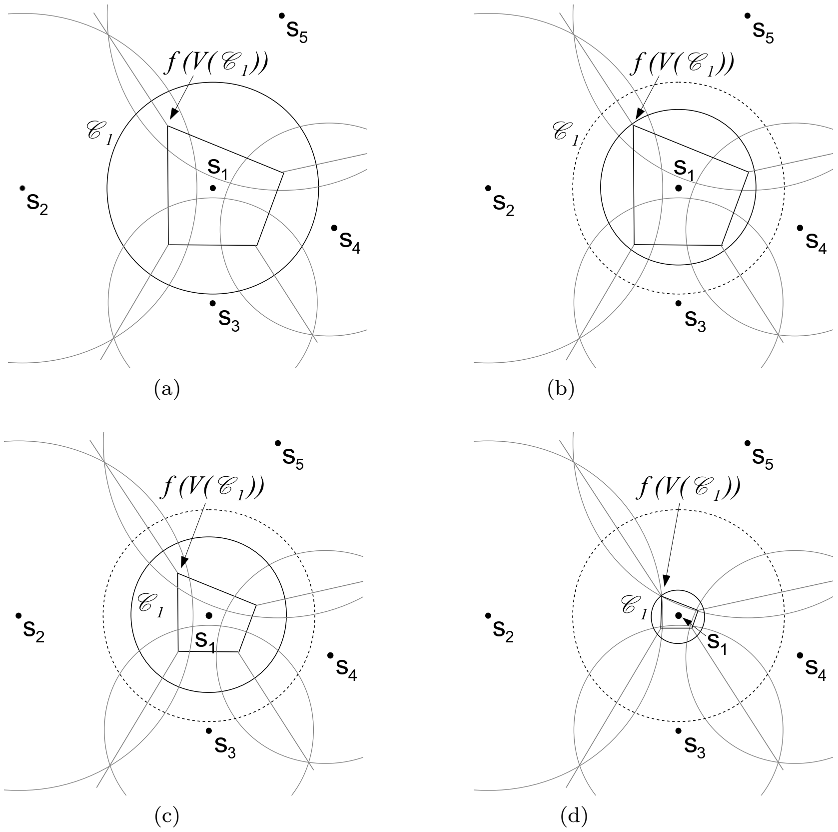


Fig. 3. Iterative reduction of the sensing radius of sensor s_1 to the farthest vertex of its Voronoi-Laguerre polygon.

Laguerre polygons of s_1 and its Laguerre neighbors to be recomputed, as shown in Figure 3(c). This reduction step can be repeated until the radius of the sensor s_1 is such that the farthest vertex of the polygon $V(\mathscr{C}_1)$ is on the circle $\mathscr{C}_1$ and the radius cannot be reduced anymore (see Figure 3(d)).

This repeated reduction of the sensing radius is at the basis of SARA, where sensing radii of adjustable sensors are reduced until even a single radius reduction would leave a coverage hole. Note that this process may even lead some sensors to shrink their sensing radius to zero (in case of redundant sensors), which implies that such sensors are put to sleep.

3.2.1 *A characterization of boundary farthest vertices: Loose and strict farthest vertices.* SARA typically considers the distance to the farthest vertex of a Voronoi-Laguerre polygon as a lower bound for the reduction of the sensing radius of the generating sensor. If the radius is reduced below this threshold, there is a loss of coverage in almost all cases. Nevertheless in some extremely rare configurations[3] it is possible to reduce the radius below this distance without any coverage loss, by

[3] In the experiments we obtained such a situation only by construction.

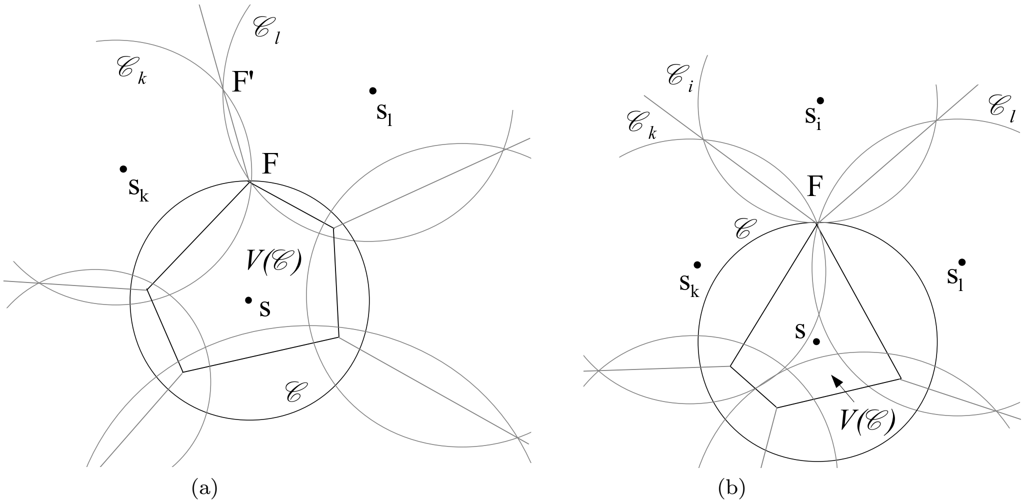


Fig. 4. Strict (a) and loose (b) farthest vertices

enforcing an ordering in the radius reduction of neighbor sensors.

Given a sensor s_i and the farthest vertex of its polygon $f(V(\mathscr{C}_i))$, the sensor s_i is called the *generating sensor* of the farthest vertex $f(V(\mathscr{C}_i))$, while and $f(V(\mathscr{C}_i))$ is called a *boundary farthest* if it lies on the boundary of $\mathscr{C}_i$.

A boundary vertex is the intersection point of at least three circles and of their three Voronoi-Laguerre axes, and therefore is a boundary vertex for at least three sensors [Delman and Galperin 2003]. In the following we say that the boundary farthest vertex of a sensor s_i is a *strict farthest* if the radius of s_i cannot be reduced without leaving a coverage hole. Otherwise such a vertex is called a *loose farthest*. An example of strict and loose boundary farthest vertex is given in Figure 4 (a) and (b), respectively. In the example all sensor nodes have reduced their radius to their farthest vertex F which is therefore a boundary farthest vertex. The point F in Figure 4 (a) is a strict boundary farthest for all the generating sensors. By contrast, in Figure 4 (b), F is a loose boundary farthest for sensor s, in fact, s can significantly reduce its sensing radius without compromising coverage. However, a common farthest that is loose for a generating sensor is not necessarily loose for the others. Point F is a strict farthest for the three other sensors s_i, s_l and s_k which cannot reduce their radius.

In general, if s is the only generating sensor for which a boundary farthest is loose, it can reduce its radius without creating any coverage hole. The other generating sensors cannot perform any concurrent reduction since their farthest vertex is strict. In this case, in order to calculate its new radius, s has to subtract from its responsibility region $V(\mathscr{C})$ all the areas covered by the neighbor sensors generating the loose farthest and guarantee to cover the farthest point of the remaining region $\overline{V}(\mathscr{C})$, in this case $\overline{V}(\mathscr{C}) = V(\mathscr{C}) \setminus (\mathscr{C}_k \cup \mathscr{C}_l)$. Figure 5(a) shows how the sensor s seen in Figure 4(b) can reduce its radius to the minimum needed to cover the farthest point B of the region ABCD $=\overline{V}(\mathscr{C})$, shaded in the figure. After this radius reduction, s needs to recalculate its Voronoi-Laguerre polygon and possibly perform a further radius reduction, as in Figure 5(b).

Although it is very unlikely to occur, it is theoretically possible for a boundary farthest vertex to be loose for two or more generating sensors. In such a case, a concurrent radius reduction of the two or more sensors having a loose farthest vertex might result in a coverage hole. For this reason we introduce a simple *decision serialization scheme for loose farthest vertices*. This can be easily implemented by means of either a back-off policy or a leader arbitrated radius reduction. As there are many well established techniques to solve the problem of serializing decisions in a distributed computing setting, for the sake of simplicity and brevity, we do not address this aspect in the presentation of the algorithm.

We refer the reader to [Bartolini et al. 2010] for the details of the simple geometrical rules sensors adopt to determine if their boundary farthest vertex is strict or loose.

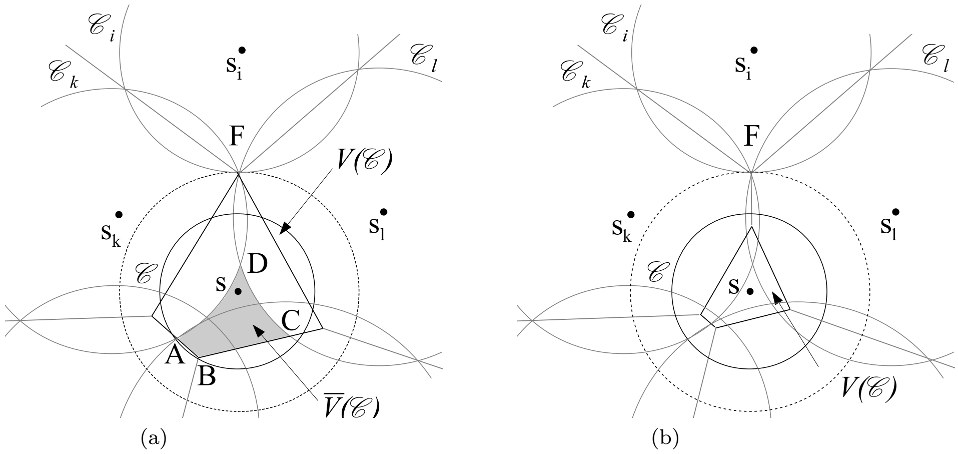


Fig. 5. Reduction of the sensing radius in a situation of loose boundary farthest vertex.

3.3 Putting to sleep sensors with fixed sensing radius

Not having the capability of tuning the extent of its sensing radius, the only way that a node with fixed radius has to save energy is to go to sleep when it is redundant. Therefore, the approach we take for selecting which node with fixed radius should go to sleep is based on a greedy algorithm run by each node s. After a local exchange of information, s determines if neighboring nodes can completely cover for it, and if s is the "best" node for going to sleep, i.e., the node that allows the most energy conservation.

The extent of information needed by a node for deciding whether or not to go to sleep can be kept significantly low by exploiting the Voronoi-Laguerre tessellation, in agreement with Corollaries 3.1, 3.2 and 3.3. Three cases may occur: (1) the sensing circle $\mathscr{C}$ of s does not cover any point of its Voronoi-Laguerre polygon $V(\mathscr{C})$, (2) the sensing circle $\mathscr{C}$ only partially covers $V(\mathscr{C})$, (3) the sensing circle $\mathscr{C}$ completely covers the polygon $V(\mathscr{C})$.

In case (1), Corollary 3.1 states that s is certainly redundant. In case (2), Corollary 3.2 states that sensor s is necessary for coverage and therefore cannot be put to sleep. In case (3) the sensor s must evaluate the coverage of its Voronoi-Laguerre neighbors and of the sensors intersecting $V(\mathscr{C})$ which have null polygons, and determine its redundancy on the basis of Corollary 3.3. The mentioned corollaries set the limit to the number of nodes with which s needs to exchange information in order to decide whether to go to sleep or not.

