# Partially Controlled Deployment Strategies for Wireless Sensors[*]

Mauro Leoncini[†] Giovanni Resta[‡] Paolo Santi[‡]

September 25, 2007

**Abstract**

In this paper we study the following problem: we are given a certain one- or two-dimensional region $R$ to monitor and a requirement on the degree of coverage (DoC) of $R$ to meet by a network of deployed sensors. The latter will be dropped by a moving vehicle, which can release sensors at arbitrary points within $R$. The node spatial distribution when sensors are dropped at a certain point is modeled by a certain probability density function $\mathcal{F}$. The network designer is allowed to choose an arbitrary set of drop points, and to release an arbitrary number of sensors at each point. Given this setting, we consider the problem of determining the best performing strategy among a certain set of grid-like strategies that reflect the (one- or two-dimensional) symmetry of the region to be monitored. The best performing deployment strategy is such that (1) the DoC requirement is fulfilled and (2) the total number of deployed nodes $n$ is minimum. We study this problem both analytically and through simulation, under the assumption that $\mathcal{F}$ is the two-dimensional Normal distribution centered at the drop point. The main contribution of this paper is an in-depth study of the inter-relationships between environmental conditions, DoC requirement, and cost of the deployment.

**Keywords**: wireless sensor networks, sensor dropping, deployment strategy, network coverage, network connectivity

## 1 Introduction

Wireless Sensor Networks (WSNs) are expected to revolutionize the way natural phenomena and human activities are monitored in the next few years. By connecting tiny, smart sensor nodes by means of wireless transceivers, large WSNs

[*] Part of this work appeared in Proc. IEEE/ACM Symposium on Information Processing in Sensor Networks (IPSN), 2005.

[†] Univ. of Modena e Reggio Emilia, Modena – Italy, e-mail: `leoncini@unimore.it`

[‡] Istituto di Informatica e Telematica del CNR, Via G.Moruzzi 1, 56124 Pisa – Italy. e-mail: `g.resta, paolo.santi@iit.cnr.it`.

can be formed and used in several application scenarios: smart home environments, intrusion detection, tracking of animal movements/behaviors, large-scale environmental monitoring, and so on [8, 12, 14, 15].

Typically, a WSN used is designed to provide a certain level of QoS, which essentially measures the degree of spatio/temporal accuracy required by the particular application[1]. In this paper, we focus on one such spatial measure, termed the *degree of coverage*, which evaluates the percentage of a given region sensed by a set of network-connected sensors. We will formally define this concept in Section 3. While the degree of coverage is just one of the QoS measures that one would like to consider (others are, e.g., delay, throughput and packet loss rate), it is our opinion that the approach taken in this paper can be suitably adopted to investigate other spatial and/or temporal parameters, such as the exposure.

The degree of coverage (hereafter, DoC), as well as other QoS measures provided by a sensor network, depends heavily on the infrastructure, i.e., on the number and positions of the sensors used to monitor the area. Thus, a fundamental step in the network planning stage is to define the node deployment strategy. The strategy used to deploy WSNs depends on the application considered: when the environment is sufficiently known, not hazardous, and relatively small, sensors can be placed manually in pre-determined positions. This strategy provides full control on the node placement, allowing a careful choice of the optimal network topology at the design stage. However, manual sensor placement is impossible in many situations. In fact, the cost of sending human or robot operators in the geographical region to be monitored can be enormous, especially if the region is vast. In other cases, it can be physically impossible to send the operator in the monitored area due to a highly hostile environment.

In situations where manual deployment is not feasible, a typical alternative is to drop sensors from a moving vehicle, such as an airplane or a helicopter. This deployment strategy results in a somewhat random node distribution, in which the human operator can only control the sensor drop point(s). Since the network designer has only a partial control on sensor positions, estimating the expected DoC provided by the deployed network becomes a fundamental and challenging task.

