Route in Mobile WSN and Get Self-Deployment for Free

Eric Fleury, Kévin Huguenin, Anne-Marie Kermarrec

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Route in Mobile WSN and Get Self-Deployment for Free

Kévin Huguenin — Eric Fleury — Anne-Marie Kermarrec

N° 6819

January 2008

Thème COM

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--6819--FR+ENG

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

INRIA

centre de recherche
RENNES - BRETAGNE ATLANTIQUE

Route in Mobile WSN and Get Self-Deployment for Free

Kévin Huguenin , Eric Fleury , Anne-Marie Kermarrec

Thème COM — Systèmes communicants
Équipe-Projet ASAP

Rapport de recherche n° 6819 — January 2008 — 21 pages

Abstract: We consider a system composed of a set of mobile sensors, disseminated in a region of interest, which mobility is controlled (as opposed to mobility imposed by the entity on which they are embedded). A routing protocol in this context enables any point of the region to be reached. In this paper we present, Grasp, a GReedy stAteless Routing Protocol for mobile wireless sensor networks (WSN). The strength of Grasp, beyond its simplicity, is that routing enables a free and close to optimal self-deployment of sensors over a given region. Grasp transparently copes with dynamic changes of the region of interest. In addition, Grasp is independent from the underlying communication model. Grasp ensures *(i)* that routing is always possible in a mobile WSN irrespective of the number of sensors and *(ii)* above a given number of sensors in a considered zone, the protocol eventually ensures that routing does no longer require sensors to move, thus providing self-deployment. In one dimension, Grasp converges to a full connected-coverage of the region with the minimum required number of sensors in a finite number of steps, ensuring an optimal deployment. In two dimensions, sensors reach autonomously a stable full coverage following geometrical patterns. This requires only 1.5 the optimal number of sensors to cover a region. A theoretical analysis of convergence proves these properties in one and two dimensions. Some simulation results matching the analysis are also presented.

Key-words:

Centre de recherche INRIA Rennes – Bretagne Atlantique
IRISA, Campus universitaire de Beaulieu, 35042 Rennes Cedex
Téléphone : +33 2 99 84 71 00 — Télécopie : +33 2 99 84 71 71

Routage et déploiement spontané dans les réseaux de capteurs mobiles sans fil

Résumé :

Mots-clés :

1 Introduction & Background

Context Over the last decade, the emergence of wireless sensor devices has made possible the design of networked distributed systems able to interact with the physical world. This provides the ground for real-world applications ranging from intrusion detection to environment monitoring. The deployment of static sensors is a key point in the design of such systems since it determines their *coverage* and *connectivity*. Coverage refers to their ability to interact with the physical world while connectivity refers to the ability to route information between the entities forming the network. In the context of actuator networks [16], the first characterizes the global interaction range of the whole network and the latter its controllability. A large amount of research has focused on finding an optimal deployment ensuring both full coverage and full connectivity in static Wireless Sensor Network (WSN) [1]. Though full connectivity together with full coverage ensures that there exists a path between any source node and any point of the region, finding a path from the source to the destination – referred as *routing* – remains a difficult problem in a mesh network of low-capability devices with respect to memory and computational capabilities [14, 4].

Background Recent technological advances in the area of robotic sensors have led to the development of mobile WSNs. They provide a dynamic and flexible framework for a much wider application spectrum than static WSNs, ranging from adaptive sampling where the nodes may move to perform sampling depending on the application needs [6], to self-deployment [5]. More specifically, both the coverage and connectivity issues vanish as soon as a node is able to move in order to reach any point of the region. However, due to the limited amount of energy available at each node, mobility is often seen as a way to make a system autonomous and unmanned, a typical application being self-deployment. As opposed to static sensor networks, mobile WSN do not rely on preliminary computation for sensor placement. Yet, the limited and intermittent connectivity of the network makes the design of distributed algorithms more challenging. This paper tackles the problem of leveraging nodes mobility for the purpose of routing and self-deployment of a mobile WSN.

Most of research on routing in mobile networks has focused so far on putting up with mobility rather than controlling it. Such works deal with *uncontrolled* mobility of the nodes themselves, exploiting *uncontrolled* external moving entities [8, 21] – referred as *message ferrying*. Yet, there has been very few works on leveraging sensors mobility for the purpose of routing. In [20], Zhao *et al.* advocate the use of ferries to ensure delay-tolerant connectivity in a network of uncontrolled mobile nodes and propose an algorithm for computing ferries' path to meet the traffic demand while minimizing the data delivery delays. Given the bandwidth requirements of each node in the network, the ferries' route minimizing the delivery delay under bandwidth constraints is computed using linear programming. In addition, they assess the problem of ferry synchronization in the case where there is no static node to ensure delay-tolerant relaying between ferries.

Besides, a large amount of research has focused on finding an optimal deployment ensuring both full coverage and full connectivity in static WSNs. The resulting deployment schemes are based on regular meshes but the elementary pattern depends on the ratio of the communication radius r_c and the sensing radius r_s. In [1], Bai *et al.* propose a deployment pattern to achieve both full coverage and connectivity and proved its optimality for all values of r_c/r_s. Recently, movement-assisted deployment has received great attention leading to efficient distributed techniques based on connected coverage extension [17], potential virtual forces [22] or hole detection [19]. Such methods achieve optimal [18] or near-optimal node positioning with respect to a variety of optimization criterion such as the total distance covered or the number of moves. Yet, all these techniques assume a full knowledge of the ROI at each node.

Contributions In this paper, we consider a system composed of a set of mobile sensors whose mobility is controlled (as opposed to mobility imposed by the entity on which they are embedded). We address the problem of *on-demand sensing* in this context. A routing protocol should enable any point of the region to be reached for that purpose. Given a source node and a destination point x of the Region Of Interest (ROI), routing is used to reach *a* node in charge of sensing the destination x. In the process of routing, nodes are able to either forward a message to a neighbor or to move from a point to another for the message to progress toward the destination. We propose an efficient *GReedy stAteless Routing Protocol* (GRASP) for mobile wireless sensor networks based on geographic considerations. GRASP is fully independent from the communication model and thus can be used for any mobile WSN. In addition, GRASP automatically copes with dynamic changes of the shape or size of the region of interest, *i.e.* without requiring to be explicitly aware of such changes. Beyond its ability to reach any point with probability one, we prove that our routing protocol provides – for free, with zero knowledge on the ROI and no coordination between nodes – the network with efficient self-deployment properties, fairly comparable to *on purpose* distributed algorithms with respect to the resulting configuration. More specifically, we prove that the routing protocol in one dimension converges to a full coverage of the ROI with the minimum required number of sensors in a finite number of steps. We also show in two dimensions that sensors reach autonomously a stable full coverage following geometrical patterns, requiring only 1.5 the optimal number of sensors to cover the ROI. We also present simulation results matching our theoretical analysis. In addition, we address practical matters and consider *(i)* how to handle concurrent message routing and *(ii)* how to leverage the geometrical structure to achieve sensors sleep-wakeup in order to increase the WSN lifetime.

Roadmap The rest of the paper is organized as follow: Section 2 presents the design rationale behind GRASP along with the detailed algorithm. Section 3 provides a theoretical analysis of GRASP with respect to the self-deployment properties both in the one and two dimensional cases. Section 4 presents experimental results obtained by computer simulations and gives a performance analysis of GRASP with respect to *(i)* its efficiency in terms of routing delays and energy, and *(ii)* its impact on the network topology in terms of self-deployment. Sections 5.1 and 5.2 tackle the practicality of GRASP by proposing an algorithm to handle concurrent message routing and a sleep-wakeup scheme leveraging the network geometry resulting from GRASP to increase the system lifetime. Finally, we provide in Section 6 a list of perspectives and on-going work to increase GRASP performance.

2 GRASP: A Routing Algorithm for Mobile WSNs

Before proceeding with the details of our novel routing protocol, we briefly survey traditional routing techniques in Ad-Hoc networks.

2.1 Routing protocols

Routing protocols for such networks are divided into several classes depending on whether they cache or not routes, use or not geographic coordinates, *etc.* For instance, pro-active routing algorithms – such as Optimized Link State Routing Protocol (OLSR) [3] – store dynamically updated routing tables. Reactive routing algorithms – such as Ad hoc On-Demand Distance Vector (AODV) Routing [15] – discover routes by flooding the network. Geographic routing algorithms – such as Greedy Perimeter Stateless Routing (GPSR) [9] – use geographic coordinates to determine a path from the source to the destination. In the context of our work, the high dynamics of the network and the fact that a message is composed of a single message (as opposed to an information flow), the first two classes of algorithms are not relevant for the applications we are targeting. On the other hand, as the destination of a message is a point of the ROI, GRASP definitely belongs to the latter class. The main difference of GRASP over traditional face transversal methods is the fact that it leverages node mobility to cross routing holes.