4. THE ALGORITHM `SARA`

`SARA` is executed in parallel by all the sensors of the network. Its execution results in the selection of a subset of sensors to be kept awake while the others go to sleep. `SARA` also allows a node with adjustable radius that is awake to tune its sensing radius. The obtained sensor activation and radius adjustment is used for a time, called *operative time interval*, that lasts until `SARA` is re-executed. The operative time interval is not necessarily fixed since `SARA` execution can be event-driven.[4]

Each sensor makes the decision about whether to stay awake and about reducing its radius (if possible) iteratively. In order to do so, at each iteration k, each node determines its own Voronoi-Laguerre polygon. This requires the node to be aware of its one-hop neighbors (nodes it can communicate with directly), their location[5] and their sensing radius. The iteration is then composed by two phases. During the first phase nodes with fixed radius decide whether to go to sleep or not. In the second phase, the nodes with adjustable radius perform their radius reduction. Each node s_i bases its decision on a parameter $\alpha_i^{(k)} \in (0, 1]$, which depends on the energy gain that the sensor will achieve by either going to sleep or by reducing its sensing radius. This parameter is used differently depending on whether a node has a fixed or an adjustable radius. Specifically, a node s_i with fixed radius will go to sleep with probability $\alpha_i^{(k)}$ provided that there are neighboring nodes that are awake and redundantly cover its sensing circle. On the other hand, if s_i has an adjustable radius it will reduce it by the fraction $\alpha_i^{(k)}$ of the maximum radius reduction that does not alter the coverage of its responsibility region. As we will prove in Section 5, the iterative execution of the two phases leads to a network configuration in which there is no redundant fixed sensor and it is not possible to further reduce the radius of any adjustable sensor without creating new coverage holes.

4.1 `SARA` in details

4.1.1 *Initialization.* `SARA` is described by Algorithms 1 and 3, for nodes with fixed and adjustable radius, respectively. At the start of `SARA` operations, each sensor sets the iteration counter k and the value of its sensing radius (the maximum value in the case of sensors with adjustable radius). The flag `decision_made` is set

[4] An event-driven reconfiguration requires that sensors operating in low power mode can be contacted by the sink by means of an interest dissemination. Sleeping nodes equipped with a wake-up radio [Gu and Stankovic 2004] can be woken up upon need and can therefore safely put to sleep their radio for the whole duration of the operative time interval. If such extra HW is not available nodes in low power mode must periodically wake up according to a very low duty cycle so that changes in the mode of operation of the network can be signaled.

[5] This information may be obtained through extra hardware such as GPS, if available, or through one of the many localization schemes recently proposed.

to `false` indicating that the node is undecided. The node remains awake and undecided until, in one of the iterations, it makes a final decision on the value of its sensing radius to be used till a new `SARA` execution.

Initialization also includes the setting of a timer needed for protocol operations.

4.1.2 *Computing* $\alpha_i^{(k)}$. Consider the k-th iteration of `SARA`. Let $S_A^{(k)} = S_{\texttt{fixed}}^{(k)} \cup S_{\texttt{adjustable}}^{(k)}$ be the set of sensors that are still awake, and let $S_{\texttt{undecided}}^{(k)} \subseteq S_A^{(k)}$ be the set of sensors that have not made their final configuration decision. Similarly, $S_{\texttt{decided}}^{(k)} = S_A^{(k)} \setminus S_{\texttt{undecided}}^{(k)}$ is the set of sensors that are still awake and have already made their configuration decision.

Consider $s_i \in S_{\texttt{undecided}}^{(k)}$. When making its decision s_i for the current iteration, s_i takes account of decided and undecided neighbors in a different manner. In particular, let $\mathscr{L}^{(k)}(s_i)$ be the subset of $S_{\texttt{undecided}}^{(k)}$ including s_i and all the undecided sensors that are either Voronoi-Laguerre neighbors of s_i or have a null polygon and their sensing circle intersects $\mathscr{C}_i$: $\mathscr{L}^{(k)}(s_i) = S_{\texttt{undecided}}^{(k)} \cap (\mathcal{N}_{S_A^{(k)}}(s_i) \cup \mathcal{N}^{\emptyset}_{S_A^{(k)}}(s_i) \cup \{s_i\})$. Let also $\mathscr{D}^{(k)}(s_i) = \mathcal{S}_{\texttt{decided}}^{(k)} \cap (\mathcal{N}_{S_A^{(k)}}(s_i) \cup \mathcal{N}^{\emptyset}_{S_A^{(k)}}(s_i))$ be the subset of the sensors that have already made their decision and are either Voronoi-Laguerre neighbors of s_i or have a null polygon and overlap the sensing circle $\mathscr{C}_i$.

The computation of the parameter $\alpha_i^{(k)}$ depends on the comparison between s_i and the nodes in $\mathscr{L}^{(k)}(s_i)$ with respect to the decrease in energy consumption that is achievable through sensing radius reduction while ensuring coverage. The comparison is motivated by the fact that these nodes are those that still have the chance to reduce their sensing radius and consequently their energy expenditure. The value of $\alpha_i^{(k)}$ should be higher for a node s_i when choosing it for sensing radius reduction or for going to sleep leads to a better performance gain than choosing the other nodes in the neighborhood.

The criterion we propose to compute $\alpha_i^{(k)}$ is based on the *energy gain*, defined as the amount of energy that a sensor can save by reducing its sensing radius to the farthest point of the responsibility region (in case of sensors with adjustable radius) or by going to sleep (case of sensors with fixed sensing radius).

We recall that $E_{\texttt{sensing}}$ is the energy expenditure per unit time due to sensing, defined in Equation 1.

For sensors with fixed sensing radius, the energy gain of sensor s_i in the k-th iteration is defined as $\Delta E_i^{(k)} = E_{\texttt{sensing}}(r_i^{\texttt{fixed}})$. For sensors with adjustable sensing radius, it is $\Delta E_i^{(k)} = E_{\texttt{sensing}}(r_i^{(k-1)}) - E_{\texttt{sensing}}(d_E(s_i, f(\overline{V}(\mathscr{C}_i^{(k)}))))$, with $\overline{V}(\mathscr{C}_i^{(k)}) = V(\mathscr{C}_i^{(k)}) \setminus \cup_{s_j \in \mathscr{D}^{(k)}(s_i)} \mathscr{C}_j$. For adjustable sensors having a null or a completely uncovered polygon, the energy gain is $\Delta E_i^{(k)} = E_{\texttt{sensing}}(r_i^{(k-1)})$.

The energy gain criterion sets the value of $\alpha_i^{(k)}$ as follows:

$$\alpha_i^{(k)} = \max\left\{ \frac{\Delta E_i^{(k)} - \Delta E_i^{\texttt{min}\ (k)}}{\Delta E_i^{\texttt{max}\ (k)} - \Delta E_i^{\texttt{min}\ (k)}}, \alpha_{\texttt{min}} \right\}, \tag{3}$$

where the parameter $\alpha_{\texttt{min}}$ is an arbitrarily small constant, such that $0 < \alpha_{\texttt{min}} \ll 1$, $\Delta E_i^{\texttt{max}\ (k)} = \max_{s_j \in \mathscr{L}^{(k)}(s_i)} \Delta E_j^{(k)}$ is the maximum achievable gain in the neighborhood of s_i and $\Delta E_i^{\mathtt{min}\ (k)} = \min_{s_j \in \mathscr{L}^{(k)}(s_i)} \Delta E_j^{(k)}$ is its minimum value. If $\Delta E_i^{\mathtt{max}\ (k)} = \Delta E_i^{\mathtt{min}\ (k)}$ we consider $\alpha_i^{(k)} = 1$. According to Equation 3, the more a node s_i allows energy saving the higher is the probability that it is selected for going to sleep if s_i is a fixed sensor, or the higher is the reduction of sensing radius that is allowed if s_i is an adjustable sensor. This setting of $\alpha_{\mathtt{min}}$ ensures that even the sensor with smallest potential energy gain can make a decision that improves its energy expenditure.

The energy gain criterion has been compared by means of extensive simulations with several others, including one based on the node residual energy and one based on an estimate of the node expected lifetime. In all the scenarios the energy gain criterion showed superior performance. Therefore, we will focus only on such a criterion for the remainder of the paper. The interested reader can find more details on this aspect in [Bartolini et al. 2010].

4.1.3 `SARA` *for sensors with fixed sensing radius.* At the beginning of `SARA` operations, all the sensors with fixed radius are awake and undecided. All the nodes (fixed and adjustable) exchange position information at the initialization phase. Let us consider the k-th iterative step of `SARA` (k-th execution of the `while` cycle in Algorithm 1). We recall that the set of sensors that are still awake at the k-th iteration is referred to as $S_A^{(k)} = S_{\mathtt{fixed}}^{(k)} \cup S_{\mathtt{adjustable}}^{(k)}$.

Each undecided sensor $s_i \in S_{\mathtt{fixed}}^{(k)}$ performs an information exchange with its neighbors that are still undecided to gather information regarding their radius[6]. With this information, s_i is able to construct its Voronoi-Laguerre polygon $V(\mathscr{C}_i^{(k)})$ and to determine the set $\mathcal{N}_{S_A^{(k)}}(s_i)$.

Node s_i then informs its neighbors with which it has a sensing overlap about whether its polygon is null. This information allows its neighbors to compute their sets $\mathcal{N}_{S_A^{(k)}}^{\emptyset}$. Each node then evaluates its redundancy status (according to Corollaries 3.1, 3.2 and 3.3).

If s_i is not redundant at the k-th iteration, it cannot become redundant subsequently because `SARA` in each iteration can only reduce the number of sensors that can cover an area. Therefore, in the case of non redundancy, s_i decides to stay awake, communicates this decision to the neighbors with a sensing overlap (sending an `I_am_awake` message), and ends the decision phase (setting the `decision_made` flag to `true`).

If sensor s_i is redundant it communicates its potential energy gain to the nodes in $\mathcal{N}_{S_A^{(k)}}(s_i) \cup \mathcal{N}_{S_A^{(k)}}^{\emptyset}(s_i)$. Nodes with a null polygon also send their potential energy gain to all their neighbors with sensing overlap. Each node is then able to construct the set $\mathscr{L}^{(k)}(s_i)$ and compute $\alpha_i^{(k)}$. The calculus of $\alpha_i^{(k)}$ is executed by running the function `get_alpha` described in Algorithm 2.

Since more than one sensor may decide to go to sleep at the same iteration, possibly leaving coverage holes, we introduce a simple back-off scheme to avoid conflicting decisions. More precisely, given a back-off interval $t_{\mathtt{max}}^{\mathtt{backoff}}$, each sensor

[6]It is not necessary to exchange information with the sensors that have already made their configuration decisions.

Algorithm 1: Algorithm `SARA` for fixed sensors

```
Algorithm SARA executed by node s_i
  Initialization:
  k = 0;
  Back-off interval = [0, t_max^backoff];
  r_i^(k) = r_i^fixed;
  decision_made=false;
  Exchange position information with neighbors;

  Iterative Voronoi-Laguerre diagram construction:
  while !decision_made do
    Exchange info on radius with neighbors;
    Construct the VorLag polygon V(C_i^(k));
    Exchange info on null polygons;
    Evaluate redundancy and energy gain;
    if s_i is not redundant then
      // Case of fixed sensors that need to stay awake
      Send I_am_awake message;
      decision_made=true;
      Stay awake;
    else
      // Case of redundant fixed sensor
      Exchange info on energy gain;
    Build set L^(k)(s_i);
    alpha_i^(k) =get_alpha(L^(k)(s_i));
    Choose a random instant t_i^* ∈ [0, t_max^backoff];
    while t < t_i^* do
      Listen to update messages from the neighborhood;
    if s_i is not redundant anymore then
      Send I_am_awake message;
      decision_made=true;
      Stay awake;
    else
      With probability alpha_i^(k)
        Send going_to_sleep message;
        decision_made=true;
        Go to sleep;
    k = k + 1;
```

Algorithm 2: Function to compute parameter α_i

Function `get_alpha(`$\mathscr{L}^{(k)}(s_i)$`)`

Set $\Delta E_i^{\mathtt{max}\ (k)} = \max_{s_j \in \mathscr{L}^{(k)}(s_i)} \Delta E_j^{(k)}$ and

$\Delta E_i^{\mathtt{min}\ (k)} = \min_{s_j \in \mathscr{L}^{(k)}(s_i)} \Delta E_j^{(k)}$;

$\alpha_i^{(k)} = \max\left\{ \frac{\Delta E_i^{(k)} - \Delta E_i^{\mathtt{min}\ (k)}(n)}{\Delta E_i^{\mathtt{max}\ (k)} - \Delta E_i^{\mathtt{min}\ (k)}}, \alpha_{\mathtt{min}} \right\}$;

return $\alpha_i^{(k)}$;

s_i chooses a random instant $t_i^* \in [0, t_{\mathtt{max}}^{\mathtt{backoff}}]$, hereafter called *backoff timeout.* It then waits for a time t_i^*, during which it considers all the messages received from the nodes in radio proximity that may contribute to the redundancy of s_i.