In this paper, we consider the following network planning problem. We are given a certain region $R$, either one- or two-dimensional, to be monitored, and a certain DoC requirement to be fulfilled (e.g., a certain percentage of the area to be covered). Sensors are assumed to be dropped by a moving vehicle. The spatial distribution of the sensors released at a certain drop point is accurately modeled by a certain probability density function $\mathcal{F}$. The network designer is allowed to choose an arbitrary set of drop points within $R$, and to release an arbitrary number of sensors at each point. Given this setting, we consider the problem of determining the *best drop strategy* among a set of possible strategies,

[1] In many application scenarios, the WSN should be able to provide several levels of QoS depending on the events currently going on in the monitored region. However, at the network planning stage the WSN must be designed as if the most stringent QoS requirement were to be met.

i.e., the strategy such that the DoC requirement is met, and the total number of deployed nodes $n$ is minimized. We concentrate on classes of strategies that take the symmetry of the region into account (i.e., they are center-symmetric), in both the one- and two-dimensional setting. Further, we assume that $\mathcal{F}$ is the two-dimensional (even if the region $R$ to monitor is one-dimensional) Normal distribution, and we investigate both analytically and through simulation the (environmental) conditions under which a certain deployment strategy perform better than the others. We also investigate through simulation for which range of values of the dispersion factor (std of the Normal distribution at the release points) partially controlled deployment of sensors turns out to be more effective (in terms of number of deployed sensors needed to achieve a certain DoC requirement) than uncontrolled (random uniform) deployment.

The rest of this paper is organized as follows. In Section 2, we survey related work and highlight our original contributions. In Section 3, we formally define the network deployment problems considered in this paper. In Section 4, we consider the problem of covering a certain one-dimensional region, while in Section 5 we consider the two-dimensional setting. Finally, Section 6 concludes.

## 2 Related work and our contribution

Sensor network deployment strategies have been investigated in several recent papers. A great deal of work has been devoted to study strategies for WSNs used for intrusion detection. In [9], Meguerdichian et al. introduce the concept of exposure of a WSN, which is a measure of how well an object, moving on an arbitrary path within the monitored area, can be observed by the sensor network over a period of time. The higher the exposure, the more likely it is to detect a target moving in the sensor field. In the paper, the authors analyze the exposure of several regular and random node deployments. The problem of computing the exposure provided by an arbitrary WSN has also been addressed [10, 17]. The problem of optimal sensor placement for target location in a grid-like sensor field has been investigated in [2]. The authors solve the problem using integer linear programming. This clearly implies that computing the optimal sensor placement is computationally infeasible for even moderate size networks.

Several authors have also investigated multi-step deployment strategies, which are based on the idea of deploying groups of sensors incrementally, possibly using the information provided by the sensors already on the field to guide the deployment of the next group of nodes. This idea has been exploited in particular in the context of multi-robot exploration. In this scenario, sensors are mounted on mobile robots, and the goal of the robot team is to form a network with certain characteristics. For instance, in [4] Howard et al. propose an incremental deployment algorithm in which sensors (robots) are placed one at a time, and the location of the next sensor is calculated based on the information provided by the robots already on the field. Other proposals along this line of research are [5, 19]. A weakness of this approach is that it can be applied only in restricted application scenarios, and that node deployment is a time consuming process.

Another approach to multi-step node deployment suitable for clusterized WSNs has been proposed in [23]. In this approach, sensors are initially randomly deployed. After initial deployment, the sensors communicate their location and ID to the cluster-head. Given this information, the cluster-head executes an algorithm based on “virtual forces”, which computes a new position for every node in the cluster. New positions are chosen in such a way that network coverage is maximized and node movements are minimized. Once the new positions have been computed and communicated to the nodes, a one-time movement is carried out, and the sensors are re-deployed to these positions. Although interesting, this approach is feasible only in a restricted set of application scenarios, since it is based on very demanding assumptions (location-awareness and autonomous sensor mobility).

A multi-step random deployment strategy is proposed also in [3]. The goal of the designer is to deploy a WSN that guarantees a certain level of exposure while minimizing the cost, which in turn depends on the number of deployed nodes and on the number of sensor deployments. A maximum of $M$ sensors can be positioned on the field, but node positioning is a totally random process. The basic idea is simple: only part of the available sensors are deployed first, and the sensors report their position to the base station. Given node positions, the base station computes the network exposure, and verifies whether the desired level of exposure is achieved. If not, additional sensors are deployed, and the process is repeated until either the level of coverage is satisfactory, or all the sensors have been deployed. In the paper, the authors analyze the infrastructure cost for varying number of sensors deployed in each step.