Greedy algorithms have received great attention for the design of geographic routing algorithms, consisting in forwarding a message to the neighbor closest to the destination. Due to the geographic sparsity of the network, a node may not be able to forward a message, as illustrated

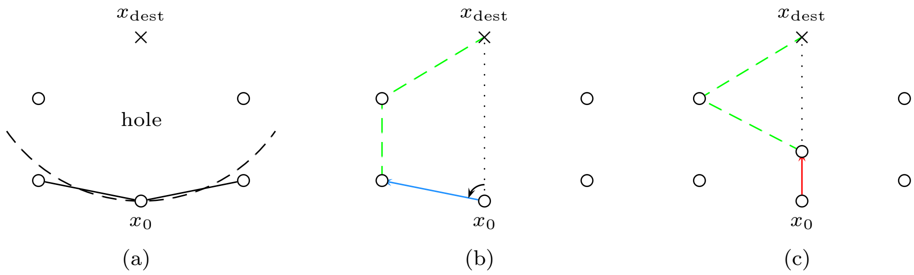


Figure 1: Illustration of a *routing hole* in a WSN. (a) Node x_0 is closer to the destination x_{dest} than its neighbors making greedy routing fail. (b) Face-based routing algorithm – such as GPSR – route along the edges bordering the void while (c) Grasp leverages the node mobility to move toward the destination until greedy forwarding is possible. Green dashed lines denote *greedy forwarding* and blue and red arrows denote respectively hole circumventing and moving.

on Figure 1(a). Most geographic routing algorithms address the problem of holes in the topology by routing around the empty zone. This very popular class of algorithms, referred as face-based routing, rely on preliminary computation on the communication graph to make it planar (using most of the time heuristics such as Relative Neighborhood Graph or the Gabriel Graph) and suffer from numerous pitfalls due to the gap between the fixed radius communication model and the real-life [10].

2.2 System model

Our Mobile Routing Algorithm (Grasp) considers a network of mobile entities with wireless communication capabilities deployed in an obstacle-free region. We assume a disc model for sensing (i.e., a node is able to sample its environment up to a distance r_s from its current position) and symmetric communication links. We further assume that nodes are able to orientate and localize themselves inside the ROI by the use of a compass and a localization system such as GPS or a distributed location algorithm [12]. In addition, we assume that they also know their neighbors coordinates using for instance periodic beacons.

2.3 Design rationale

The design rationale behind Grasp can be explained through the analogy with a soccer game. In a game, a set of mobile intelligent entities, namely the *players*, are deployed on a rectangle area, the *pitch*, and collaborate in order to deliver a packet, the *ball* at a given position, the *goal*. To succeed, the players can either run or pass the ball to a team-mate provided that the distance between them is not too large. Obviously, passing the ball to a team-mate is less tiring than running to put the ball in the goal on his own. In this context, the energy constraint is that the players must keep the ability to move until the end of the game. On one hand, the players must pass as often as possible so as to save their energy but on the other hand, at some point, a player's reachable team-mates may be all in worst position than himself to reach the goal. Two questions come naturally to mind: assuming a limited view of the game and limited passing capabilities, *(i)* "to which of their reachable team-mates should a player pass the ball?", and *(ii)* "when should a player move instead of passing?"[1].

Following this analogy, we propose a simple geographic routing algorithm leveraging nodes mobility to transparently cross holes in the topology. Nodes act greedily for both forwarding and moving: the distance between the current position and the destination should be decreased. If the destination lies in the sensing disc of the node in charge of the request, the node fulfills the request itself. If not, it can either forward the message or move as follow.

[1]Note that we do not claim here that such a greedy strategy provides the best strategy to win a soccer game.

- Forward: using local information on its neighbors position updated by means of periodic beacons, the node in charge of the message picks the closest node, *and closer than itself*, to the destination – if any – and forwards.
- Move: otherwise, the node starts moving in straight line toward the destination until it can either forward the message to a node closer than itself to the destination or sense itself at the destination (meaning that its distance to the destination is lower than r_s). The node stops and performs the adequate action.

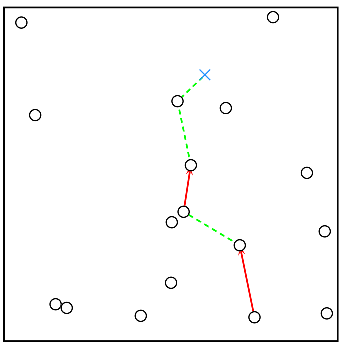
Algorithm 1 gives a detailed pseudo-code version of GRASP. Figure 1(c) illustrates the way GRASP deals with routing voids and Figure 2 shows a sample path – combining both forwarding and moving – used by GRASP to deliver a message in a sparse two-dimensional mobile WSN.

Algorithm 1 GRASP: A routing algorithm for mobile WSN

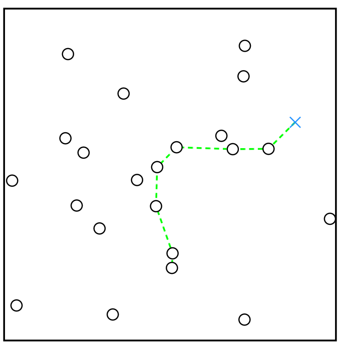
Input: upon reception of a sampling request (for position x_{dest}) at node x_0
  while $\nexists x \in \text{neighborhood}\,(x_0)\ \mid d\,(x, x_{\text{dest}}) < d\,(x, x_0)$ and $d\,(x_0, x_{\text{dest}}) > r_s$
    move toward x_{dest}
  end while
  if $d\,(x_0, x_{\text{dest}}) \leq r_s$ **then**
    sample x_{dest}
  else {exists a node in x_0's neighborhood closer to x_{dest} than x_0}
    forward to $\underset{x \in \text{neighborhood}(x_0)}{\text{argmin}}\ d\,(x, x_{\text{dest}})$
  end if

The strength of the algorithm described above is its simplicity and the resulting characteristics. First, no assumption is made on the the radio communication model. More specifically, the communication range is not explicitly used by the algorithm. Actions are only based on a pure localized and distributed information built from periodic beacons advertising the identifier and the position of a node. The beacon messages are locally broadcast (1 hop message).

The second important characteristic relates to the deployment: GRASP acts with zero knowledge on the ROI. GRASP is purely distributed and decentralized. Its ultimate goal is to allow a sampling of any location of the ROI. The deployment of the nodes is not explicitly controlled by the geographical shape of the ROI but by the application needs. If the application needs to sample specific positions, it sends requests toward these positions. Nodes route requests and thus potentially move in order to satisfy them. The deployment of the nodes is dynamic and self adapts to the shape and to any potential evolution of the region of interest.



(a)



(b)

Figure 2: Sample paths where (a) Grasp leverages the node mobility to cross a routing hole ; (b) Grasp does not require any node to move (i.e., uses only greedy forwarding). Green dashed lines denote greedy forwarding while red arrows denote moving.

Assuming no packet loss, no node failure and unlimited energy, Grasp offers the following properties:

- **Sampling request fullfilment with probability one**, regardless of the number of sensors: at each step the distance between the node in charge of the message and the destination is reduced either by moving toward the destination or by forwarding the message to a closer node.
- **Transparent hole filling**: intuitively, a node is required to move when the area between itself and the destination does not contain any other node. Therefore, moving toward the destination fills the routing hole. More concretely, Grasp offers a spreading property in the sense that, in addition to filling routing holes, when making a node move this last may not get closer than a given distance called *repulsion radius*, close to the communication radius (a closed form expression of the radius is given in Section 3.2), to any other node.

Under such ideal assumptions, Grasp outperforms traditional geographic routing as it takes benefit of the nodes mobility to overcome dead-end routes. Moreover, the deployment of nodes is on demand. Such reactive behavior requires neither specific hole detection nor a pre-deployment computation phase. In addition, all properties of Grasp hold with evolving, in shape and size, regions of interest without explicitely requiring to be aware of such changes.

Note that we do not establish one rigid path from one node to another: routing a packet from the same source node to the same destination may require some nodes to move. This is not a burden as most of the targeted applications require to send an order without expecting an answer straight away. For instance, the ultimate goal of an adaptive sampling application is to have *any* node sample at a given location and not to get this information at a specific node.