After the expiration of the backoff timeout t_i^*, the sensor s_i verifies if it is still redundant or not. If it is not redundant anymore, s_i decides to stay awake and sets the `decision_made` flag to `true`. It then communicates this decision to its neighbors by sending them an `I_am_awake` message.

If instead s_i is still redundant, it goes to sleep with probability $\alpha_i^{(k)}$. If the node goes to sleep, it sets the `decision_made` flag to `true` and communicates its decision by sending a `going_to_sleep` message.

Notice that a redundant sensor with fixed sensing radius does not necessarily go to sleep at the first iteration. Therefore, the execution of a single iteration of the algorithm does not eliminate the existing redundancy completely. Nevertheless, at each iteration the sensors with higher priority are the ones that more likely will go to sleep. The other redundant sensors will eventually either go to sleep or become non-redundant in one of the subsequent iterations depending on the decisions of their neighbors.

4.1.4 `SARA` *for sensors with adjustable sensing radius.* At the beginning of `SARA` operations all adjustable sensors are undecided and set their radius to the maximum value. As in the case of fixed sensors, all the nodes exchange position information at the initialization phase.

We consider the generic k-th iteration of `SARA` (k-th execution of the `while` cycle in Algorithm 3).

At every algorithm iteration, each sensor $s_i \in \mathcal{S}^{(k)}_{\texttt{adjustable}}$ communicates with its neighbors that are still undecided to disseminate and gather information about their activation status and their currently calculated radius value. The radius reduction for the current iteration of a node $s_i \in \mathcal{S}^{(k)}_{\texttt{adjustable}}$ is calculated after the back-off phase of its neighbors in $\mathcal{S}^{(k)}_{\texttt{fixed}}$. At the end of such a phase, every sensor $s_i \in S^{(k)}_{\texttt{adjustable}}$ updates its Voronoi-Laguerre polygon $V(\mathscr{C}_i^{(k)})$, updates its information for computing $\alpha_i^{(k)}$ if any of its fixed radius neighbor goes to sleep during the back-off, and determines the sets $\mathscr{L}^{(k)}(s_i)$ and $\mathscr{D}^{(k)}(s_i)$. In this way, the sensor s_i has the necessary information to calculate the maximum radius reduction that does not create coverage holes. Notice that in this calculus, the sensors belonging to the two sets $S^{(k)}_{\texttt{decided}}$ and $S^{(k)}_{\texttt{undecided}}$ play a different role since the sensors in $S^{(k)}_{\texttt{decided}}$ will no longer change their configuration for the current execution of `SARA`, therefore their sensing circles can be considered definitely covered and can be subtracted from the responsibility region of those sensors that still have to make their configuration decision. This is the reason why the maximum radius reduction for s_i is computed as the one that does not alter the coverage of the region $\overline{V}(\mathscr{C}_i^{(k)}) = V(\mathscr{C}_i^{(k)}) \setminus \cup_{s_j \in \mathscr{D}^{(k)}(s_i)} \mathscr{C}_j^{(k)}$.

We now define the minimum extent $\overline{d}_i^{(k)}$ of s_i sensing radius on the basis of Corollaries 3.1, 3.2, and 3.3. If s_i is not able to cover any point of $\overline{V}(\mathscr{C}_i^{(k)})$) or $\overline{V}(\mathscr{C}_i^{(k)})) = \emptyset$ then $\overline{d}_i^{(k)} = 0$ (due to Corollary 3.1). If s_i only partially covers its polygon (this occurs if $d_E(s_i, c(\overline{V}(\mathscr{C}_i^{(k)}))) < r_i^{(k)} < d_E(s_i, f(\overline{V}(\mathscr{C}_i^{(k)})))$, where

$c(\overline{V}(\mathscr{C}_i^{(k)}))$ is the closest point of $\overline{V}(\mathscr{C}_i^{(k)})$ from s_i) then $\overline{d}_i^{(k)} = r_i^{(k)}$ (the radius does not change, as determined by Corollary 3.2). Finally, if s_i completely covers its polygon, $\overline{d}_i^{(k)}$ is set to $d_E(s_i, f(\overline{V}(\mathscr{C}_i^{(k)})))$, that is the Euclidean distance between s_i and the farthest point of $\overline{V}(\mathscr{C}_i^{(k)})$.

The sensor s_i, whose radius at the k-th iteration is $r_i^{(k)}$, will then reduce its radius to an intermediate value in the range $[\overline{d}_i^{(k)}, r_i^{(k)})$, whose position is determined by the priority value $\alpha_i^{(k)}$. Therefore s_i calculates the new value of its radius $r_i^{(k+1)}$ as $r_i^{(k+1)} = r_i^{(k)} - \alpha_i^{(k)} \cdot (r_i^{(k)} - \overline{d}_i^{(k)})$.

Each sensor belonging to $\mathcal{S}^{(k)}_{\texttt{adjustable}}$ that reduces its radius affects the potential decisions of its Voronoi-Laguerre neighbors, so the process is iterated until no further reduction is possible, because either a strict farthest vertex is on the boundary of the sensing circle, or the radius of the sensor gradually became null, and the sensor is put to sleep.

5. PROPERTIES OF SARA

The execution of SARA on a set of sensors $\mathcal{S}$ leads to a final configuration that will be hereby called *cover set*. In the following we will shortly denote with $S_{\texttt{SARA}}$ such a cover set, where $S_{\texttt{SARA}}$ is the set of awake sensors with their radius configuration decided by SARA.

The following theorem shows that $S_{\texttt{SARA}}$ provides the same coverage as the starting configuration (the one where all sensors are awake at maximum radius).

THEOREM 5.1. **(Coverage equivalence)** *Consider a set of adjustable and fixed sensors* $\mathcal{S} = \mathcal{S}_{adjustable} \cup \mathcal{S}_{fixed}$. *Let* $\mathscr{A} \subseteq AoI$ *be the area that the sensors in* $\mathcal{S}$ *are able to cover if they are all awake and the adjustable sensors work at their maximum radius. The coverage extension of* $S_{\texttt{SARA}}$ *is equal to* $\mathscr{A}$.

PROOF. Let us denote with $S^{(k)}_{\texttt{SARA}}$ the cover set determined by SARA at the k-th iteration, with $S^{(0)}_{\texttt{SARA}} = S$. Let us also denote with $\mathscr{A}^{(k)} \subseteq AoI$ the portion of the AoI that is covered by $S^{(k)}_{\texttt{SARA}}$, therefore

$$\mathscr{A}^{(k)} = \cup_{s_j \in S^{(k)}_{\texttt{SARA}}} \mathscr{C}_j^{(k)}.$$

The Voronoi-Laguerre diagram of $S^{(k)}_{\texttt{SARA}}$ creates a partition of the AoI. Therefore, in order to prove that the coverage extension does not decrease after the algorithm execution, it is enough to prove that, at each iteration, the coverage of each polygon is preserved, that is:

$$V(\mathscr{C}_i^{(k)}) \cap \mathscr{A}^{(k)} \subseteq \mathscr{A}^{(k+1)}, \forall s_i \in S^{(k)}_{\texttt{SARA}}. \tag{4}$$

Regarding fixed sensors, SARA allows them to go to sleep one at a time and only if their polygon is already covered by other sensors, so if one of them decides to go to sleep the coverage of its polygon does not decrease, thus guaranteeing that Equation 4 is trivially verified for fixed sensors.

For what concerns the case of adjustable sensors, let us consider any sensor s_i still awake in the k-th iteration. Theorem 3.1 affirms that the covered area of $V(\mathscr{C}_i^{(k)})$

Algorithm 3: Algorithm `SARA` for adjustable sensors

Algorithm `SARA` executed by node s_i
`// before starting the next operative time interval, the sensor` s_i `works with the radius`
`// determined at the previous execution of SARA`
Initialization:
$k = 0$;
Back-off interval $= [0, t_{\mathtt{max}}^{\mathtt{backoff}}]$;
$r_i^{(k)} = r_i^{\mathtt{max}}$;
`decision_made=false`;
Exchange position information with neighbors;

Iterative Voronoi-Laguerre diagram construction:
while !*decision_made* **do**
  Exchange info on radius with neighbors;
  Construct the VorLag polygon $V(\mathscr{C}_i^{(k)})$;
  Exchange redundancy/polygon nullity information messages and potential energy gain;
  while $t < t_{max}^{backoff}$ **do**
    listen to update messages from the fixed nodes in the neighborhood;
  Update the VorLag polygon $V(\mathscr{C}_i^{(k)})$;
  Build sets $\mathscr{L}^{(k)}(s_i)$ and $\mathscr{D}^{(k)}(s_i)$;
  Let $\overline{V}(\mathscr{C}_i^{(k)}) = V(\mathscr{C}_i) \setminus \cup_{s_j \in \mathscr{D}^{(k)}(s_i)} \mathscr{C}_j$;
  Let $f(\overline{V}(\mathscr{C}_i^{(k)}))$ be the farthest point of $\overline{V}(\mathscr{C}_i^{(k)})$ from s_i;
  Let $c(\overline{V}(\mathscr{C}_i^{(k)}))$ be the closest point of $\overline{V}(\mathscr{C}_i^{(k)})$ from s_i;
  if $(d_E(s_i, c(\overline{V}(\mathscr{C}_i^{(k)}))) < r_i^{(k)} < d_E(s_i, f(\overline{V}(\mathscr{C}_i^{(k)})))) \vee$ $(f(\overline{V}(\mathscr{C}_i^{(k)}))$ *is a strict farthest* $) \vee (r_i^{(k)} = 0))$ **then**
    `// reached minimum radius`
    `decision_made` $=$ `true`;
  else
    if $r_i^{(k)} < d_E(s_i, c(\overline{V}(\mathscr{C}_i^{(k)})))$ **then**
      `// completely uncovered polygon`
      $\overline{d}_i^{(k)} = 0$;
    else
      $\overline{d}_i^{(k)} = d_E(s_i, f(\overline{V}(\mathscr{C}_i^{(k)})))$;
    $\alpha_i = get\_alpha(\mathscr{L}(s_i))$;
    $r_i^{(k+1)} = r_i^{(k)} - \alpha_i^{(k)}(r_i^{(k)} - \overline{d}_i^{(k)})$;
    $k = k + 1$;

if $r_i^{(k)} = 0$ **then**
  `// null or completely uncovered polygon`
  go to sleep;
else
  Adjust the sensing radius to $r_i^{(k)}$;

is all covered by s_i. This means that, for any $s_i \in S^{(k)}_{\texttt{SARA}}$ and for any iteration k:

$$V(\mathscr{C}_i^{(k)}) \cap \mathscr{A}^{(k)} \overset{Th.3.1}{=} V(\mathscr{C}_i^{(k)}) \cap \mathscr{C}_i^{(k)}.$$

Therefore in order to prove Equation 4, it is sufficient to prove that

$$V(\mathscr{C}_i^{(k)}) \cap \mathscr{C}_i^{(k)} \subseteq \mathscr{A}^{(k+1)}, \forall s_i \in S^{(k)}_{\texttt{SARA}} \cap \mathcal{S}_{\texttt{adjustable}}. \tag{5}$$

Let us consider a further partition of $V(\mathscr{C}_i^{(k)})$ in the following two subsets: $V^1_{i,k} \triangleq \overline{V}(\mathscr{C}_i^{(k)}) = V(\mathscr{C}_i^{(k)}) \setminus (\cup_{s_j \in \mathscr{D}_i^{(k)}} \mathscr{C}_j^{(k)})$, and $V^2_{i,k} \triangleq V(\mathscr{C}_i^{(k)}) \setminus \overline{V}(\mathscr{C}_i^{(k)}) = V(\mathscr{C}_i^{(k)}) \cap (\cup_{s_j \in \mathscr{D}_i^{(k)}} \mathscr{C}_j^{(k)})$.