Contrary to the approaches discussed so far, the emphasis in [16] is on routing protocols for sensor networks. In particular, the authors investigate the effect of infrastructure decisions (number of sensors and deployment strategy) on the performance of routing protocols such as DSR, DSDV and AODV. They consider different deployment strategies (grid-like and random), and several performance metrics. The main finding of the paper is that by simply deploying more sensors, network performance can be harmed. Thus, the network infrastructure must be carefully optimized to fulfill the application requirements.

Other papers focused on optimizing the performance of the WSN *after* deployment. In particular, papers [18, 20, 22] present distributed coordination algorithms for alternating nodes’ sleeping times in such a way that (1) the coverage of the monitored region is guaranteed and (2) the network remains connected. These approaches rely on the implicit assumption that node density is very high, and many nodes can be actually shut down without impairing coverage/connectivity.

The study presented in this paper differs from previous work in several ways. With respect to other studies, our model is less demanding on the sensors: in particular, we do not assume location-awareness nor autonomous mobility. Thus, our results can be applied in a wider range of scenarios. More importantly, to the best of our knowledge, we are the first to addresses *partially controlled* network deployment: the network designer can choose the drop points but, once the sensors are dropped, their exact location cannot be controlled (random placement). This is in sharp contrast with the current literature, which considers only *totally controlled* or *completely uncontrolled* deployment strategies. While totally controlled deployment is feasible in some application scenarios, completely uncontrolled deployment seems overly pessimistic. We want to stress that current investigations on random deployments are based on the assumption of uniform node distribution [3, 16, 23], which seems to be little representative of real world scenarios.

In our approach, we assume that nodes are concentrated around the drop points according to a two-dimensional Normal distribution, where the amount of this concentration is modeled by the parameters $\sigma_x$ and $\sigma_y$ of the distribution (which we call the *dispersion factors*). Admittedly, assuming that nodes are normally distributed around the drop points might be not realistic in some situations. However, we believe that normal deviates are more adequate that uniform ones to model real world conditions. In particular, the dispersion factors are very important, since they provide a handle that can be used to account for different environmental conditions. In turn, these conditions determine the degree of control the network designer has on the final node deployment: the higher the dispersion factors, the less control the designer has on the final node positions.

In the two-dimensional case, we assumed $\sigma_x = \sigma_y = \sigma$. This allowed us to ease the mathematical technicalities in the theoretical analysis reported in Section 5.1. In general, assuming the same dispersion along the two axes can be seen as a limitation, although we do not expect substantially different results with respect to the ones obtained. In the one-dimensional case, however, the dispersions along the $x$ and $y$ axes play a very different role. For this reason, we investigated the more general case where $\sigma_x$ and $\sigma_y$ can be different.

For the one-dimensional setting, we present a theoretical analysis of two deployment strategies in a restricted deployment problem, and formally show that which one of the two strategies performs better depends on the dispersion factor. We also estimate through extensive simulation the minimal node density necessary to fulfill a certain DoC requirement under various environmental conditions. Finally, we consider the slightly different problem of guaranteeing barrier coverage, and estimate the number of sensor to be deployed for a randomly moving intruder to get detected with a given target probability.

For the two-dimensional case, we first theoretically prove a result similar to the one proved for the one-dimensional setting. More specifically, we are given a region $R$ and a division of $R$ into square cells, and we consider dropping $k$ sensors at the center of each cell vs dropping $4k$ sensors at the unique common point of four cells. Although the number and choices of drop points are virtually infinite, the two scenarios considered are nonetheless very important, since they are the minimum drop point strategies among those that satisfy a natural 2D symmetry constraint. We investigate the probability that any of the four cells receives at least one sensor as a function of $\sigma$. Similarly to what happens in the one-dimensional case, we prove that there is a value $\bar{\sigma}$ such that, for $\sigma < \bar{\sigma}$ the 4 drops approach is better, while for $\sigma > \bar{\sigma}$ the 1 drop solution is to be preferred. We also characterize, under both scenarios, the expected number of sensors that must be dropped in order to have at least one sensor *in each cell*, finding a matching tradeoff result. This is the first theoretical proof that *the optimal network deployment strategy depends on the environmental conditions.*

We also perform an extensive simulation-based investigation of a class of grid-based deployment strategies. The main finding of this investigation is that, given a value of $\sigma$ and a certain DoC requirement, the best drop strategy can be identified among the set of candidate strategies. Furthermore, the results of our simulations suggest that in many practical cases *the best deployment strategy depends only on the environmental conditions, and not on the desired DoC.* Finally, the results suggest that, for practical purposes, the 'finer' strategy (i.e., dropping sensors at the center of each cell considered for coverage purposes) gives 'close to optimal' results independently of the environmental conditions (dispersion factor).