3 Theoretical analysis

In this section, we present a theoretical analysis of our routing protocol for mobile WSN with respect to self-deployment properties considering both one and two dimensional ROIs. One dimensional ROIs are relevant to many applications such as barrier coverage [11, 2] for the purpose of intrusion detection. In this context, the nodes are deployed on a one-dimensional space forming the frontier of the area where intrusion should be detected. Considering a two-dimensional ROI is obviously relevant to most environmental monitoring applications.

We denote by an *optimal deployment* a network configuration in which any point of the ROI can be reached without requiring any node to move while using the routing algorithm presented in Section 2. Due to the greedy nature of the forwarding algorithm, such a configuration should provide *full greedy-connectivity.* Our analysis consider the case where the sensing radius r_s is equal to the communication range r_c. We assume the disk model in order to derive analytical results. Note that lifting the assumption $r_s = r_c = R$ would impact on the optimal configuration but the general sketch of the proofs still holds. The purpose of this paper is not to study exhaustively all the possible ratio values between r_s and r_c but to present a formal framework for Grasp. Under such assumptions, we derive the optimal deployment that uses a minimal number of nodes in one and two dimensions. We prove that Grasp converges, in one dimension, to such a full greedy-connectivity with the minimum required number of sensors in a finite number of steps. For two dimensional ROIs, we give an intuition to why Grasp makes the network converge to an optimal configuration and what the final configuration would be when the number of nodes is greater than a given threshold.

3.1 One-dimensional ROI

In this section, we consider the simple one-dimensional case where the nodes are deployed on a segment and we prove that, provided that the number of nodes is sufficient, Grasp impacts on the node positioning in such a way that eventually the network achieves both full coverage and full connectivity.

In one dimension, the optimal deployment achieving full coverage and full greedy connectivity of a segment of size ℓ (which is equivalent to full connectivity in this particular case) with the minimum number of nodes is the regular subdivision of step R (see Figure 3), and the number of nodes used is $N_{\text{opt}} = \lceil \ell/R \rceil - 1$.

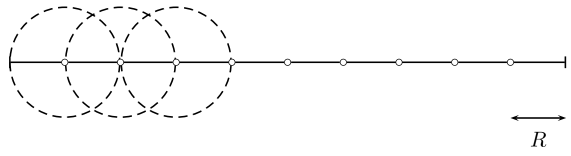


Figure 3: Optimal deployment of a WSN on a one-dimensional region ensuring both full coverage and full connectivity using the minimum number of nodes ($N_{\text{opt}} = \lceil \ell/R \rceil - 1$).

We consider a network of N nodes deployed on a ROI of size $\ell = (N+1) \cdot R$ and we assume that every packet is sent to a randomly chosen point on the segment and originates from a random node in the network drawn from a uniform distribution on the full set of nodes. The random destination is drawn from the uniform distribution on the segment with probability $1 - p_0$ and from a uniform distribution on its ends with probability p_0. We sort the nodes so that their positions $X = \{x_i\}_{i=1}^{N}$ enjoy:

$$0 \leq x_1 \leq \ldots \leq x_N \leq \ell$$

We set $x_0 = 0$ and $x_{N+1} = \ell$, and we denote by S_i^k and s_k respectively the i-th subsegment $[x_i, x_{i+1}]$ and the number of segments of size R after the transmission of k messages.

Lemma 1 *Routing a packet to one end of the ROI does not decrease s_k.*

The topology of the networks changes only when a node moves. This situation occurs in two cases: *(i)* move and forward and *(ii)* move and deliver (see Figure 4). As shown in Figure 4(c), when the node x_i moves and forwards, it *may* destroy a segment of size R (i.e., S_{i-1}) and it creates a new one (i.e., S_i). Since the destination is one of the segment ends, $[x_N, x_{\text{dest}}] = S_N$ as shown in Figure 4(d), when a node moves and delivers, it *may* destroy a segment of size R (i.e., S_{i-1}) and it creates a new one (i.e., S_i). In conclusion, s_k does not decrease. □

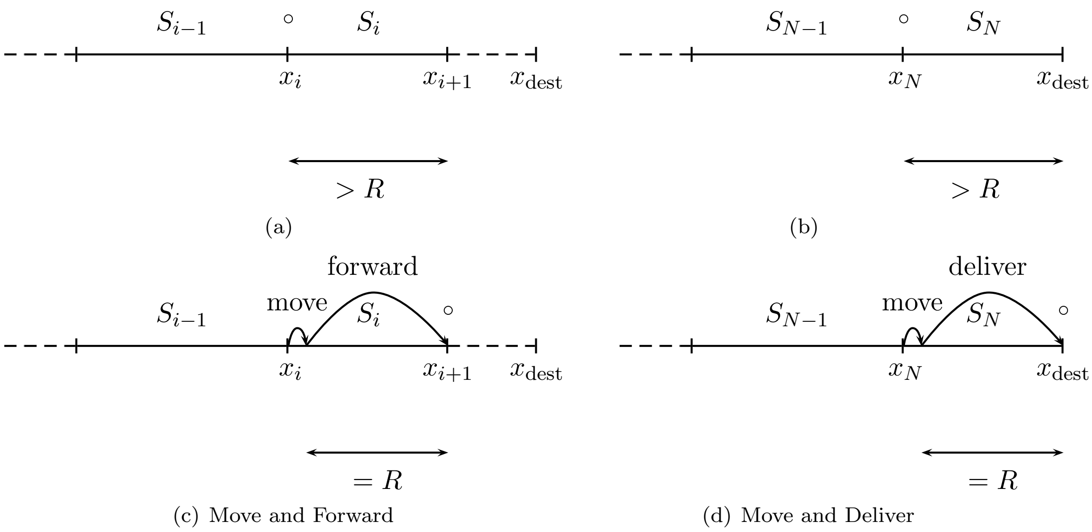


Figure 4: Impact of Grasp (movement phase) on the network topology with respect to the segments size when routing toward one of the segment ends.

Lemma 2 *For any non-optimal configuration the probability of s_k increasing after routing N messages is low-bounded by a strictly positive quantity independent from the nodes positions X.*

Let X be a non-optimal configuration of the network meaning that at least one segment is of size different from R. Since $\sum_{i=0}^{N} |S_i| = (N+1) \cdot R$, there exists two segments of size respectively smaller and greater than R. So, one can find i and j so that $|S_i| > R$, $|S_j| < R$ and $|S_n| = R$ for any n between i and j (one can consider without loss of generality that $i < j$). Figure 5(a) illustrates this configuration of the network.

Figure 5: Impact of routing N packets from node $i+1$ to node $j+1$ on the network topology with respect to the segments size: $s_{k+N} \geq s_k + 1$.

Consider the sequence of N messages to be routed from node $i+1$ to the right end of the segment. Using Lemma 1, routing from node $j+1$ to the right end of the segment does not decrease s_k and routing from node $i+1$ to node $j+1$ *(i)* increases the size of S_i by $(|S_j| - R)$, *(ii)* will not change the size of S_n $(i < n < j)$ and *(iii)* fixes the size of S_j to R (see Figure 5). In conclusion: $s_{k+N} \geq s_k + 1$.

$$\mathbb{P}\left(s_{k+N} > s_k \mid s_k < N\right) \geq \left(\frac{1}{N} \times \frac{p_0}{2}\right)^N > 0$$

□

Theorem 1 *The number of segments of length R tends to N as the number of sent messages tends to infinity with probability one and the expected time to reach the limit $T_k = \min\{k, s_k = N\}$ is finite.*

As we mentioned in the introduction, the optimal deployment is stable in the sense that if the network is optimally deployed then any point of the ROI can be reached without moving and so the topology does not change ; that is $\mathbb{P}\left(s_{k+1} = s_k \mid s_k = N\right) = 1$. Together with Lemma 2, it implies that T_k is "low-bounded" by the geometric law and so we get:

$$s_k \overset{k \to +\infty}{\longrightarrow} N \ \text{ w.p. } 1 \qquad \text{and} \qquad \mathbb{E}\left[T_k\right] \leq N \cdot \left(N \frac{2}{p_0}\right)^N$$

□

Remark Following the same line of reasoning, when considering $r_s \neq r_c$ it can be proved that the optimal deployment that uses a minimum number of nodes is a regular subdivision of step length equals to $R = \min(r_c, 2r_s)$. The node deployment induced by GRASP still converges toward this optimal configuration.