We will now prove that Equation 5 is verified by separately considering the two subsets $V^1_{i,k}$ and $V^2_{i,k}$. Let us first consider $V^1_{i,k}$. SARA reduces the radius of an adjustable sensor s_i to a value such that the coverage of the region $V^1_{i,k} = \overline{V}(\mathscr{C}_i^{(k)})$ is not altered. Therefore,

$$V^1_{i,k} \cap \mathscr{C}_i^{(k)} \overset{\texttt{SARA}}{=} V^1_{i,k} \cap \mathscr{C}_i^{(k+1)} \subseteq \mathscr{C}_i^{(k+1)} \subseteq \mathscr{A}^{(k+1)}.$$

We now show that the same property holds for $V^2_{i,k}$. By the definition of $\mathscr{D}_i^{(k)}$, sensors belonging to $\mathscr{D}_i^{(k)}$ are such that their sensing circles do not change in the following iterations, therefore if $s_j \in \mathscr{D}_i^{(k)}$ then $\mathscr{C}_j^{(k)} = \mathscr{C}_j^{(k+1)}$. Therefore,

$$V^2_{i,k} \cap \mathscr{C}_i^{(k)} \subseteq V^2_{i,k} = V(\mathscr{C}_i^{(k)}) \cap (\cup_{s_j \in \mathscr{D}_i^{(k)}} \mathscr{C}_j^{(k)}) \overset{\texttt{Def of } \mathscr{D}_i^{(k)}}{=}$$

$$= V(\mathscr{C}_i^{(k)}) \cap (\cup_{s_j \in \mathscr{D}_i^{(k)}} \mathscr{C}_j^{(k+1)}) \subseteq \cup_{s_j \in \mathscr{D}_i^{(k)}} \mathscr{C}_j^{(k+1)} \subseteq \mathscr{A}^{(k+1)}.$$

Since $V(\mathscr{C}_i^{(k)}) \cap \mathscr{C}_i^{(k)} = (V^1_{i,k} \cup V^2_{i,k}) \cap \mathscr{C}_i^{(k)} \subseteq \mathscr{A}^{(k+1)}$, Equation 4 is verified.

□

THEOREM 5.2. **(Convergence in the case of adjustable sensors)** *Given a set $\mathcal{S} = \mathcal{S}_{\texttt{adjustable}}$ of only adjustable sensors, under the execution of* SARA*, each sensor will converge to a final configuration decision.*

PROOF. Consider the adjustable sensor $s_i \in \mathcal{S}$, positioned in C_i. Let $r_i^{(k)}$ be its sensing radius at the k-th iteration of SARA, and let $\mathscr{C}_i^{(k)}$ and $V(\mathscr{C}_i^{(k)})$ be its sensing circle and its Voronoi-Laguerre polygon, respectively. We distinguish three cases: (1) $V(\mathscr{C}_i^{(k)})$ is completely covered (notice that this case includes the situation of null polygons which can be considered a degeneration of non null polygons), (2) $V(\mathscr{C}_i^{(k)})$ is only partially covered and (3) $V(\mathscr{C}_i^{(k)})$ is not covered (neither by s_i nor by any other sensor, due to Theorem 3.1).

Convergence in case (1). Theorem 3.1 ensures that $V(\mathscr{C}_i^{(k)})$ is completely covered by s_i. Since SARA preserves coverage (for Theorem 5.1), the new polygon and its farthest point will also be covered by s_i at any successive iteration of SARA. We recall that $\overline{V}(\mathscr{C}_i^{(k)}) = V(\mathscr{C}_i^{(k)}) \setminus (\cup_{s_j \in \mathscr{D}_i^{(k)}} \mathscr{C}_j^{(k)})$, $\overline{d}_i^{(k)} = d_E(s_i, f(\overline{V}(\mathscr{C}_i^{(k)})))$ and

$d_i^{(k)} = d_E(s_i, f(V(\mathscr{C}_i^{(k)})))$. As $\overline{V}(\mathscr{C}_i^{(k)}) \subseteq V(\mathscr{C}_i^{(k)}) \subseteq \mathscr{C}_i^{(k)}$ the following holds:

$$0 \leq \overline{d}_i^{(k)} \leq d_i^{(k)} \leq r_i^{(k)}. \tag{6}$$

Since $r_i^{(k)}$ is strictly decreasing and non-negative, when $k \to \infty$, it converges to a value $R_i \geq 0$. SARA sets the radius of s_i for the next iteration as: $r_i^{(k+1)} = r_i^{(k)} - \alpha_i^{(k)} \cdot (r_i^{(k)} - \overline{d}_i^{(k)})$, where $\alpha_i^k \in (0, 1]$. It follows that $R_i = R_i - \lim_{k\to\infty} \alpha_i^k \cdot (R_i - \lim_{k\to\infty} \overline{d}_i^{(k)})$. As $\alpha_i^k > \alpha_{\mathtt{min}}$ is strictly positive and lower than 1, then $\lim_{k\to\infty} \overline{d}_i^{(k)} = R_i$.

The convergence of $\lim_{k\to\infty} d_i^{(k)}$ follows, due to Equation 6, by applying the comparison criterion. This means that the radius of s_i converges to the minimum value to cover the farthest vertex of its polygon, which is a *boundary farthest configuration.*

If such a boundary farthest vertex is strict, then s_i terminates its execution of SARA. Otherwise, the adoption of the *serialization scheme for loose farthest vertices* discussed in Section 3.2.1 ensures that all the sensors with loose vertices will perform their additional radius reduction one at a time. After this radius reduction, s_i will never generate again a loose farthest with the same neighbors (as this would require an increase in the sensing range of at least one sensor, which is not allowed by SARA). Since there is a finite number of neighbor sensors that can generate a loose farthest with s_i, then s_i will eventually reach a strict farthest situation and will exit.

Convergence in case (2). In this case, as the coverage of the polygon is only partial, the sensor cannot reduce its radius (due to Corollary 3.2) and SARA immediately terminates.

Convergence in case (3). Consider $k = 0$. In this case $V(\mathscr{C}_i^{(0)}) \cap \mathscr{C}_i^{(0)} = \emptyset$. Notice that, as for all Voronoi polygons, at the successive iterations, the polygon of s_i can only be altered by its own radius reduction or by the reductions performed by its neighbors.

As the polygon $V(\mathscr{C}_i^{(0)})$ is not covered, the polygons of the Voronoi-Laguerre neighbors of s_i are either partially covered or completely uncovered, because they share an edge with $V(\mathscr{C}_i^{(0)})$. A radius reduction of a neighbor with completely uncovered polygon may result in an extension of the polygon of s_i with new uncovered zones. By contrast, the neighbors of s_i which partially cover their polygons will not change their radius. Therefore, for any iteration $k > 0$, $V(\mathscr{C}_i^{(k)}) \cap \mathscr{C}_i^{(k)} = \emptyset$, that is a polygon which is initially uncovered will remain uncovered, even if the polygon changes.

Hence, for a sensor s_i being in case (3), $\overline{d}_i^{(k)} = 0$, $\forall k \geq 0$. This implies that $r_i^{(k+1)} = (1 - \alpha_i^{(k)}) \cdot r_i^{(k)} \leq (1 - \alpha_{\mathtt{min}}) \cdot r_i^{(k)}$, $\forall k \geq 0$. Therefore $\lim_{k\to\infty} r_i^{(k)} \leq \lim_{k\to\infty} (1 - \alpha_{\mathtt{min}})^k \cdot r_i^{(0)} = 0$, proving that the sensor s_i converges to a final configuration in which it will be put to sleep[7]. ∎

[7]Notice that, although s_i knows from the beginning that its decision will be to go to sleep, we decided to let it execute the algorithm iteratively in order not to alter the decision priority established by the energy gain criterion.

THEOREM 5.3. **(Termination in the case of fixed sensors)** *Given a set $\mathcal{S} = \mathcal{S}_{fixed}$ of only fixed sensors,* SARA *puts to sleep all the redundant sensors in a finite time.*

PROOF. At the k-th iteration of SARA, every fixed sensor determines whether it is redundant or not. If it is not redundant it immediately ends its execution with the decision to stay awake. If instead it is redundant it goes to sleep with probability α_i (see Algorithm 1). At every iteration k of the algorithm, there is at least one sensor s_i (namely the one with maximum value of ΔE_i) whose value of $\alpha_i^{(k)}$ is equal to 1 and therefore has probability 1 to go to sleep. It follows that at each iteration at least one redundant sensor goes to sleep (although in practice many sensors go to sleep at each iteration, as shown in Section 7). Hence, in a finite number of steps all redundant fixed sensors will go to sleep. ∎

THEOREM 5.4. *(***Convergence of SARA in the general scenario***) Given a set $\mathcal{S} = \mathcal{S}_{adjustable} \cup \mathcal{S}_{fixed}$ of both adjustable and fixed sensors, under the execution of* SARA*, each sensor converges to a final configuration decision.*

PROOF. The convergence of SARA easily descends from Theorems 5.2 and 5.3.

It has to be noted that although the presence of fixed sensors does not alter the convergence property of the adjustable sensors, the opposite is not true. In fact, the presence of adjustable sensors in the mix alters the behavior of the fixed sensors as it is no longer guaranteed that at every iteration k there will be a redundant fixed sensor that will go to sleep. Although it is still true that there will be at least one sensor $s_i^{(k)}$ in $\mathcal{S}$ with $\alpha_i^{(k)} = 1$, this sensor may belong to the adjustable class. Therefore, the convergence speed of the fixed class is slowed down by the presence of the adjustable sensors[8]. ∎

Theorem 5.4 states the convergence of SARA in the mixed scenario. The adjustable sensors might theoretically reduce their radius of an infinitesimal step at each iteration. In order to ensure the theoretical termination of the algorithm in a finite number of steps we can set an upper limit K on the number of iterations (*faster termination condition*). Despite convergence might theoretically take quite long time we have observed that no more than 20 iterations are sufficient to achieve termination of the 95% of sensors. Setting a value of K as low as 20 has a negligible impact on the performance of SARA, but has the advantage to ensure a very fast termination of the algorithm execution.

The following Lemma 5.5 analyzes the property of the cover set obtained after the execution of SARA focusing in particular on the polygons generated by the adjustable sensors.

LEMMA 5.5. **(Properties of the cover set)** *Consider a mixed set of adjustable and fixed sensors $\mathcal{S} = \mathcal{S}_{adjustable} \cup \mathcal{S}_{fixed}$. If $s_i \in S_{SARA} \cap \mathcal{S}_{adjustable}$, either s_i partially*

[8]This is because we want the two classes of sensors to reduce their radius in parallel without favoring a given class. If, due to a particular operative setting, one of the two classes should have a higher priority in making configuration decisions, this can be handled by redefining accordingly the priority parameter $\alpha_i^{(k)}$.

covers its polygon $V(\mathscr{C}_i)$, or its farthest vertex $f(V(\mathscr{C}_i))$ is a strict boundary farthest vertex.

Proof. Let s_i exit `SARA` at iteration K_i with its radius set to $r_i^{(K_i)} > 0$ (Notice that the case $r_i^{(K_i)} = 0$ is excluded because s_i belongs to the cover set $S_{\texttt{SARA}}$). According to Algorithm 3, s_i terminated `SARA` execution either because its polygon is not completely covered or because it has reached a strict boundary farthest configuration. We now show that changes in the sensing coverage of other nodes s_j which occur at iteration $k > K_i$ cannot change this property. As this is obvious for sensors which partially cover their polygons, let us consider the case of s_i completely covering its polygon.