Another observation which we can derive from the simulation results is that the effect of an increased DoC on the infrastructure (number of sensors to deploy) heavily depends on the environmental conditions: for small values of $\sigma$, node deployment can be controlled quite carefully, and relatively few sensors are sufficient to meet even very stringent DoC requirements. On the contrary, when the dispersion factor is quite high, a relatively modest increase in the DoC requirement results in a dramatic increase of the number of deployed nodes.

# 3 Preliminaries

We consider a scenario in which sensors are dropped from a moving vehicle, e.g. an airplane or helicopter, and investigate the problem of monitoring both one- and two-dimensional regions. As typical applications, we mention the detection of moving vehicles along a road, or the monitoring of international borders to detect intruders for the one-dimensional case, and environmental monitoring of a vast geographical region or intrusion detection for the two-dimensional case.

In both the one- and two-dimensional case, the deployed network must satisfy a DoC requirement, which specifies the accuracy of the monitoring provided by the network. In particular, we define DoC as follows.

In case of a one-dimensional region $R$, we say that a point $x \in R$ is covered iff the closest sensor is within distance $r_s$ from $x$, where $r_s$ is the sensing range of the nodes. We assume that all the nodes have the same sensing range. We are given a one-dimensional region $R$ of a certain length $l$, and the goal is to cover at least a fraction $q$ of $R$ with a certain target probability $p$. This DoC requirement is summarized as $DoC(q, p)$ in the following.

The definition of DoC in the two-dimensional case is slightly different. Let $R$ be the two-dimensional geographical region to be monitored. $R$ is divided into a number of non-overlapping subregions (also called *sensing cells*, or *cells* for short), which are used to determine the requirement on the sensing coverage provided by the network. In particular, we consider a cell to be *covered* if at least one sensor resides in the cell.

To ease presentation of results, in the following we will make two simplifying assumptions:

- $R$ is a square region of side 1 Km;
- $R$ is divided into equally-sized square cells of side $s$, for some parameter $s$.

We remark that these assumptions are made only to simplify the presentation of our results. In principle, the techniques presented in this paper can be applied to arbitrary shapes of the deployment region and of the cells. However, the shape of the deployment region can have a significant influence on the number of sensors needed to achieve a certain DoC level, hence the results presented in the following (especially for what concerns simulations) cannot be immediately applied to different shapes of the deployment region or of the cells.

In our model, the value of $s$ (and consequently, the total number of cells in $R$) determines the *sensing granularity* required by the particular application. For instance, if sensors are used to monitor temperature, a smaller value of $s$ results in a finer temperature field generated by the network. Thus, parameter $s$ can be seen as an alternative to the popular concept of sensing range which is used in the one-dimensional case. The motivation for using sensing granularity instead of sensing range to model network coverage lies in the fact that this choice eases considerably both the theoretical and the simulation-based analysis of the two-dimensional network deployment problem at hand.

A typical DoC requirement used in our analysis for the two-dimensional case is: "Given a certain sensing granularity $s$, at least a fraction $q$ of the cells are covered with a target probability $p$". For brevity, we denote this specific requirement with $DoC(s, q, p)$.

The following Theorem, whose immediate proof is omitted, states the relation between the sensing granularity $s$ as defined here and the sensing range $r_s$.

**Theorem 1.** *Assume $R$ is divided into non-overlapping square sensing cells of side $s$, and that the sensing range of the dropped sensors is $r_s$. If at least a fraction $q$ of the cells are covered and $r_s \geq s\sqrt{2}$, then at least a fraction $q$ of the area of $R$ is covered.* □

In words, Theorems 1 states that, under the hypothesis that $r_s \geq s\sqrt{2}$, $DoC(s, q, p)$ is a stronger condition than requiring that at least the same fraction of area is within the range of at least one sensor with probability $p$. Thus, our results can be easily restated in terms of sensing range, instead of sensing granularity.