3.2 Two-dimensional ROI

A recent work of Iyengar *et al.* [7] explores the problem of the optimal deployment of a static WSN in a two-dimensional region ensuring a *connected-coverage* (*i.e.*, full coverage with full connectivity) focusing on the case $r_s = r_c = R$. Using geometric considerations, they derive a lower-bound on the optimal node density to cover a zone in a connected way: $d_{\text{opt}} \geq 0.52/R^2$. In addition they propose a strip-based configuration which approaches tightly the bound $d_{\text{strip}} \approx 0.54/R^2$. The strip-based configuration is asymptotically optimal. This configuration is composed of horizontal strips (i.e., optimal configuration in a one-dimensional region) spaced by $\left(1+\sqrt{3}/2\right)R$. This way, any two nodes on the same line can communicate. On the other hand, a vertical strip of nodes connects the horizontal strips together ensuring the full connectivity of the network. Figure 6 depicts the strip-based deployment. In [1], Bai *et al.* extended those results by proving the asymptotic optimality of the strip-based deployment pattern for any $r_c/r_s < \sqrt{3}$ (not only $r_c = r_s$).

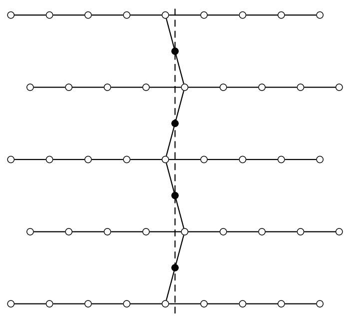


Figure 6: Strip deployment ensuring asymptotically optimal connected-coverage: the network is composed of fully connected horizontal strips connected together through a vertical strip. Still, a greedy geographic algorithm may not be able to reach to reach any point of the ROI.

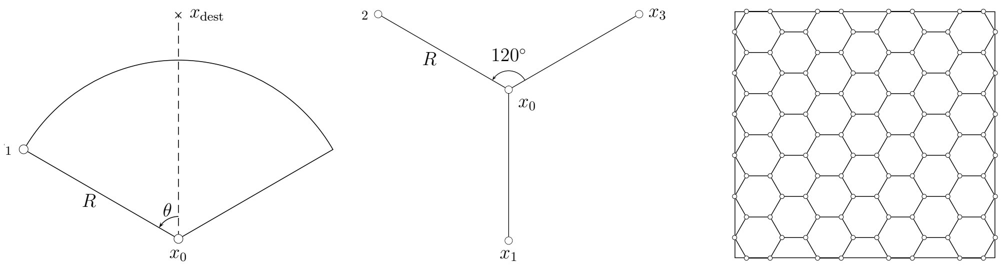
In their work Iyengar *et al.* considered full connectivity which characterizes a configuration where there exists a path between any two nodes while our work focuses on network deployment where a greedy geographic algorithm can find a path between any two nodes. The full greedy connectivity can be formalized as follow: for any destination point of the ROI, any node x_0 in the network can communicate with a node x_1 closer to the destination than x_0. Based on this definition we prove that the hexagonal lattice (see Figure 7(c)) is the optimal deployment ensuring full greedy-connectivity and we compute the required number of sensors to cover the area using this mesh. In addition we prove that this configuration also ensures full coverage of the ROI assuming that the communication radius and the sensing radius are equal.

Let x_0 be the forwarding node, x_1 one of its neighbors and x_{dest} the point of the ROI to be reached. We denote by x the euclidean distance between x_0 and x_{dest}, by r the euclidean distance between x_0 and x_1 and by θ the angle between x_1 and x_{dest} from x_0's point of view (see Figure 7(a)). The direction of x_{dest} is reachable through x_1 if and only if, for any $x > R$, $d\left(x_0, x_{\text{dest}}\right) > \left(x_1, x_{\text{dest}}\right)$. That is

$$\begin{aligned} \forall x > R, \qquad x^2 &> (r\sin\theta)^2 + (x - r\cos\theta)^2 \\ x^2 &> r^2 + x^2 - 2\cdot r\cdot x\cos\theta \\ \cos\theta &> \frac{1}{2}\cdot\frac{r}{x} \end{aligned}$$

This property must hold for any x greater than R and the node density is minimized for $r = R$, so the maximum angle $\theta_{\max}$ enjoys $\cos\theta_{\max} > 1/2$. Thus, the maximal angle offering greedy forwarding with probability one is 30°. That is a maximum angle of 60°between two consecutive neighbors. The base pattern of the optimal configuration is an equilateral triangle of radius R (see Figure 7(b)) and the optimal deployment is a regular hexagonal mesh of radius R (see Figure 7(c)).

Since each hexagon of the mesh is fully covered by the sensing disc of its vertices, the optimal deployment with respect to full greedy-connectivity ensures the full coverage of the ROI. In that configuration, a closed form expression for the number of nodes required to cover a zone of area A can be derived: each node participates to exactly three hexagons which contributes for $3/2\cdot\sqrt{3}R^2$ in the coverage of the zone. Each hexagons being composed of six vertices, the minimum number



(a) (b) (c)

Figure 7: (a) Maximum angle between two neighbors ensuring that greedy forwarding is possible for any destination. (b) Optimal local configuration and (c) optimal deployment of a WSN on a two-dimensional region ensuring both full coverage and full greedy-connectivity using the minimum number of nodes

of nodes is given by:

$$N_{\text{opt}} = 6 \cdot \frac{1}{3} \cdot \frac{A}{6 \cdot \frac{1}{2} \cdot R \cdot \left(\frac{\sqrt{3}}{2} R\right)} = \sqrt{3} \frac{A}{\left(\frac{3}{2} R\right)^2}$$

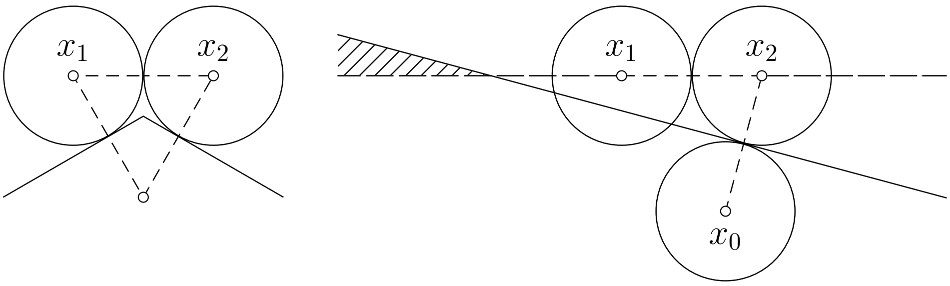
The necessary and sufficient condition on the angle between two consecutive neighbors implies that meshes following regular n-polygonal patterns ensure both full coverage and full greedy-connectivity for $3 \leq n \leq 6$. We evaluate these deployments with respect to density (i.e., the number of nodes N needed to cover a zone of area A) and the degree of greedy-connectivity k (i.e., the minimum number of neighbors closer to the destination than the forwarding node itself) which reflect respectively the cost of the system and its resilience to failures. These two criteria are dual as increasing the system fault tolerance requires a larger number of nodes. Table 1 summarizes theoretical comparison results of the hexagon, square and triangular latices.

pattern	hexagon	square	triangular
k	1	1	2
N/A, $(R = 1)$	$\frac{4}{9}\sqrt{3} \approx 0.77$	1	$\frac{2}{3}\sqrt{3} \approx 1.15$

Table 1: Comparison of n-polygonal patterns: k-greedy connectivity *versus* node density.

Convergence of the deployment Here, we give a brief outline of the configuration convergence of a network running Grasp in a two dimensional region of interest. A first evidence is that nodes can not get closer from each other than $\frac{\sqrt{3}}{2}R$. Note that the minimum distance between two nodes is lower than R. Effectively, we show in Appendix A that a node x_0 is able to cross the R-disc centered on a node x_1 when traveling toward its destination x_{dest} keeping the distance between x_0 and x_{dest} always smaller than the distance between x_1 and x_{dest}. We denote by *repulsion radius* this minimum distance. One may model all nodes by physical balls of radius ranging from $\frac{\sqrt{3}}{2}\frac{R}{2}$ to $\frac{R}{2}$.

Based on the physical model analogy, running Grasp on a mobile WSN can be thought of as packing a set of balls inside a given frame. As one may recall, the intrinsic action of Grasp on mobile nodes is rather to push them than to pack them. Each couple (x_1, x_2) of two adjacent balls $(d(x_1, x_2) \leq R)$ presents two attraction sites (see Figure 8(a)). An attraction site is a place where the probability to move, in order to sample a location behind the line x_1x_2 is null. Note that such a position also minimizes the size of the associated Voronoï region (the set of destinations making a node located at this position move to fulfill a sampling request). If a ball x_0 is not located in a attraction site, that is, if it is not adjacent to both x_1 and x_2 but only to one of them (x_2 without loss of generality) then for each sampling location behind the line x_1x_2, it has a strictly positive probability to move. More precisely, if the sampling location is located on the left of the median of $[x_0x_2]$ and behind the line x_1x_2, the probability to move is non zero (related to the hatched area on Figure 8(b)). Moreover, moving toward a sampling location inside this area pushes the node closer to the attraction site and the ball x_0 then touches the x_1. During the process additional nodes may also arrive and stick one to another. Eventually, all nodes will converge to a attraction site with no more possibility to escape. Such final configuration is the well-known stable triangular lattice. Note that we do not claim that only triangular tiles can be formed but rather that they are the most likely to be formed. More specifically, hexagonal tiles allow greedy routing (as shown in the previous paragraph) and require less nodes but they are not stable. Effectively, small deviations on the nodes positions in a hexagonal tile may lead to a reconfiguration of the tile into a triangular-like one: a node routing toward the opposite node in the hexagon would move and stop near the center of the hexagon forming equilateral triangles.