Two types of events can occur after the iteration K_i which affect sensor s_i responsibility region: 1) other adjustable sensors s_j reduce their radius, 2) fixed or adjustable sensors are put to sleep. Both these events may result in an increase of sensor s_i responsibility region. However, since s_i radius cannot change (s_i has exited), since the reduction of other nodes radius preserves coverage (Theorem 5.1) and since if a point P is covered it is covered by the node to which responsibility region it belongs to (Theorem 3.1) it derives that s_i responsibility region stays within the circle centered in s_i and with radius equal to $r_i^{(K_i)}$. Therefore, each boundary farthest point at iteration K_i is still a boundary farthest at the end of `SARA` execution. ∎

According to `SARA`, each sensor pursues an individual utility that is to reduce its power consumption and at the same time to do its best to cover the AoI. In terms of this utility function, the cover set $S_{\texttt{SARA}}$ obtained by `SARA` starting from $\mathcal{S}$, is Pareto optimal. In fact, it is not possible to increase the utility of a single sensor (i.e., by reducing the sensing range of an adjustable sensor or putting a fixed one to sleep) without decreasing the utility (i.e., increasing the sensing range of an adjustable sensor or waking up a fixed one that was previously sleeping) of at least another sensor in the network.

Theorem 5.6. **(Pareto optimality)** *Given a set $\mathcal{S} = \mathcal{S}_{adjustable} \cup \mathcal{S}_{fixed}$ of sensors, after the execution of* `SARA` *(without the faster termination condition), the produced cover set S_{SARA} is Pareto optimal.*

Proof. In order to prove the Pareto optimality of `SARA` we need to show that there is no action that could improve the utility of a single sensor, i.e. a sensor reduces its radius or goes to sleep, without a reduction of the sensing coverage achieved by $S_{\texttt{SARA}}$.

This property is true for fixed sensors, since all redundant fixed sensors will eventually go to sleep according to the back-off scheme provided by `SARA`. This trivially derives from Theorems 5.3 and 5.4.

In the case of adjustable sensors, consider $s_i \in \mathcal{S}_{\texttt{adjustable}}$, Theorem 5.2 states that under the execution of `SARA` s_i will eventually reach a final configuration decision, while Lemma 5.5 gives a characterization of the final solution, affirming that if s_i completely covers its polygon, s_i is in a strict boundary farthest vertex configuration whereas if s_i covers its polygon only partially, Corollary 3.2 proves that in this case s_i cannot reduce its radius without affecting coverage. ∎

Pareto optimality is a *necessary condition for global optimality.* Unfortunately, the Pareto optimality of the cover set does not have implications in terms of quality of the solution to the lifetime problem, as there are infinite Pareto optimal solutions. Nevertheless, by adopting an energy-aware policy, SARA is able to choose a cover set among all the possible Pareto-optimal ones, which reduces the energy consumption of the network and prolongs its lifetime, as experimentally shown in Section 7.

6. TWO RECENTLY PROPOSED SELECTIVE ACTIVATION AND RADIUS ADAPTATION ALGORITHMS

To the best of our knowledge there is no prior work in the literature that addresses the problem of selective activation and sensing radius adaptation in a general applicative scenario combining fixed sensors and sensors endowed with variable sensing capabilities. Moreover, previous works rarely consider device heterogeneity. For these reasons, we compare SARA to the Distributed Lifetime Maximization (DLM) algorithm [Kasbekar et al. 2009] which is designed to work with fixed radius sensor and to the Variable Radii Connected Sensor Cover (VRCSC) algorithm [Zou et al. 2009] which is designed to work only with devices that can adjust their sensing radius. The choice of these two algorithms is motivated by the performance analysis carried out by the same authors which shows that DLM and VRCSC achieve better performance with respect to previous schemes proposed in the same class for which they are designed.

In this section we give a short description of DLM and VRCSC and of our extensions to adapt them for a general scenario. We also discuss why they do not provide Pareto optimal solutions.

DLM addresses the problem of activating a subset of sensors so that each point of the AoI is monitored by at least k sensors [9]. DLM considers the case of heterogeneous sensors with fixed sensing radii. The authors call *intersection point* any point where two sensing circles intersect with each other and observe that if each intersection point is k-covered, then the whole AoI is k-covered. DLM is a round based algorithm. At each round, maximum coverage is obtained by iteratively waking up sensors according to an ordered list of nodes that are in radio proximity. The list is sorted on the basis of the energy consumed by the nodes and of the number of intersection points that they can cover. Such a list provides the priority order for the iterative waking up of the sensors in a neighborhood. At each iteration, the sensors whose sensing range is already k-covered by other already awake sensors are removed from the list (they will not wake up). We refer to [Kasbekar et al. 2009] for the details of the algorithm.

We extend DLM to the case of sensors with adjustable sensing radii by considering the devices with variable radii as if they were fixed. This means that each sensor, independently of the class to which it belongs, will either wake up (i.e., operate at maximum transmission radius) or go to sleep. As DLM is not designed to deal with variable radii devices, this variant is introduced only to show that to apply DLM to a more general setting requires non trivial changes.

[9] For the sake of simplicity, in this paper we do not address the problem of k-coverage. Hence in all our experiments, detailed in Section 7, we assume that all the algorithms work with $k = 1$.

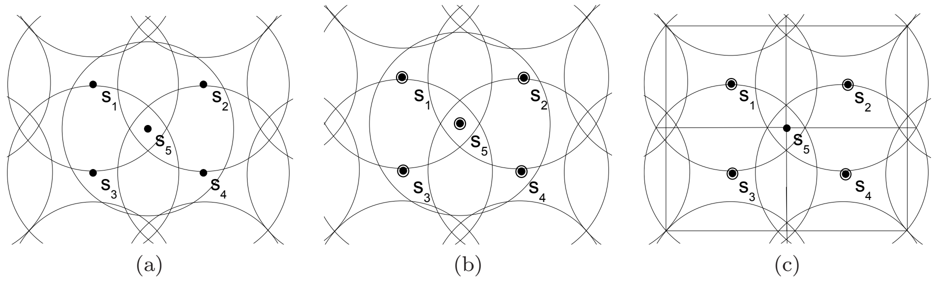


Fig. 6. About Pareto optimality. Initial configuration (a). Selective activation with DLM (b) and SARA (c). The nodes with double circle are awake, while the other ones are sleeping.

VRCSC explicitly addresses the problem of k-covering the AoI with sensors with adjustable radii (both transmission and sensing radii).

VRCSC makes use of Voronoi diagrams to determine which sensors are completely redundant. It then reduces the radius of each sensor to the minimum necessary to cover the farthest point of its Voronoi polygon. For each redundant sensor s, VRCSC calculates the energy benefit obtained by putting it to sleep. This benefit is compared to the additional energy expenditure that the neighbors of s would incur to enlarge their radius with respect to their minimum setting (i.e. the one needed to cover their Voronoi polygon) so as to cover the Voronoi polygon of s on its behalf. We refer the reader to [Zou et al. 2009] for more details on VRCSC.

We extend the use of VRCSC to the case of sensors with fixed radii. In the case of fixed sensors VRCSC only operates the wake up/put to sleep decisions, while the rules to reduce sensor radius are disabled. The purpose of this variant is to show how trivial extensions of VRCSC perform in a more general scenario than the one for which it is designed.

We recall that Pareto optimality is a necessary condition for global optimality, as we discussed in Section 5. Unlike our approach, both DLM and VRCSC do not produce Pareto optimal solutions. This is explained in Figures 6 and 7.

Figure 6(a) represents an initial configuration with all fixed sensors. Observe that sensors s_1, s_2, s_3 and s_4 must be awake to ensure complete coverage of the AoI, as they cover portions of the AoI that cannot be covered by any other sensor in the network. According to DLM, if the energy available to sensor s_5 is sufficiently high, s_5 can be the first sensor to be woken up in its neighborhood. In this case it stays awake despite subsequent wake up of the other four sensors makes s_5 unnecessary (see Figure 6(b)).

Under the same initial setting SARA would not activate s_5, as the backoff policy ensures that all redundant sensors are put to sleep. This is shown in Figure 6(c).

Figure 7 displays a scenario with all adjustable sensors having equal sensing capabilities. Figure 7(a) shows the initial configuration where all sensors are awake and work at their maximum radius. The figure also highlights the Voronoi diagram of the considered sensors. In this example all sensors (s_1, s_2, s_3, s_4 and s_5) are needed to achieve full coverage. Sensors s_1, s_2, s_3 and s_4 cannot reduce their radius as their uniquely covered zone reaches the boundary of their sensing circle. Sensor s_5, instead, can significantly reduce its radius without affecting coverage.

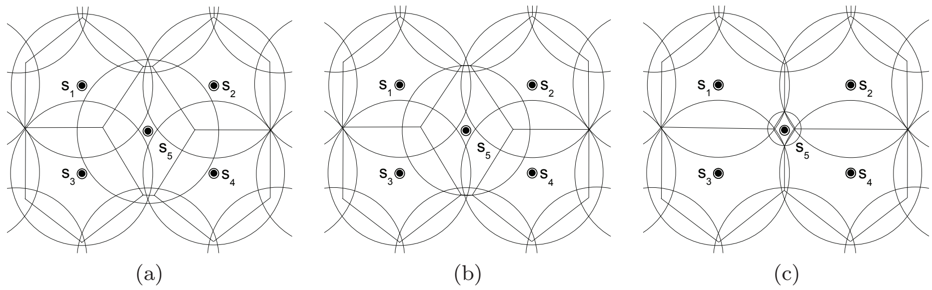


Fig. 7. About Pareto optimality. Initial configuration (a). Selective activation with VRCSC (b) and SARA (c).

According to VRCSC each sensor sets its radius to the distance from it to the farthest vertex of its Voronoi polygon. Therefore, s_5 reduces its radius as shown in Figure 7(b). Since no sensor can be put to sleep, this is the final configuration achieved by VRCSC. Nevertheless, sensor s_5 can still significantly reduce its radius. By iteratively adjusting the radius of s_5, SARA reaches a Pareto optimal configuration, where the radius of s_5 is set to the minimum value that does not leave any coverage hole, as shown in Figure 7(c).

We conclude this subsection by underlying that if DLM and VRCSC are not properly extended as discussed above, VRCSC cannot be used in the case of non adjustable radii and, vice-versa, DLM cannot be applied to the case of variable radii. Our algorithm, instead, works in both the operative settings. Moreover, our algorithm is also able to work in a mixed scenario characterized by both sensors with adjustable and fixed radii, even in the presence of heterogeneous devices, showing an impressive versatility.

We summarize the features of the three schemes in Table I.

	Fixed type		Adjustable type		Both types	
	Hom.	Het.	Hom.	Het.	Hom.	Het.
DLM	Y	Y	N	N	N	N
VRCSC	N	N	Y	N	N	N
SARA	Y	Y	Y	Y	Y	Y

Table I. Scenarios where the considered algorithms are applicable.

To give a fair performance comparison, in Section 7 we compare SARA to DLM and VRCSC in their restrictive operative settings and then we extend their use to the general applicative scenario where devices belong to both the two classes of sensors with fixed and adjustable radii and are heterogeneous in their sensing capabilities. We will show that SARA achieves significant performance improvements over the other two schemes in all operative settings, including the ones for which they are specifically designed.

7. EXPERIMENTAL RESULTS

7.1 Experimental setting

The three algorithms SARA, VRCSC and DLM were implemented using the Wireless module of the OPNET modeler software [OPNET Technologies]. In our experiments we use the following settings. The AoI is a square shaped region of size 80m $\times$ 80m where a number of sensors in the range $[100, 1000]$ are randomly and uniformly deployed. Each sensor node has a transmission range of 30m and an energy consumption when transmitting, receiving, when in idle and asleep mode which follows the TelosB [Polastre et al. 2005] energy model. The battery capacity is 1840 mAh. Sensor nodes are endowed with an initial energy that is uniformly distributed in the interval (0, 1840mAh]. The sensing radius varies from 2m to 6m, depending on the operative scenario. Sensors have been modeled according to the datasheets of Maxbotix sonar devices [MAXBOTIX sonar datasheets 2010] which work at 2Hz and have different orientations. For such sensors the increase in energy consumption with distance increases according to a cubic law ($c = 3$ in Equation 1).