It is also worth mentioning the relation between the notion of DoC and the well-known notion of exposure [9]. We recall that exposure aims at measuring the likelihood of an intruder following a certain path in the monitored area being detected by the WSN, and is computed by integrating the distance of the intruder to the closest sensor along the path. Assume a certain DoC requirement is fulfilled by a WSN, e.g., that at least 90% of the cells contains at least one sensor with high probability. It is then easy to compute a lower-bound to exposure by assuming the uncovered cells are located in the worst-possible manner (i.e., they are all contiguous), and then compute the exposure of an intruder traversing the uncovered region. This approach is likely to give a crude underestimate of exposure. An average case approach would consider a random pattern of uncovered cells (the amount of uncovered cells is given by the DoC requirement), and compute the exposure of a 'random' path within the monitored area.

Besides network coverage, we are also interested in ensuring network *connectivity*, i.e. the ability of a node to communicate (possibly using multi-hop paths) with any other node in the network. This property is fundamental in order to guarantee that *any* sensor node can be used to acquire a global view of the monitored area. This way, a mobile user connected to any of the sensor nodes can obtain a complete view of $R$. In case the information generated by the WSN must be sent to a fixed base station, we must also guarantee connectivity between the base station and some of the network nodes. If the base station is within $R$, the problem of connecting the WSN to the base station can be reduced to the problem of ensuring connectivity in the original WSN with one additional node. This node, which lies in a fixed position, mimics the base station, and it could have a larger transmitting range with respect to sensor nodes. If the base station lies on the boundary of $R$, the problem of connecting the WSN to the base station can be easily solved by dropping few additional sensors in the sub-region of $R$ which is closest to the base station. For these reasons, and for clarity of presentation, in this paper we focus our attention on ensuring connectivity of the WSN only.

In the remainder of this paper we focus our attention on coverage only for the following reasons. First of all, from a theoretical standpoint, coverage implies connectivity under many circumstances. This occurs, for instance, if the nodes' transmitting range $r$ satisfies $r \geq 2r_s$, the requested coverage is 100%, and the area being sensed is bounded and convex [18, 22]. In the next section, we observe that this bound can be generalized to arbitrary coverage of a one-dimensional region $R$. Also, the convexity requirement is not necessary under special region partitionings like the square tessellation we adopt in this paper for the two-dimensional case: in this setting, topological connectedness is sufficient to ensure that coverage implies connectivity. Under a more practical viewpoint, the simulation results presented in the following show that network connectivity is ensured when the required percentage of the region to be covered is large (this is true in both the one- and two-dimensional case), as it is typically the case in practical situations. Thus, we will explicitly deal with network connectivity only in our simulation study, where we adopt the following simple (yet widely accepted) rule to generate the communication graph: there exists a bi-directional wireless link between sensors $u$ and $v$ if and only the sensors are at distance at most $r$, where $r$ is the nodes' transmitting range. We assume that all the nodes have the same transmitting range.

Taking connectivity into account, our typical DoC requirement $DoC(s, q, p)$ (or $DoC(q, p)$) can be restated as follows: "Given a certain sensing granularity

$s$ (or sensing range $r_s$), require that at least a fraction $q$ of the cells are (or at least a fraction $q$ of $R$ is) covered with a target probability $p$, and that all the sensors that contribute to the coverage form a unique connected component".

## 4 One-dimensional case

In this section we consider the following problem:

**Definition 1** (1D Sensor Dropping Problem(1-SDP)). *We are given a one-dimensional region $R$ of length $l$ to cover, and a DoC requirement $DoC(q, p)$. We consider different shapes of $R$ (straight line, and two types of curved lines). Sensor nodes are dropped from a moving vehicle, and can be released at arbitrary points in $R$. We assume that when sensors are dropped at a certain point $(x, y) \in R$, their spatial distribution is accurately approximated by a two-dimensional probability density function (pdf) $\mathcal{F}$. Given this setting, which is the minimum number of nodes to drop in order to fulfill $DoC(q, p)$?*

Note that in the above definition the region $R$ to be covered is one-dimensional, but node dispersal is assumed to be two-dimensional. We believe that above definition captures the salient features of application scenarios such us tracking of vehicles moving on a road or monitoring of international borders, in which the region $R$ to be covered is essentially one dimensional, but node dispersal cannot be confined within $R$.