(a) (b)

Figure 8: Attraction sites of a couple of nodes: (a) the attraction sites are the two points which forms, together with the two existing nodes, an equilateral triangle. Plain lines represent the border of the Voronoï region associated to the attraction site. (b) When x_0 can communicate with only one of the two other nodes, the probability that x_0 must move to route a message behind the line x_1x_2 is non null (related to the hatched area)

4 Experimental results

In this section we give an experimental evaluation of GRASP based on computer simulations. First, we present the experimental setup, and motivate the metrics used for the evaluation. Then, we give the experimental results in one and two dimensions together with their analysis backed up by the theoretic arguments developed in Section 3.

4.1 Experimental setup

In order to evaluate the self-deployment capabilities of GRASP, sensor nodes are deployed uniformly at random in a restricted area of the ROI. If not specified otherwise, in our simulations, nodes are initially deployed on 1/10th of the ROI.

The considered regions require 500 nodes to be connected-covered, with respect to the expressions of optimal density derived in Section 3. We evaluate GRASP's performance for several values of $\rho = N/N_{\text{opt}}$, close to 1. This allows us to evaluate the behavior of GRASP under slight under or over-dimensioning in terms of number of nodes. The results can be used to evaluate the impact of failures on the deployment.

The system dynamic is controlled, *via* GRASP, by message emission. We assume a constant message emission rate λ, uniform amongst the nodes. Thus, during one time unit $\lambda \cdot N$ messages are routed from a randomly chosen node drawn from a uniform distribution on the full set of nodes (the selection method of the destination depends on the dimension of the ROI and is further detailed in Sections 4.2 and 4.3). For the sake of simplicity, we consider the routing of *one message at a time*, the case of concurrent routing being tackled in Section 5.1.

We evaluate GRASP along the following metrics:

- **Average distance covered per node to deliver a message** $\tilde{d}$: assuming that a move is much more time and power consuming than a wireless transmission, this metric reflects *(i)* the global energy consumption of the system and *(ii)* the average delivery delay. An optimal deployment being a network configuration where any point of the ROI can be reached without moving, $\tilde{d}$ should decrease to zero as time tends to infinity (assuming a constant emission rate λ) provided that the number of nodes is sufficient.

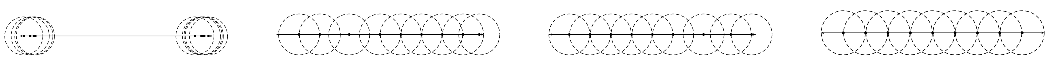
- **Distribution of the average distance covered by a node to deliver a message** p_D: this metric brings additional information on the network load. More specifically, it reflects how the energy consumption (i.e., the movement) is distributed amongst the nodes. Typically, an ideal pdf for d is a Dirac, located at zero provided that the number of node is sufficient, or at a strictly positive value otherwise. Assuming a homogeneous initial distribution of energy resources, p_D evaluates the system lifetime.

- **Probability that at least one node moves to deliver a message** p_m: the order of magnitude of the delivery delay is dependent on whether the messages are delivered using only wireless communication or by moving to cross routing holes, regardless of distance covered. Thus, p_m is a good indicator of the system QoS dual to $\tilde{d}$.

4.2 One-dimensional ROI

In this set of experiments, the ROI is a segment. To match the analysis, there must be a non zero probability that the destination is one extremity. We consider here that the destination is picked with probability p_0 at one extremity of the segment, the specific extremity being chosen with probability 0.5. Otherwise, the destination is picked uniformly at random on the segment.

The results presented below are averaged over 25 independent runs of *Monte-Carlo* simulations. The metrics are computed numerically from the corresponding closed form formula.

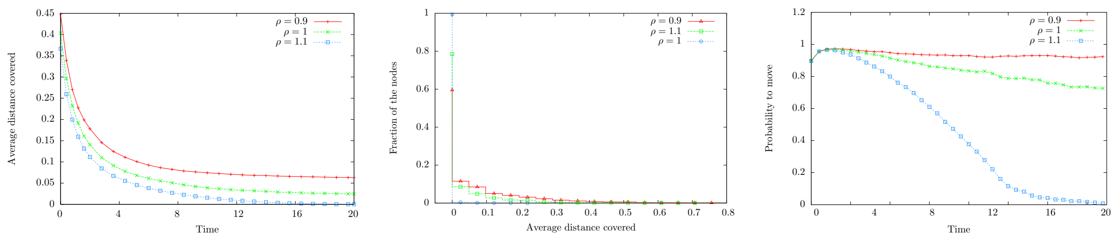


(a) $t = 0$ (b) $t = 5$ (c) $t = 10$ (d) $t = 15$

Figure 9: Evolution of the network topology over time. Ten nodes are initially uniformly deployed on two small area near the ends of a one-dimensional ROI. The message emission rate is set to $\lambda = 1$ and the probability to pick the ends of the region as destination is set to $p_0 = 0.1$.

Evolution of the network topology Figure 9 depicts the evolution over time of the network topology in one dimension. We consider in this experiment the number of nodes required for the optimal deployment. We observe that starting from a skewed distribution of nodes positions over the segment (one half of the nodes is deployed in the left 1/10th of the segment and the other half in the right 1/10th), GRASP achieves an optimal deployment (connected-coverage) in no more than 15 iterations, corresponding to 150 packets. We observe on Figure 9(b) and Figure 9(c) that the intermediary configurations are close to the uniform distribution of nodes over the segment. This shows that the foreseen average distance covered per node to deliver a message decreases quickly.

Average distance covered Figure 10(a) depicts the average distance covered by a node (relatively to the communication radius R) to deliver a message over time. A point (x,y) on the curve shows that after x time units, on average each node is expected to move of a distance y to deliver



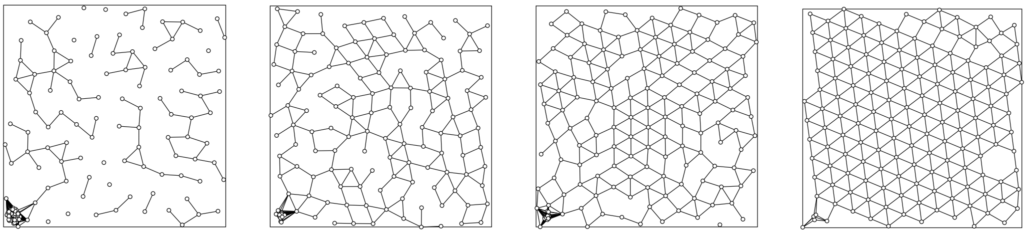
(a) (b) (c)

Figure 10: Experimental evaluation of Grasp in a one-dimensional ROI: (a) average distance covered per node (relatively to the communication radius R) as a function of time (assuming a fixed message emission rate $\lambda = 1$) ; (b) load distribution in the systems with respect to the average distance to be covered by a node to deliver a message (relatively to the communication radius R); and (c) the probability that Grasp uses mobility to deliver a message (as a function of time, $\lambda = 1$)

a packet. We conducted the simulations for three configuration of ρ=0.9, 1, 1.1, considering respectively 90%, 100% and 110% of the optimal number of nodes. The plot shows that the average distance decreases quickly in the three configurations. The ρ=0.9 configuration converges toward a non zero, yet low (approximatively 0.05, meaning that for a radius of 100m, each node would be expected to move on average of 5m.). We proved that the ρ=1 configuration converges toward zero. We observe on the experiments that after 20 time units the system has not converged yet. However, we observe that with only 10% of additional nodes (as compared to the optimal number of nodes), the average distance converges to zero in about 15 cycles.