The length of the operative time interval between two successive executions of SARA and DLM is set to 24hrs which is equal to 1.5% of the total time a sensor can remain awake before fully depleting its battery. The algorithm VRCSC instead reconfigures the network every time a sensor has exhausted its available energy, as specified in [Zou et al. 2009].

In all the experiments we adopted the faster termination condition for SARA and set a limit $K = 20$ to the number of algorithm iterations. Our extensive experimentation show that SARA naturally terminates before reaching this threshold in about 95% of the experiments. In order not to overburden the exposition, here we omit the study of the effects on performance of varying K and the way the decision priority $\alpha_i^{(k)}$ (described in Section 4.1.2) is computed; an extensive discussion of these experiments is given in [Bartolini et al. 2010].

7.2 Experiments with only adjustable sensors

7.2.1 *Homogeneous adjustable sensors.* This section is devoted to a comparative analysis of the performance of SARA and of the extended versions of DLM and VRCSC in a scenario with 900 homogeneous sensors with adjustable radius. All such sensors have the same capability to adjust their sensing radius in the interval [2m, 6m]. It should be noted that this scenario is the one for which VRCSC was specifically designed. It is therefore quite impressive that SARA significantly outperforms VRCSC even in this case.

Indeed, VRCSC is not able to fully exploit the adaptability of the sensing range as SARA does thanks to the use of Voronoi diagrams in the Laguerre geometry.

Figure 8(a) shows how the coverage achieved by the three algorithms decreases with time. The loss in coverage under DLM is much faster than under VRCSC and SARA, showing that DLM cannot be trivially extended to operate in this scenario. SARA significantly outperforms VRCSC: For instance, after 350 days of operations, SARA is able to cover about twice the extension of the area covered by VRCSC. This shows the ability of SARA to prolong the network lifetime when this is formulated as the time within which the network is still capable to cover a given percentage of the AoI, while working at maximum coverage extension.

Figures 8(b) and (c) show how the percentage of awake and sleeping sensors varies with time. These percentages are computed with respect to the whole set of available sensors. Although DLM keeps awake only a very small percentage of the available sensors, it is penalized by the fact that the radius of the awake sensors cannot be modulated by the algorithm (Figure 8(e)). Hence, the energy consumption per sensor is very high, as demonstrated by Figure 8(f) which shows that in DLM the residual energy is quite low after a few operative time intervals, resulting in a very high percentage of dead sensors[10].

Notice that, under DLM, the number of awake sensors (Figure 8(b)) shows a peak after about 60 days. At the beginning of the network operation all the sensors experience a similar consumed energy. When this is the case, according to DLM the value of the waking up priority is dominated by the number of intersection points that each sensor can cover. Therefore, initially DLM is able to cover the AoI with a very low number of awake sensors. As time increases, the sensor energy consumption starts differing, which makes DLM privilege the energy consumption criterion when deciding which sensors to wake up. This is not usually the best choice in terms of redundancy reduction, therefore more and more sensors quickly deplete their energy. The decrease in the percentage of awake sensors after 60 days (the peak of DLM in Figure 8(b)) is in fact due to the increase in the percentage of dead sensors (shown in Figure 8(d)). The algorithms SARA and VRCSC are able to modulate the sensing radius of the awake sensors to reduce coverage overlaps and save energy. Therefore, with respect to DLM, more sensors are woken up (Figure 8(b)) operating with lower sensing radius (Figure 8(e)). This allows VRCSC and SARA to save more energy than DLM (Figure 8(f)). When comparing SARA and VRCSC we observe that SARA wakes up a higher number of sensors with smaller radius than VRCSC, thus reducing the amount of consumed energy and being able to prolong the network lifetime.

Figure 9(a) shows the network lifetime achieved by the three algorithms, namely the time when the algorithms are no longer able to achieve a coverage $\geq 80\%$ of the AoI when working at maximum extent. Values are displayed for different densities of the sensor nodes, which correspond to a number of sensors ranging from 200 to 1000. Results confirm that SARA outperforms the other two algorithms with network lifetimes which can be over fourfold those of DLM. Although this scenario is the most favorable to the algorithm VRCSC, SARA is able to always achieve a longer lifetime. For instance, when the number of sensors is 1000, the algorithm SARA achieves an increase of 20% in the network lifetime with respect to VRCSC (350 days for SARA versus 290 days for VRCSC).

In order to understand why the comparison of the lifetime achieved by the three schemes shows higher gaps when the number of nodes increase, Figure 9(b) and Figure 9(c) display the percentage of the AoI which is covered by K sensors, for different values of K, when the number of sensors is 300 and 900, respectively. Figure 9(b) clearly shows that when the number of nodes (and the density) is low a significant portion of the area of interest is either uncovered or covered by only few sensors. These sensors deplete their energy faster than others no matter which

[10]Hereby we refer to *dead sensor* as to devices which have fully depleted their available energy.

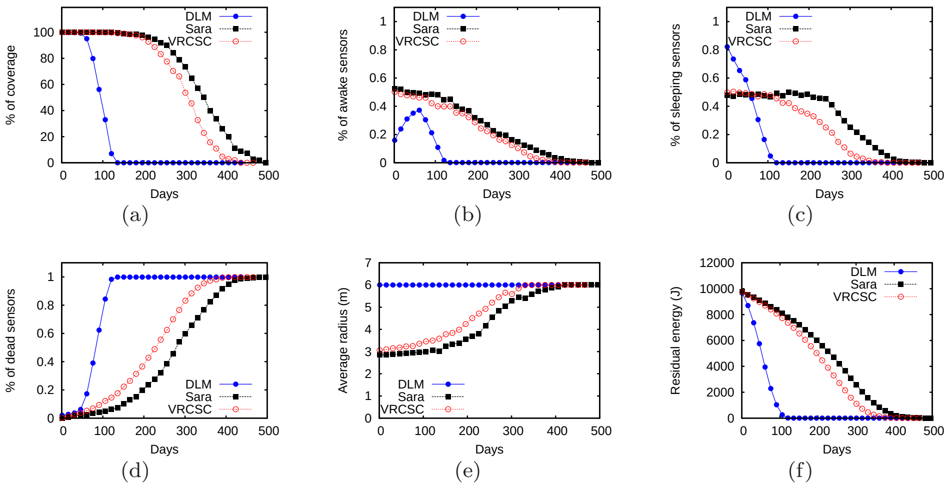


Fig. 8. Adjustable sensors: homogeneous setting. Comparative analysis of `SARA`, `DLM` and `VRCSC`. Percentage of AoI covered (a), percentage of awake sensors (b), percentage of sleeping sensors (c), percentage of dead sensors (d). Average radius of the awake sensors (e) and average residual energy per sensor (f). Scenario with 900 sensors.

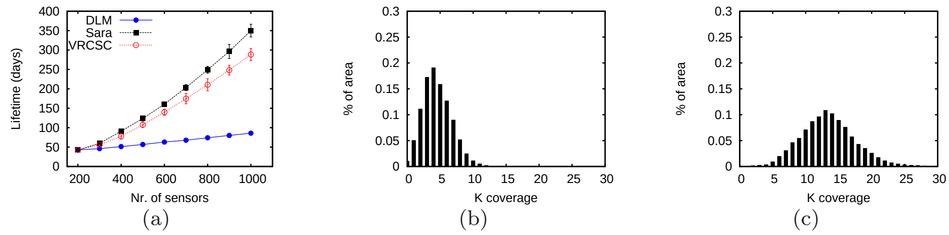
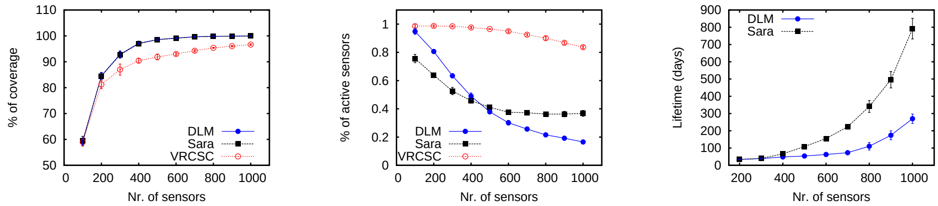


Fig. 9. Adjustable sensors, homogeneous setting. Lifetime (a) and percentage of the AoI which is covered by K sensors, for different values of K, when 300 sensors (b) and 900 sensors (c) are deployed over the AoI.

algorithm is in use. This implies that, with a tolerance of only 20% in coverage loss, the three algorithms cannot do much to improve the network lifetime. For this reason, with 300 sensors, the lifetime of `DLM`, `VRCSC` and `SARA` is about the same, as seen in Figure 9(a).

Figure 9(c) shows the same metric for the case where 900 sensors are deployed over the AoI. Due to the higher density large portions of the AoI are covered by several sensors, giving the opportunity to selective activation and radius adaptation schemes to perform smart choices in order to improve network lifetime. Higher density scenarios are therefore those where `SARA` makes the difference as shown in Figure 9(a). When the number of available sensors is high `SARA` outperforms the other two algorithms by achieving a significant longer lifetime, being able to perform a more efficient activation policy.



(a) (b) (c)

Fig. 10. Adjustable sensors, heterogeneous setting. Percentage of the AoI covered (a), percentage of awake sensors (b) and network lifetime (c). The network lifetime is defined as the time at which coverage falls below 80%.

Notice that, the performance of SARA with respect to the other algorithms is qualitatively similar to the one shown in Figure 9(a), even under other settings of the coverage threshold used to define the network lifetime. For the sake of brevity we omit the figures of the related experiments. The interested reader can find more details regarding these experiments in [Bartolini et al. 2010].

7.2.2 *Heterogeneous adjustable sensors.* In this section we focus on a scenario where adjustable sensors having different sensing capabilities are deployed over the AoI. Each sensor belongs to one of two classes of sensors. Sensors in the first class can vary their sensing range between 2m and 6m. Sensors in the second class can only vary their sensing range between 2m and 3m. Sensors are equally splitted among the two classes.

Figures 10(a) and (b) show the percentage of the AoI covered and the percentage of awake sensors after the first algorithm execution (day 0), at the beginning of the first operative time interval. Despite all sensors have significant residual energy VRCSC is not able to guarantee maximum coverage of the AoI, even if it wakes up a higher percentage of sensors than the other two schemes (Figure 10(b)). The reason lays in the use of Voronoi diagrams to regulate the extent of the node sensing radius. Voronoi tessellation correctly defines node responsibility regions only in case of homogeneous sensing radii, while in the case of heterogeneous sensors the Voronoi-Laguerre tessellation has to be used to model the problem correctly.

Figure 10(b) shows that, although DLM makes sensors work at full range, it wakes up more sensors than SARA when the number of sensors is small. As the number of deployed sensors grows, SARA wakes up more sensors than DLM but makes them work at much smaller radii. This motivates the fact that SARA outperforms DLM in terms of network lifetime (Figure 10(c)) for all the considered sensor densities. The gap between SARA and DLM is quite significant: when the number of sensors is 900, SARA has a network lifetime which is almost twice the lifetime of DLM. As the coverage percentage achieved by VRCSC is not optimal, it can fluctuate over time. Therefore in this scenario, likewise in any scenario with sensors having heterogeneous sensing ranges, VRCSC is not suitable to solve our problem, which requires the network to work at maximum coverage. Therefore, evaluating the lifetime of VRCSC in terms of time until a given coverage percentage is guaranteed makes no sense. In the following we will focus on the other two algorithms when showing results of experiments that involve heterogeneous sensors.