We start by stating a generalization of the well-known result that coverage implies network connectivity when $r \geq 2r_s$ [18, 22]. We recall that the latter holds under the assumption that the region $R$ to be monitored is entirely covered.

**Proposition 1.** *Let $R$ be a straight line of length $l$. If at least a fraction $q$ of $R$ is covered and $r \geq 2r_s + (1 - q)l$, then the resulting network is connected.*

*Sketch.* Let us consider an arbitrary node deployment such that at least a fraction $q$ of $R$ is covered. It is easy to see that the longest uncovered segment in $R$ has length at most $(1 - q)l$. This implies that there is a region $S$, such as the one depicted in Figure 1, that does not contain any node. The lower bound on $r$ results by computing the diameter of $S$, which corresponds to the distance between points $s$ and $d$. It is immediate to see that this distance is $2r_s + (1-q)l$, and the proposition follows. □

Proposition 1 can be used to set the nodes' transmitting range depending on the sensing range and the required coverage. For instance, if we have to monitor a 1km road, the sensing range is 10m, and we have to cover at least 90% of the road, then setting $r = 120$m guarantees that, if the coverage requirement is met, then the resulting network is connected. However, Proposition 1 accounts for the worst-case scenario, and a much lower setting of the transmitting range is usually sufficient in practice (see Section 4.2).

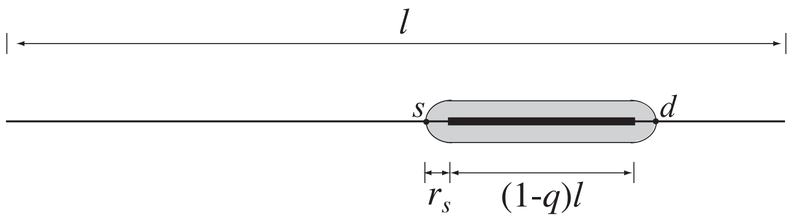

Figure 1: Illustration used in the proof of Proposition 1.

## 4.1 Theoretical analysis

In this section we consider a restricted version of 1-SDP under the simplifying assumption that node dispersal is one-dimensional. More specifically, $\mathcal{F}$ is assumed to be the one-dimensional Normal distribution centered at the drop point and with std $\sigma$.

Analogously to the two-dimensional case which will be analyzed in the following, we assume that the region is partitioned into cells (segments) of the same length, and that the DoC requirement is expressed in terms of covered cells (cells containing at least one sensor). The goal of the analysis is to compare two deployment strategies (dropping sensors at the center of each cell, or at the intersection between two adjacent cells), and to determine under which condition one strategy performs better than the other.

An analytical approximation of the coverage probability of a region $R$ partitioned into several adjacent cells can be obtained by using pairs of adjacent cells as building blocks. In particular, we compare two experiments: $(a)$ $k$ sensors are dropped at the center of each cell, and $(b)$ $2k$ sensors are dropped at the boundary between the two cells; i.e., the same number of sensors are dropped, but with a different strategy.

Assuming for sake of clarity unit length cells, the probabilities that in case (a) a sensor dropped at the center of one cell ends up in that cell ($p_0$), in the adjacent cell ($p_1$), or in neither of them ($q$), are

$$p_0 = \frac{1}{\sqrt{2\pi}\sigma}\int_{-\frac{1}{2}}^{\frac{1}{2}} e^{-\frac{x^2}{2\sigma^2}}\,dx\,, \quad p_1 = \frac{1}{\sqrt{2\pi}\sigma}\int_{\frac{1}{2}}^{\frac{3}{2}} e^{-\frac{x^2}{2\sigma^2}}\,dx\,, \quad q = 1 - p_0 - p_1\,,$$

respectively. By properly combining these values it is possible to explicitly compute (at least for small values of $k$) the probability $Pa(k)$ that, dropping $k$ sensors at the center of each cell, both cells receive at least one sensor. For example, $Pa(1) = p_0^2 + p_1^2$ and $Pa(2) = p_0^4 + 4p_0^3(p_1+q) + 4p_0^2(p_1+q)^2 + 4p_0p_1(p_1+q)^2 + p_1^2(p_1+2q)^2$. For larger values of $k$ the expression of $Pa(k)$ can be computed using a CAS such as Mathematica$^{\text{TM}}$.