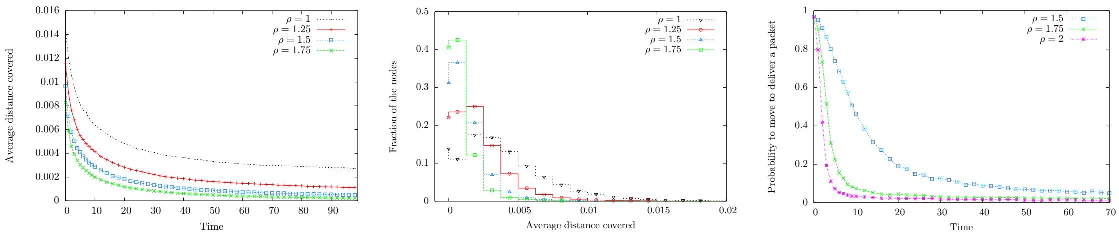
Network load Figure 10(b) depicts the distribution of distance that nodes need to cover to deliver a packet after 20 time units. The step x=0.1, y=0.05-0.1 shows that 10% of the nodes cover a distance ranging from 5% to 10% of the communication radius. The impulses on the left of the figure (i.e., at $d = 0$) refer to the fraction of nodes that will no longer move. For ρ=1.1, after 20 time units, no node will move as the configuration has already converged. In the ρ=1 configuration, we observe that the system is in the process of converging. In the ρ=0.9 configuration, we observe a heavy tail, corresponding to the nodes at the extremes of the connected components that will keep moving, as the number of nodes is not sufficient.

Probability to move Figure 10(c) presents the probability that at least one node moves to deliver a message (p_m) for the same three values of ρ. We observe that when ρ=0.9, the probability that at least one node moves stays very close to one. Counter-intuitively, this is a good news as far as the system lifetime is concerned. This reflects the fact that there is approximatively a uniform spread of the nodes over the segments, as opposed to a resulting configuration of two connected components separated by a hole. Such a configuration would force the nodes on the frontier of the hole to handle the moves required to route packets. As already observed in Figure 10(a), we observe a slow convergence for the optimal number of nodes, while only a 10% additional node, the convergence is much faster and significantly impacts the shape of the curve. Note that although we conducted experiments with 10% additional nodes, we know that only one additional node is sufficient to avoid the slow convergence (due to the oscillations around the optimal), observed in the ρ=1 configuration. Note that p_m increases slightly during the first steps of the simulation. Initially the nodes are deployed in 1/10th of the ROI, thus the $p_m \approx 0.9$: regardless of the source of the packet, a node is required to move if the destination is in the uncovered zone of the ROI (the communication radius being negligible compared to the size of the ROI), that is 9/10th of its area. When a node moves to deliver a message, it becomes isolated from the main connected component. Thus a message originating from that node requires a movement with a probability close to one.



(a) $t = 1$ (b) $t = 5$ (c) $t = 50$ (d) $t = 1000$

Figure 11: Evolution of the network topology over time. A hundred nodes are initially uniformly deployed on a small area (1/100th) of a two-dimensional toric ROI. The message emission rate is set to $\lambda = 10$. The plain lines represent the communication graph. For the sake of simplicity, edges between border vertices are not represented. Note the existence of an hexagonal tile and a few square tiles in the communication graph.



(a) (b) (c)

Figure 12: Experimental evaluation of Grasp in a *toric* two-dimensional ROI: (a) average distance covered per node (relatively to the communication radius R) as a function of time (assuming a fixed message emission rate $\lambda = 1$) ; (b) load distribution in the systems with respect to the average distance to be covered by a node to deliver a message (relatively to the communication radius R) ; and (c) the probability that Grasp uses mobility to deliver a message (as a function of time, $\lambda = 10$)

4.3 Two-dimensional case

In this set of experiments, the ROI is a square folded into a torus shape. The destinations of the messages are drawn from a uniform distribution on the entire ROI. The results presented below are averaged over 25 independent runs of *Monte-Carlo* simulations. The metrics are computed using *Monte-Carlo* simulations ($20 \times N$ independent simulations): for instance, to evaluate $\tilde{d}$, we measure and average the distance covered by nodes to deliver $20 \times N$ messages. The sources of these messages are picked uniformly at random amongst the full set of nodes and their destinations are picked uniformly at random in the entire ROI. As we stated in Section 3.2, a mobile WSN running Grasp is expected to converge toward a regular triangular lattice. As a matter of fact, the expected minimum number of nodes to ensure greedy connected-coverage is 1.5 the optimal: we evaluate Grasp for a number of nodes around 1.5 the optimal ($\rho = 1, 1.25, 1.5, 1.75$).

Evolution of the network topology Figure 11 depicts the evolution over time of the network topology in two dimensions. To match our theoretical analysis, we consider in this experiment 1.5 the number of nodes required for an optimal deployment with the minimum number of nodes. We observe that starting from a localized initial distribution of nodes positions over the ROI (the nodes are deployed in the 1/100th of the square), Grasp *(i)* converges to an optimal deployment (connected-coverage). We observe on Figure 11(a) that the first stage of the deployment spreads the nodes inside the entire ROI. Figures 11(b)-11(d) illustrate the aggregation process and the final configuration described in Section 3.2. One may notice that throughout the process, the communication graph of the WSN is planar (except from in the deployment area). This property of Grasp is detailed in the next sections and its formal proof is provided in Appendix B.

Average distance covered Figure 12(a) depicts the evolution of the average distance covered by node to deliver a message $\tilde{d}$, in a two-dimensional region of interest. We consider a network with a number of nodes equal to $\rho = 1$, 1.25, 1.5 and 1.75 the optimal. As expected from Section 3.2, $\tilde{d}$ tends to a non-null limit for $\rho < 1.5$ and to zero otherwise. This matches the theoretical results, since the triangular lattice formed by Grasp requires 1.5 the minimum number of nodes to ensure greedy connected-coverage. The convergence is slower in two than in one dimension. Yet after 20 iterations, $\tilde{d} \approx 4.10^{-4}R$ (when $\rho = 1.5$ and $2.10^{-4}R$ when $\rho = 1.75$) which is negligible: for a radius of 100m, each node would be expected to move on average of 4cm. Note that, using only 15% more nodes than needed, reduces the average distance covered by a node of 50%. The main difference between one and two-dimension lies in the fact that while the base patterns of the optimal deployment (i.e., segments of size R) are built in one shot in one dimension, it takes several iterations in two dimensions (equilateral triangles of size R, see Figure 8).

Network load Figure 12(b) depicts the distribution of distance that nodes need to cover to deliver a packet after 20 time units. When the number of nodes is sufficient for Grasp to converge (i.e., $\rho \geq 1.5$), the plots are decreasing and show that convergence is in progress. The high fraction of nodes which never move (up to 30% for $\rho = 1.5$) together with the shape of the histogram reflects the existence of large greedy-connected components with low redundancy inside each of them: the nodes inside the greedy connected components never move and the ones on their border moves to ensure connectivity between the components. The fact that the distance covered by this nodes is very low (at max 1% of the communication radius for $\rho = 1.5$) implies that the components are spatially extended reflecting a low redundancy (with respect to the initial deployment where most of the nodes never move but the ones which move cover on average 1/4th of the ROI). When the number of nodes is not sufficient for Grasp to converge (i.e., $\rho < 1.5$), the network load has a bell-shape with a maximum at $d = 2.5.10^{-4}R$ (for $\rho = 1$), a heavy tail for the large values and a non-negligible fraction of *still* nodes (15% for $\rho = 1$). Therefore the network load in terms of distance covered is well balanced between the the nodes. Thus the energy needed for routing is evenly shared between a large fraction of the nodes resulting in an extended lifetime of the system.

Probability to move Figure 12(b) presents the probability that at least one node moves to deliver a message for a number of nodes sufficient for Grasp to converge. With the minimum number of nodes ($\rho = 1.5$), p_m decreases quickly to 20% and then converge slowly to zero. For slightly higher values of ρ, the first decreasing stage is drastically speed-up: $p_m \approx 2\%$ after less than 20 iterations. Note that the emitting rate λ is ten times larger than for the other experiments.

Summary

Experiments confirmed the theoretical results presented in Section 3. In one dimension, we observe that *(i)* considering the minimum requested number of nodes, Grasp converges toward an optimal deployment; *(ii)* by increasing slightly the number of nodes (with respect to the minimum), the convergence speed is drastically improved; *(ii)* when considering less nodes than the minimum, Grasp achieves a good quality deployment over the ROI, achieving a uniform spread over the zone. This results in spreading the load of moving among nodes rather than imposing this load on only a few nodes.