7.3 Experiments with fixed sensors

7.3.1 *Heterogeneous fixed sensors.* In this section we consider a scenario where sensors have a fixed sensing radius. We focus on the case where sensors have heterogeneous sensing capabilities: Half of the sensors have a sensing radius of 3m while the other half have a sensing radius of 6m. This is the scenario for which `DLM` was specifically designed. In this setting `VRCSC` is not able to guarantee maximum coverage in case of sensor heterogeneity, as we already discussed at the end of Section 7.2.2. Therefore we will display only results for `DLM` and `SARA`.

The experiments show that `SARA` outperforms `DLM` in terms of percentage of the AoI covered over time (Figure 11(a)) and results into a lower number of dead sensors over time (Figure 11(c)). The percentage of awake sensors, displayed in Figure 11(b), shows a similar trend (for the same reason) than that discussed in Section 7.2.1. `DLM` experiences a higher number of awake sensors than `SARA` during the first 120 days. As a consequence, the number of sensors which are put to sleep (obtained as a complement to 1 of the sum of awake and dead sensors) will be much lower than in `SARA`. When time increases the reduced number of awake sensors in `DLM` reflects the high number of dead nodes, and consequently the poor coverage performance. These observations motivate the fact that `SARA` experiences longer network lifetimes than `DLM`. This improvement is as high as twofold (Figure 11(f)).

Figure 11(d) and (e) shows the percentage of active sensors with large and small radius under the execution of `DLM` and `SARA`, respectively. It is interesting to note that initially `DLM` activates the same percentage of sensors with small and large radius. As a consequence, nodes with large radius quickly deplete their energy, and after day 100, `DLM` can only work with sensors having small radius. Nevertheless `SARA` is able to successfully exploit device heterogeneity from the beginning, by activating sensors with large and small radius in different percentages, on the basis of coverage requirements. As a consequence, only at day 200 `SARA` works with only sensors having small radius. For this reason the peak in Figure 11(b) in the number of active sensors is located on the right with respect to the one of `DLM`.

7.4 Mixed scenario: adjustable and fixed sensors

7.4.1 *Scenario A.* We consider an operative scenario with 900 uniformly deployed sensors. The 50% of the available sensors is composed by fixed sensors with sensing radius equal to 6m, while the remaining 50% consists in devices with adjustable sensing radius varying in the interval [2m,6m].

Figure 12(a) shows the percentage of the AoI covered by `SARA` as time increases. The figure also shows the percentage of the AoI that is covered by the only sensors with adjustable radius, and by those with fixed radius, separately. It is worth noting that at the first operative time intervals `SARA` privileges the sensors with adjustable radius in the cover set, as also detailed in Figure 12(b). This is due to the higher flexibility of this class of devices. As time progresses, the adjustable sensors that have been used extensively in the previous intervals deplete their energy, thence `SARA` requires more fixed sensors to be included in the cover set. It should be noted also that the percentage of dead sensors is about the same for the sensors of the two classes, see Figure 12(c). This is due to the fact that, in this setting, as long

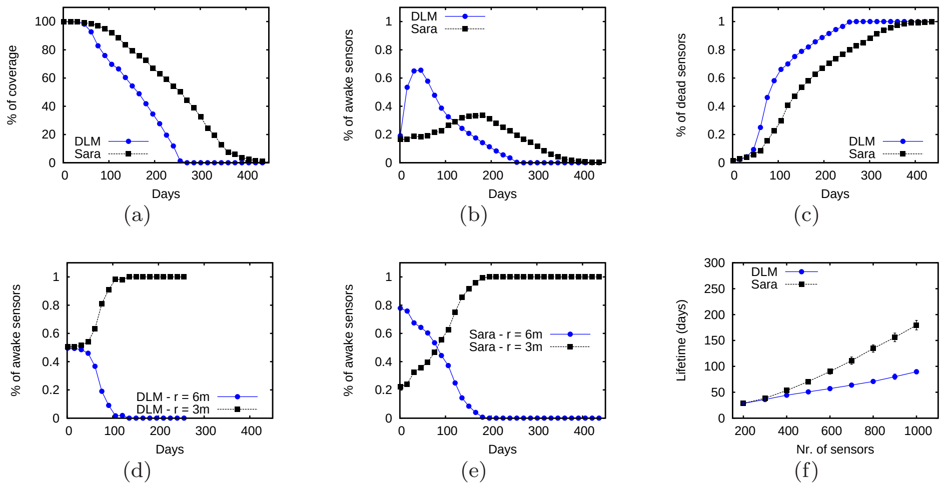


Fig. 11. Fixed sensors, heterogeneous setting. Scenario with 900 sensors. Comparative analysis of SARA and DLM. Percentage of AoI covered (a), percentage of awake sensors (b), percentage of dead sensors (c). Composition of the set of awake sensors under SARA (d) and DLM (e). Network lifetime (f).

as a fixed sensor is woken up, it consumes energy at the same rate of an adjustable sensor working at maximum sensing range. This evidences the capability of SARA to exploit the residual energy of the two classes of sensors in an equal manner, when the two classes have equal sensing capabilities. Figure 12(d) shows the composition of the set of sleeping sensors complementing the information given by Figures 12(b) and (c).

Notice that both VRCSC and DLM (in their modified versions) consider the set of sensors as if it were composed by all adjustable and all fixed sensors, respectively. As the maximum radius of adjustable devices is equal to the radius of the fixed devices, both VRCSC and DLM work in this scenario as if the set of sensors were homogeneous. Therefore, VRCSC is able to work in this scenario without creating coverage holes. Nevertheless, the presence of fixed sensors in the available set, compromises the capability of VRCSC to correctly determine the maximum extent of the radius reduction to be adopted by sensors with adjustable range, because this reduction is calculated as if the fixed sensors were also able to reduce their radius. This is evident in Figure 12(e) which shows the lifetime of the network when the percentage of fixed sensors in the available set increases. Not surprisingly, the performance of DLM is not affected by this increase as it treats the two classes alike. Both VRCSC and SARA show a decreasing behavior of the network lifetime due to the decreasing flexibility of the network. Indeed, it is intuitive that by increasing the percentage of fixed sensors, we are significantly reducing the set of possible solutions that can be reached by any algorithm. Nevertheless, SARA exploits the capabilities of the two classes of sensors with more specifically tailored decisions.

All the above considerations justify the significant improvement in terms of lifetime achieved by SARA with respect to DLM and VRCSC. To highlight this difference

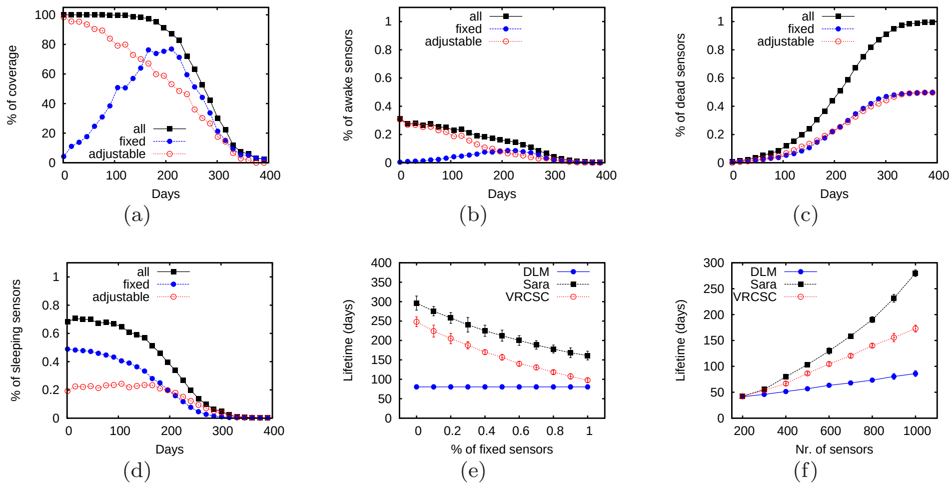


Fig. 12. Mixed sensors: scenario A. The maximum radius of adjustable sensors and the radius of fixed sensors are 6m. Case of 900 sensors: 50 % fixed and 50 % adjustable range. Coverage of the AoI (a), awake sensors (b), dead sensors (c), and sleeping sensors (d). Lifetime of the network by varying the percentage of fixed sensors with respect to total (e). Lifetime of the network by varying the number of available sensors (50 % of each class) (f).

we now consider another experiment conducted by varying the number of available sensors, always with a mix of half fixed and half adjustable sensors. The Figure 12(f) illustrates the behavior of the three algorithms in this setting. For instance, when the number of sensors is 1000, `SARA` achieves a lifetime of 280 days, whereas `VRCSC` reaches 170 days, and `DLM` only 80 days.

7.4.2 *Scenario B.* In this latter subsection, we consider an operative scenario where sensors belong to both classes and where the radius of fixed sensors is 3m, while the radius of adjustable sensors varies in the interval [2m, 6m]. Notice that the maximum radius of adjustable sensors is twice as long as the radius of fixed sensors. Therefore, `VRCSC` is not able to guarantee maximum coverage extension in this scenario for the reasons discussed at the end of Section 7.2.2.

The qualitative analysis of the results shown in Figures 13(a-f) is analogous to the case of the homogeneous setting. Nevertheless, Figure 13(c) highlights that, in this case, the set of dead sensors is composed by a higher fraction of adjustable sensors with respect to Scenario A. This is due to the fact that we are considering fixed sensors with lower range, that implies for this class a lower energy consumption rate with respect to Scenario A, resulting in a higher residual energy for the fixed sensors, as shown in Figure 13(e). Finally, in Figure 13(f) we show that, as expected, the lifetime achieved by `SARA` is significantly longer than under `DLM`. For instance, when the number of sensors is 1000, `SARA` achieves a lifetime of about 750 days, while `DLM` is only capable to last 270 days.

Notice that in this setting, it does not make sense to analyze the performance of the algorithms when the percentage of the two classes of sensors varies as we did in Section 7.4.1. This is because the fixed sensors have lower sensing radius than

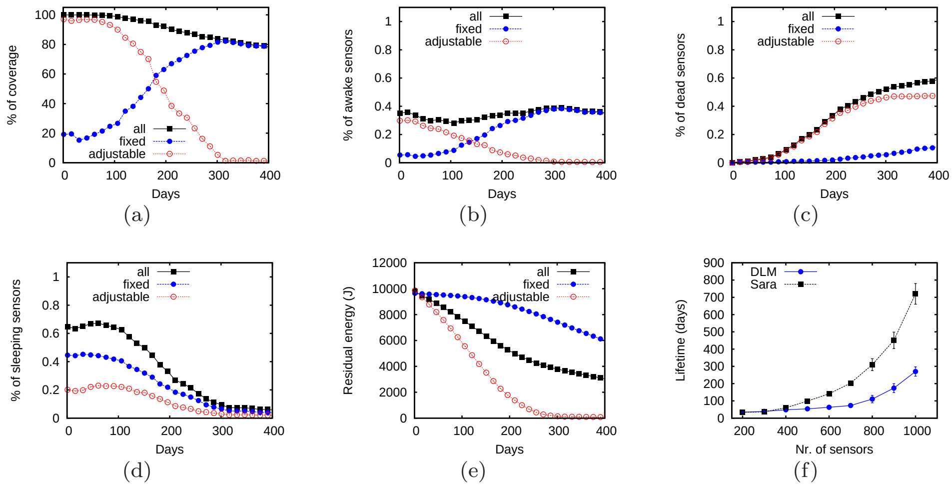


Fig. 13. Mixed sensors: scenario B. The maximum radius of the adjustable sensors is 6m while the radius of fixed sensors is 3m. Case of 900 sensors: 50% fixed and 50% adjustable. Coverage of the AoI (a), awake sensors (b), dead sensors (c), sleeping sensors (d) and residual energy (e). Lifetime of the network when varying the number of sensors (50 % fixed, 50% adjustable sensors).

the maximum radius of adjustable sensors. Therefore, by varying the composition of the mix we would alter the coverage capability of the network.