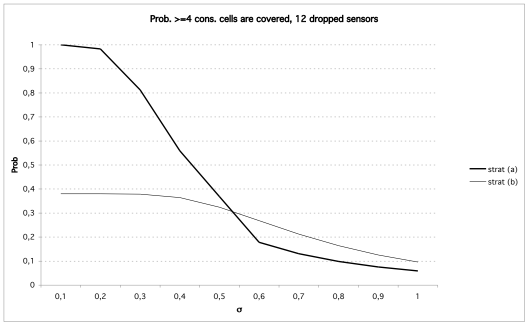

Figure 2: Probability that at least 4 consecutive cells are covered with at least one cell. The total number of dropped sensors is 12.

Analogously, in case (b) the probability that a sensor dropped at the intersection point between two cells ends up in the left (or in the right) cell is

$$p_l = p_r = \frac{1}{\sqrt{2\pi}\sigma} \int_0^1 e^{-\frac{x^2}{2\sigma^2}}\, dx\,,$$

while $1 - 2p_l$ is the probability that the sensor falls outside the two cells. As in case (a), a CAS has been used to explicitly obtain probability $Pb(2k)$ that, dropping $2k$ sensors at the intersection point between two cells, both cells receive at least one sensor. In particular, $Pb(2) = 2p^2$ and $Pb(4) = 2p^2(7p^2 - 12p + 6)$.

As an example of how the value of $\sigma$ can affect which one between strategies $(a)$ and $(b)$ is better, we have considered a scenario made of 6 cells (3 pairs of cells). Using the values computed above and standard techniques in probability theory we have evaluated, as a function of $\sigma$, the probability that at least four consecutive cells among the six receive at least one sensor when 2, 4 or 6 sensors are dropped on each pair of cells according to (a) or (b). The results are shown in figures 2–4. As seen from the figures, there exists a threshold value $\sigma'$ of $\sigma$ (around 0.5) such that for $\sigma < \sigma'$ strategy $(a)$ is better than strategy $(b)$, and the opposite holds for $\sigma > \sigma'$. Figures 2–4 clearly show the effect of environmental conditions (value of the dispersion factor $\sigma$) on the choice of the best sensor deployment strategy.

## 4.2 Simulation study

We have studied 1-SDP also through extensive simulation. In our simulations, we assumed that node dispersal is governed by the two-dimensional Normal distribution centered at the drop point, with standard deviation $\sigma_x$ (respectively, $\sigma_y$) along the $x$ (respectively, $y$) axis. The standard deviation along the $x$ and

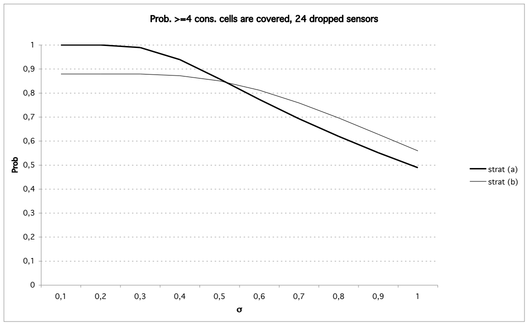

Figure 3: Probability that at least 4 consecutive cells are covered with at least one cell. The total number of dropped sensors is 24.

$y$ axis can be used to account for environmental conditions that influence node dispersal, such as wind, velocity of the vehicle used to drop the sensors, and so on.

The region to be covered is a line of length 1km, with three different shapes: line segment, a 90° arc, and a 180° arc (see Figure 5). We have chosen these shapes for investigation because we believe they are sufficiently representative of the main "building blocks" composing the more complex shapes roads and borders typically have.

The dropping strategy is as follows: when dropping $n$ sensors, we compute $n$ equally spaced drop points along $R$, and we drop one sensor at each drop point. We have also considered alternative dropping strategies, such as having $n/2$ equally spaced drop points and dropping 2 sensors at each point (in case $n$ is even), having $n/3$ equally spaced drop points and dropping 3 sensors at each point (in case $n/3$ is integer), and so on. When the expected node dispersal is high, having fewer drop points might turn out to be a better choice (see below).

In our simulations the node sensing range is set to 10m, and the transmission range is set to 25m. Note that, according to Proposition 1, this value of the transmitting range is sufficient to ensure connectivity when the requested coverage is at least 99.5%. However, setting $r = 25$m turned out to be sufficient to ensure connectivity in all our simulation experiments, where the requested coverage was 80% or higher.