In two dimensions, the results confirm the intuition of convergence. Grasp requires only 1.5 the minimum requested number of nodes to converge resulting in a triangular lattice deployment. Not only we believe this is a reasonable bound but this increased number of nodes is leveraged in several ways. First, the triangular lattice provides, over the other regular lattices (including the hexagonal optimal deployment), an increased resilience to failure as each node is provided with two potential neighbors in any direction. Second, as we will explain in Section 5.2, the fact that the triangular lattice is a subgraph of the hexagonal one can be exploited by a clever sleep-wakeup scheme to increase the lifetime of the system of 50%, fully justifying the 1.5 number of nodes over the minimum.

5 Practical matters

Most of the results presented above are of theoretical nature, yet we believe that GRASP can be efficient in practice. So far, we have assumed that one routing operation was processed at a time, we provide in Section 5.1, an algorithm to handle concurrent routing operations as this will happen in practice. Second, we provide a sleep-wakeup algorithm in Section 5.2, enabling to improve upon the lifetime of the system.

5.1 Handling concurrent routing

So far, only one sensing request at a time was considered. This simplifying assumption allows us to derive formal proof of convergence of the network topology but might be restrictive. In this section, we consider concurrent routing operations. GRASP should keep ensuring that *(i)* routing operations eventually succeed and *(ii)* two nodes should not get closer to each other than a fixed repulsion radius. We propose a set of modifications to GRASP so that such properties are ensured and discuss their efficiency with respect to delays and distance covered.

Priority queue We assume that each node maintains a queue of messages ordered by priority. To each message is associated a *Time from Emission* (TFE), updated every time a message is inserted of extracted from the priority queue. The message with the highest TFE being the head. The node movement is driven by the message being processed, namely the queue's head. Note that the number of older messages than a given message is finite and decreasing. This ensures that the delivery of messages is eventually guaranteed regardless of the heuristic chosen to forward the message.

Opportunistic forwarding The simplest forwarding heuristic is to choose a static node to forward to. A less conservative heuristic is to forward a message to a node which is closer to the destination and, if moving, gets closer to the destination. To this end, nodes exchange their speed vector, piggybacked in beacons, and fully characterized by the node's position and the destination of the current message being processed. A third heuristic, ensuring the repulsion radius between any pair of nodes, consists in considering the case of two moving nodes running into each other. Under this heuristic, two nodes getting in contact, should be repulsed from one another, in analogy to the billiard model [13]. To this end, the two nodes exchange their positions and their current head (destination and age). The node the closest to the destination of the oldest head takes over all the messages and start processing them The second node merely stops. This ensures that the two nodes are moving away from one another.

5.2 Sleep-wakeup

As foreseen in Section 3.2 and proved by simulations in Section 4, a network of 1.5 the optimal number of nodes running GRASP converges to a triangular like lattice. Interestingly enough, the triangular lattice is a subgraph of the optimal hexagonal one. Based on this remark, we propose a simple yet powerful sleep-wakeup scheme leveraging the network topology to increase the system lifetime. The triangular lattice (Figure 13(a)) is the union of three hexagonal lattices (Figures 13(b)-(d)), each node of the triangular lattice belonging to exactly two of them. Our sleep-wakeup scheme can be described as follow:

- **Clustering:** the first step is to detect the triangular lattice-based components, the sleep-wake up algorithm being executed independently in each of them. To this end, we assume that each node maintains the identifier of the component it belongs to. A node is able to determine if it is located at the center of an hexagonal tile using its neighbor's coordinates. A node at the center of an hexagonal tile sets its component identifier to its own identifier and forwards it to its neighbors forming the hexagonal tile. Otherwise, the component identifier remains unset. Upon reception of such a message, a node updates its component identifier if it is still unset or lower than the one received. Only nodes located at the center of an hexagonal tile forward it to hexogonal tile vertices. Otherwise, the message is ignored. Eventually, the nodes inside a triangular-lattice based component share the same component identifier. The node whose identifier is the one of the component is the natural leader of the component. This clustering task is periodically executed to cope with network dynamics.

- **Sleep-wake up periods:** each triangular component of the network alternates between three hexagonal configurations in a round-robin manner. The leader (node L in Figure 13) is in charge of spreading the sleep messages to the centers of the hexagonal tiles of the current configuration. The effect of a sleep message is to put in sleep mode the node receiving it for a given period of time. Leveraging the geometry of the network, the spreading of sleep messages can be done optimally with respect to the number of packets sent.

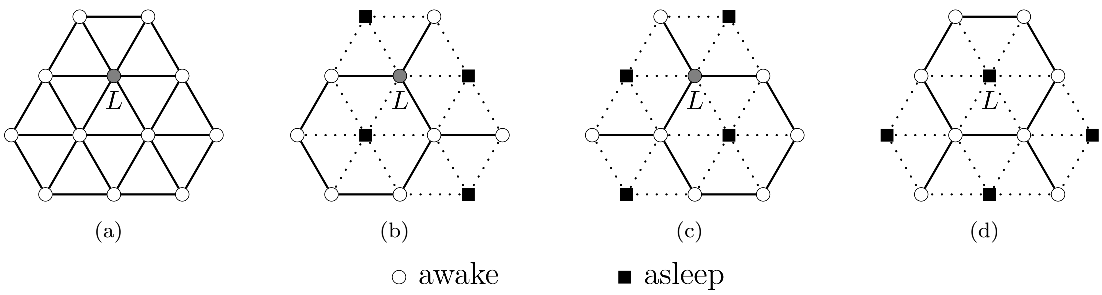


Figure 13: Sleep-Wakeup scheme: (a) nodes inside a triangular lattice adopt in a cyclic manner three sleep-wakeup policies (b),(c) and (d). For the sake of readibility, the leader L is reported in the three figures as a reference point. In each configuration, one third of the nodes are asleep and the communication graph is an hexagonal lattice. Thus, greedy connected coverage is always ensured. Each node is awake in exactly two configurations out of three so the system lifetime is increased of 50%.

Consider a system with the minimum required number of nodes deployed on an hexagonal lattice and assume that each node has a lifetime of one time unit. Obviously, the system lifetime is one time unit. Now, consider a system of $1.5N_{\mathrm{opt}}$ nodes using the sleep-wakeup scheme presented above. Setting its working period (the time length during which the system stays in each of the three configurations) to 0.25 times units, the global system lifetime is extended to 1.5 times units (*i.e.,* six periods). Effectively, each node is asleep two periods out of six yielding a total number of awake periods of four, that is one time unit. In other words, the additional number of nodes over the optimal, namely 1.5, is fully leveraged by increasing the lifetime of the system up to 1.5.

5.3 Energy constraints

As we stated before, moving is much more power consuming than message emission. For this reason or due to mechanical issues, a node may be able to only forward messages but not to move. In that case, a sensing request should be forwarded anyway. At the price of a slight modification, Grasp can overcome this drawback by switching to a more sophisticated routing techniques, while remaining stateless. The repulsion radius (i.e., minimal distance) between two nodes in a mobile network running Grasp ensures that the resulting communication graph is planar (See Appendix B for detailed proof). This property can be leveraged to switch to a face-based routing such as GPSR without any pre-planarization of the communication graph. This requires only a small overhead in the sensing requests packets. Thus, GRASP is able to ensure sensing request fullfilment with high probability even when some nodes run out of energy to move.

6 Conclusions & future work

In this paper, we considered a network of mobile wireless sensors. Their mobility being controlled, we proposed GRASP, a novel and simple stateless algorithm which leverages nodes mobility to route sensing requests. GRASP transparently adapts to evolving region of interest, with respect to size or shape, without requiring to be explicitly aware of such changes. Our algorithm is independent from the communication medium and uses very simple forwarding and motion planning techniques. Thus it is directly applicable to any low capabilities wireless network. Assuming a disc model for communications and a random choice of the sensing locations inside the region of interest, we proved that a network running GRASP converges to a configuration ensuring greedy-connected coverage of the region. In that sense, the simplest routing algorithm leveraging nodes mobility provides the network with self-deployment properties for free. The number of nodes required to ensure convergence is optimal in one dimension and 1.5 the optimal in two dimensions. Our experimental analysis of GRASP matched our theoretical analysis. Finally, we provided GRASP with an Ad-Hoc sleep-wakeup scheme for two-dimensional networks extending the system lifetime up to 50% without jeopardizing the greedy connected-coverage. This fully justifies the overhead factor to the optimal number of nodes. We also tackled briefly two practical matters: efficient concurrent requests routing and dealing with immobilized nodes. We plan to investigate these two tracks and evaluate GRASP behavior using a more realistic MAC layer for wireless communications.

References

[1] X. Bai, S. Kumar, D. Xuan, Z. Yun, and T.-H. Lai. Deploying wireless sensors to achieve both coverage and connectivity. In *Mobihoc'06*, pages 131–142, 2006.