8. RELATED WORK

The problem of exploiting the high density of sensor networks to prolong network lifetime has been investigated in the literature with different flavors and approaches. Depending on the application requirements, the approach to the problem may vary significantly. Some solutions, such as SPAN [Chen et al. 2002] and ASCENT [Cerpa and Estrin 2004], to mention some of the most acknowledged, focus on how to guarantee network connectivity over time. Due to space limitations, in this section we consider only previous papers dealing with the problem of ensuring coverage of the area of interest. The interested reader can refer to [Rowaihy et al. 2007] for a survey of sensor scheduling policies in several other applicative scenarios.

The PEAS protocol proposed in [Ye et al. 2003] was designed to address both coverage and connectivity at the same time. According to this protocol only a subset of nodes stay awake at each time while the others are put to sleep. A sleeping node occasionally wakes up to determine the presence of coverage holes in its proximity and makes waking up decisions accordingly. This approach does not ensure complete coverage, as coverage holes cannot be discovered until a nearby sleeping sensor wakes up. A randomized algorithm is proposed in [Xiao et al. 2010]. Different sets of sensors work alternatively according to a probabilistic scheduling. The authors study the performance of the proposed approach in terms of coverage extension and detection delay. Differently from the works in [Ye et al. 2003; Xiao et al. 2010], our approach aims at ensuring the coverage completeness as long as the available sensors have enough energy.

The protocol CPP proposed in [Xing et al. 2005] aims at achieving k-coverage of an area of interest while maintaining network connectivity. The authors define an operative setting in which the transmission radius is at least twice the sensing range. This means that coverage implies connectivity. They also provide necessary and sufficient conditions for an area to be k-covered. The geometric analysis made in [Xing et al. 2005] is at the basis of several follow up solutions including `DLM` [Kasbekar et al. 2009].

In [Cardei and Du 2005] sensors are divided into disjoint sets. At a specific time only one sensor set is awake while the sensors of the other sets are kept in a low power mode. The sets are scheduled in a round robin manner and operate for equal time intervals. The authors prove that finding the maximum number of disjoint sets is an NP-complete problem. They propose a heuristic to calculate the set covers which is based on a mixed integer programming model. The main drawback of this approach is that it is centralized, which is not desirable in a sensor network environment. The constraint of having disjoint set covers operating for equal time intervals is relaxed in the work [Cardei et al. 2005] where two heuristics are proposed, one using linear programming and the other using a greedy approach.

In [Funke et al. 2007], the authors consider the problem of selecting a set of awake sensors of minimum cardinality so that sensing coverage and network connectivity are maintained. The authors analyze the performance of a greedy solution for complete coverage showing that it achieves an approximation factor no better than $\Omega(\log n)$, where n is the number of sensor nodes. The authors then present an algorithm that provides approximate coverage while ensuring that the number of awake nodes is within a constant factor from the optimum.

The same problem is addressed in [Tian and Georganas 2002] and in [Bulut et al. 2008]. These papers consider the coverage problem. The objective is to wake up a minimal number of sensors while the others conserve their energy in a low power mode. Each sensor periodically evaluates its sensing area to determine whether it is also covered by other sensors. A redundant sensor goes to sleep. Since several sensors may determine that they can go to sleep at the same time, a back-off based policy is proposed to prevent conflicting decisions and impose an order to go to sleep. These proposals are similar to the way our algorithm eliminates the redundancies in the case of sensors endowed with fixed sensing capabilities. Nevertheless, the way we give priority to sensors having higher overlaps is completely different, as it is based on a more refined evaluation of the energy gain that can be obtained by putting to sleep individual sensors.

None of the aforementioned works addresses the problem of providing maximum coverage extension of an area of interest with some or all sensors being able to modulate their sensing ranges as we do in this paper. This operative setting, but with discrete coverage targets, is analyzed in [Cardei et al. 2006]. The proposed solution is based on non-disjoint set cover scheduling. The approach is centralized and the problem is proven to be NP-complete.

Two recent papers by Kasbekar et al. [Kasbekar et al. 2009], and by Zou et al. [Zou et al. 2009], propose the algorithms `DLM` and `VRCSC`, respectively. These algorithms are described in deeper details in Section 6 and experimentally compared to our proposal in Section 7.

9. CONCLUSIONS

We propose a new algorithm for prolonging the lifetime of a heterogeneous wireless sensor network (WSN) through selective Sensor Activation and sensing Radius Adaptation (SARA). Our approach is very general and is the first to be applicable to a mixed scenario combining devices with adjustable and fixed sensing radii. We prove that SARA achieves maximum sensing coverage and leads to network configurations which are Pareto optimal, namely it is not possible to further decrease the radius of any adjustable sensor or putting to sleep any fixed sensor without compromising coverage.

A thorough simulation based performance comparison shows that SARA significantly outperforms the most acknowledged solutions so far proposed, in terms of coverage and percentage of alive nodes over time, not only in the mixed scenario but also in the scenarios for which previous solutions have been specifically designed. In addition, SARA more evenly drains energy from devices belonging to different classes which is a key aspect to ensure fairness when the two classes show equal coverage capabilities.

REFERENCES

Ammari, H. and Das, S. 2008. Promoting heterogeneity, mobility, and energy-aware voronoi diagram in wireless sensor networks. *IEEE Trans. on Parallel and Distributed Systems 19,* 7.

Bartolini, N., Calamoneri, T., La Porta, T., Massini, A., and Silvestri, S. 2009. Autonomous deployment of heterogeneous mobile sensors. *Proc. of IEEE ICNP*.

Bartolini, N., Calamoneri, T., La Porta, T., Petrioli, C., and Silvestri, S. 2010. Sensor activation and radius adaptation (SARA) in heterogeneous sensor networks. *Technical Report, http://arxiv.org/abs/1009.6098*.

Basagni, S., Carosi, A., and Petrioli, C. 2008. Mobility in wireless sensor networks. In *Algorithms and Protocols for Wireless Sensor Networks*, A. Boukerche, Ed. Wiley Series on Parallel and Distributed Computing. John Wiley & Sons, Inc., Hoboken, NJ, Chapter 10, 267–305.

Bough, D. and Santi, P. 2002. Investigating upper bounds on network lifetime extension for cell-based energy conservation techniques in stationary ad hoc networks. *Proceedings of ACM Mobicom*.

Bulut, E., Wang, Z., and Szymanski, B. K. 2008. Cost-quality tradeoff in cooperative sensor networking. *Proc. of IEEE International Conference on Communication (ICC)*, 112–117.

Cardei, M. and Du, D. May 2005. Improving wireless sensor network lifetime through power-aware organization. *ACM/Springer Wireless Networks*, 333–340.

Cardei, M., Thai, M., Li, Y., and Wu, W. 2005. Energy-efficient target coverage in wireless sensor networks. *Proc. of INFOCOM*.

Cardei, M., Wu, J., and Lu, M. 2006. Improving network lifetime using sensors with adjustable sensing ranges. *Int. J. Sen. Netw. 1,* 1/2, 41–49.

Cerpa, A. and Estrin, D. 2004. Ascent: Adaptive self-configuring sensor networks topologies. *IEEE Transactions on Mobile Computing 3,* 3, 272–285.

Chen, B., Jamieson, K., Balakrishnan, H., and Morris., R. 2002. Span: an energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. *ACM Wireless Networks 8,* 5, 481–494.

Delman, C. and Galperin, G. 2003. A tale of three circles. *Mathematics magazine 76,* 1, 15–32.

Funke, S., Kesselman, A., Kuhn, F., Lotker, Z., and Segal, M. 2007. Improved approximation algorithms for connected sensor cover. *ACM/Springer Wireless Networks 13,* 2, 153–164.

Gu, L. and Stankovic, J. A. 2004. Radio-triggered wake-up capability for sensor networks. *Proceedings of the 10th IEEE Real-Time and Embedded Technology and Applications Symposium*, 27.

IMAI, H., IRI, M., AND MUROTA., K. 1985. Voronoi diagram in the Laguerre geometry and its applications. *SIAM J. Comput. 14,* 1, 93–105.

KANSAL, A., HSU, J., ZAHEDI, S., AND SRIVASTAVA, M. B. 2007. Power management in energy harvesting sensor networks. *ACM Transactions on Embedded Computing Systems 6,* 4.

KASBEKAR, G. S., BEJERANO, Y., AND SARKAR, S. 2009. Lifetime and coverage guarantees through distributed coordinate-free sensor activation. *Proceedings of ACM Mobicom*.

KOMPIS, C. AND ALIWELL, S. last accessed on April 14, 2010. Energy harvesting technologies to enable wireless and remote sensing. *Sensors and Instrumentation KTN Action Group Report, 2008, London, UK, available at http://www.costiskompis.com/pubs/pdf/energyharvesting08.pdf* .

MAXBOTIX SONAR DATASHEETS. Last accessed on April 14, 2010. *http://www.maxbotix.com/uploads/LV-MaxSonar-EZ1-Datasheet.pdf* .

MOSER, C., THIELE, L., BRUNELLI, D., AND BENINI, L. 2010. Adaptive power management for environmentally powered systems. *IEEE Transactions on Computers 59,* 4, 478–491.

NAKAMURA, E. F., LOUREIRO, A. A. F., AND FRERY, A. C. 2007. Information fusion for wireless sensor networks: Methods, models, and classifications. *ACM Computing Surveys 39,* 3 (August).

OPNET TECHNOLOGIES. http://www.opnet.com/.

OSIRIS PHOTOELECTRIC SENSORS. Last accessed on January 9, 2010. *http://www.schneider-electric.co.uk/electricity.nsf/automation-control/detection-equipment/photo-electric-devices*.

PATTEM, S., PODURI, S., AND KRISHNAMACHARI, B. 2003. Energy-quality tradeoffs for target tracking in wireless sensor networks. *Proc. of ACM International Conference on Information Processing in Sensor Networks (IPSN)*.

POLASTRE, J., SZEWCZYK, R., AND CULLER, D. E. 2005. Telos: enabling ultra-low power wireless research. *Proceedings of IEEE IPSN*, 364–369.

ROWAIHY, H., ESWARAN, S., JOHNSON, M., VERMA, D., BAR-NOY, A., BROWN, T., AND PORTA, T. L. 2007. A survey of sensor selection schemes in wireless sensor networks. *Proc. of SPIE, Volume 6562 – Unattended Ground, Sea, and Air Sensor Technologies and Applications IX*.

SHARMA, V., MUKHERJI, U., JOSEPH, V., AND GUPTA, S. 2010. Optimal energy management policies for energy harvesting sensor nodes. *IEEE Transactions on Wireless 9,* 4, 1326–1336.

TIAN, D. AND GEORGANAS, N. D. 2002. A coverage-preserving node scheduling scheme for large wireless sensor networks. *Proc. of ACM WSNA*, 32–41.

WANG, G., CAO, G., AND LA PORTA, T. 2006. Movement-assisted sensor deployment. *IEEE Trans. on Mobile Computing 6*, 640–652.

XIAO, Y., CHEN, H., WU, K., SUN, B., ZHANG, Y., SUN, X., AND LIU, C. 2010. Coverage and detection of a randomized scheduling algorithm in wireless sensor networks. *IEEE Transactions on Computers 59*, 507–521.

XING, G., WANG, X., ZHANG, Y., LU, C., PLESS, R., AND GILL, C. 2005. Integrated coverage and connectivity configuration for energy conservation in sensor networks. *ACM Transactions on Sensor Networks (TOSN) 1,* 1, 36–72.

YE, F., ZHONG, G., CHENG, J., LU, S., AND ZHANG, L. 2003. Peas: A robust energy conserving protocol for long-lived sensor networks. *Proceedings of IEEE ICDCS*.

YICK, J., MUKHERJEE, B., AND GHOSAL, D. 2008. Wireless sensor networks survey. *Elsevier Computer Networks 52,* 12, 2292–2330.

ZOU, Z., DAS, S., AND GUPTA, H. 2009. Variable radii connected sensor cover in sensor networks. *ACM Trans. on Sensor Networks 5,* 1 (February), 8–36.