[2] A. Chen, S. Kumar, and T. H. Lai. Designing localized algorithms for barrier coverage. In *MobiCom'07*, pages 63–74, 2007.

[3] T. Clausen and P. Jacquet. Optimized Link State Routing Protocol (OLSR). Rfc experimental, Internet Engineering Task Force, 2003.

[4] Q. Fang, J. Gao, and L. Guibas. Locating and bypassing routing holes in sensor networks. In *INFOCOM'04*, volume 4, pages 2458–2468, 2004.

[5] A. Howard, M. J. Matarić, and G. S. Sukhatme. An incremental self-deployment algorithm for mobile sensor networks. *Autonomous Robots, Special Issue on Intelligent Embedded Systems*, 13(2):113–126, 2002.

[6] K. Huguenin. A Distributed Algorithm for Collaborative Adaptive Sensing. In *UUST'07*, pages 1–9, 2007.

[7] R. Iyengar, K. Kar, and S. Banerjee. Low-coordination topologies for redundancy in sensor networks. In *Mobihoc'05*, pages 332–342, 2005.

[8] S. Jain, K. Fall, and R. Patra. Routing in a delay tolerant network. In *SIGCOMM'04*, pages 145–158, 2004.

[9] B. Karp and H. T. Kung. GPSR: greedy perimeter stateless routing for wireless networks. In *MobiCom'00*, pages 243–254, 2000.

[10] Y.-J. Kim, R. Govindan, B. Karp, and S. Shenker. On the pitfalls of geographic face routing. In *DIALM-POMC'05*, pages 34–43, 2005.

[11] S. Kloder and S. Hutchinson. Barrier coverage for variable bounded-range line-of-sight guards. In *Int. Conf. on Robotics and Automation*, pages 391–396, 2007.

[12] Y. Kwon, K. Mechitov, S. Sundresh, W. Kim, and G. Agha. Resilient localization for sensor networks in outdoor environments. In *ICDCS'05*, pages 643–652, 2005.

[13] B. D. Lubachevsky, R. L. Graham, and F. H. Stillinger. Spontaneous patterns in disk packings. In *Bridges'98: Conf. on Mathematical Connections in Art, Music, and Science*, 1998.

[14] D. Peleg and E. Upfal. A tradeoff between space and efficiency for routing tables. In *STOC'88*, pages 43–52, 1988.

[15] C. E. Perkins, E. M. Belding-Royer, and S. R. Das. Ad hoc On-Demand Distance Vector (AODV) Routing. RFC Experimental 3561, Internet Engineering Task Force, 2003.

[16] A. Rezgui and M. Eltoweissy. Service-oriented sensor-actuator networks: Promises, challenges, and the road ahead. *Computer Communications*, 30(13):2627–2648, 2007.

[17] G. Tan, S. A. Jarvis, and A.-M. Kermarrec. Connectivity-guaranteed and obstacle-adaptive deployment schemes for mobile sensor networks. In *ICDCS'08*, 2008.

[18] J. Wu and S. Yang. Optimal movement-assisted sensor deployment and its extensions in wireless sensor networks. In *ICPADS'06*, volume 1, 2006.

[19] M. Zhang, X. Du, and K. Nygard. Improving coverage performance in sensor networks by using mobile sensors. In *MILCOM'05*, volume 5, pages 3335–3341, 2005.

[20] W. Zhao, M. Ammar, and E. Zegura. Controlling the mobility of multiple data transport ferries in a delay-tolerant network. In *INFOCOM'05*, volume 2, pages 1407–1418, 2005.

[21] W. Zhao, M. H. Ammar, and E. W. Zegura. A message ferrying approach for data delivery in sparse mobile ad hoc networks. In *Mobihoc'04*, pages 187–198, 2004.

[22] Y. Zou and K. Chakrabarty. Sensor deployment and target localization based on virtual forces. In *INFOCOM'03*, 2003.

A Repulsion distance

A node x_0 moving toward a point of the ROI x_{dest} crosses the communication disc of a node x_1 if its distance to the destination when it enters in x_1's R-disc is smaller than the distance between x_1 and x_{dest}. That is, using the notations in Figure 14:

$$y \geq \sqrt{R^2 - \delta^2} + \sqrt{y^2 - \delta^2}$$

This inequality holds for any $y \leq R \;/\; \left(2\sqrt{1 - \left(\frac{\delta}{R}\right)^2}\right)$. The condition $y > R$ implies:

$$\delta > \frac{\sqrt{3}}{2} \cdot R$$

□

B Planarity

Consider a pair of crossing edges x_1x_2 and x_3x_4 (see Figure 15(a)). Let Q be the quadrilateral whose vertices are x_1, x_2, x_3 and x_4 and l_m the length of its shortest edge. Using a convexity argument, it can be proved that l_m is maximized when Q is a square with diagonals of size R (see Figure 15(b)). In that case, $l_m = \sqrt{2} \cdot R/2$. So, if two edges cross there exist at least two nodes x_i and x_j so that $d\left(x_i, x_j\right) \leq \sqrt{2} \cdot R/2 < \sqrt{3} \cdot R/2$ which refutes the result of Appendix A. In conclusion, the communication graph of a WSN running Grasp is planar. □

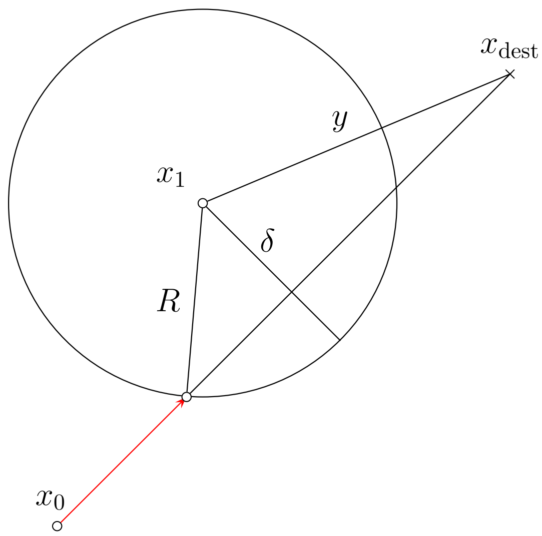


Figure 14: Repulsion radius: two nodes running Grasp cannot get closer than $\frac{\sqrt{3}}{2} \cdot R$.

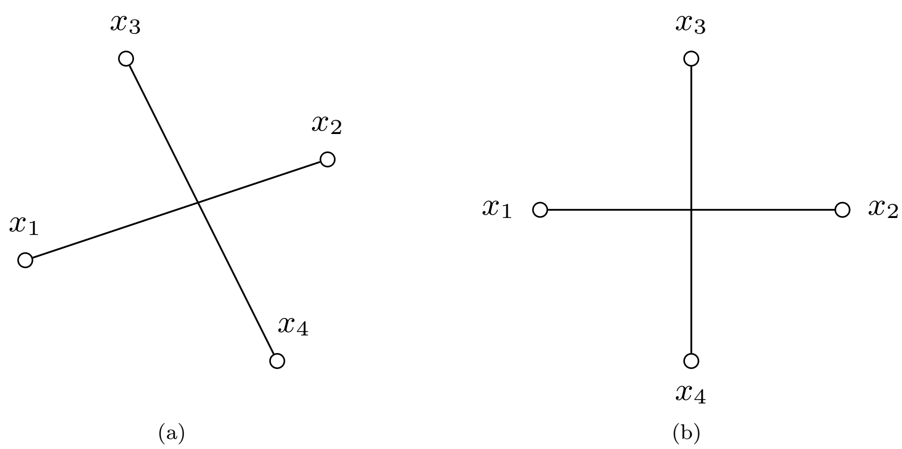


Figure 15: A pair of crossing edges in a WSN: at least two nodes are closer from each others than $\frac{\sqrt{2}}{2}R$

Centre de recherche INRIA Rennes – Bretagne Atlantique
IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)

Centre de recherche INRIA Bordeaux – Sud Ouest : Domaine Universitaire - 351, cours de la Libération - 33405 Talence Cedex
Centre de recherche INRIA Grenoble – Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier
Centre de recherche INRIA Lille – Nord Europe : Parc Scientifique de la Haute Borne - 40, avenue Halley - 59650 Villeneuve d'Ascq
Centre de recherche INRIA Nancy – Grand Est : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex
Centre de recherche INRIA Paris – Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex
Centre de recherche INRIA Saclay – Île-de-France : Parc Orsay Université - ZAC des Vignes : 4, rue Jacques Monod - 91893 Orsay Cedex
Centre de recherche INRIA Sophia Antipolis – Méditerranée : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399