In the first set of experiments, we have evaluated how the minimum number $n_{min}$ of nodes to be deployed in order to fulfill $DoC(q, 0.95)$ varies as $q$ ranges from .80 to .99. We recall that $DoC(q, 0.95)$ means that we want to cover at least a fraction $q$ of $R$ with probability at least 0.95. For a given DoC requirement, we have considered an increasing number of deployed nodes $n$, and performed 1000 random experiments for each value of $n$, counting the ratio of experiments

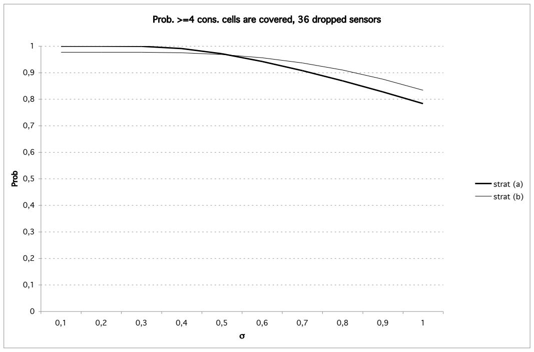

Figure 4: Probability that at least 4 consecutive cells are covered with at least one cell. The total number of dropped sensors is 36.

that resulted in a successful configuration (i.e., when at least a fraction $q$ of the region is covered). For each value of $n$, we have considered different dropping strategies as explained above. The returned value $n_{min}$ is the minimum value of $n$ such that the DoC requirement is fulfilled (i.e., when at least 95% of the experiments resulted in a successful configuration) for at least one dropping strategy. The simulator also returns the optimal(s) dropping strategy(ies).

We have run experiments considering four possible configurations for the values of $\sigma_x$ and $\sigma_y$, corresponding to each standard deviation taking either a low or a high value. The low value corresponds to setting the standard deviation to $r_s = 10$m, while the high value corresponds to setting the standard deviation to $2r_s$. We recall that, using the Normal distribution, the probability of a sensor landing within distance $2.5\sigma$ from the drop point is above 0.95. Thus, setting $\sigma_x = \sigma_y = r_s$ corresponds to a low dispersion scenario, in which sensor nodes land within a circle of radius 25m centered at the drop point with probability above 0.95. Conversely, setting $\sigma_x = \sigma_y = 2r_s$ corresponds to a high dispersion scenario, in which sensor nodes land within a circle of radius 50m centered at the drop point with probability above 0.95. The landing regions (with 0.95 confidence) of sensors correspondent to the various settings of $\sigma_x$ and $\sigma_y$ used in our experiments are shown in Figure 6. When clear from the context, we simply write $\sigma_x = 1$ instead of $\sigma_x = r_s$, and $\sigma_x = 2$ instead of $\sigma_x = 2r_s$ (similarly for $\sigma_y$).

The results of this first set of experiments are reported in Figure 7 (line segment), Figure 8 ($90°$ arc), and Figure 9 ($180°$ arc). In case of the straight line, we have repeated the experiments with a 2km region, obtaining similar results: essentially, $n_{min}$ is doubled with respect to the results reported in Figure 7.

From the results it is seen that the value of $n_{min}$ as a function of $q$ has similar

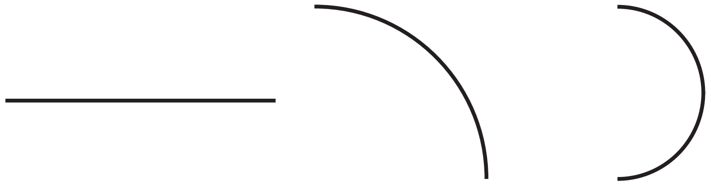

Figure 5: 1-D regions considered in our simulations.

trends for the various shapes and dispersion settings considered in our experiments: it increases slowly with $q$ until a certain critical value $\bar{q}$ is reached, after which the increase becomes sharp. This means that, for values of $q$ below $\bar{q}$, a relatively modest increase in the number of deployed nodes is sufficient to significantly increase coverage; beyond this critical value, coverage can be improved only at the expense of a considerable increase in the number of deployed nodes. For instance, in case of a straight line with high dispersion ($\sigma_x = \sigma_y = 2$), the number of deployed nodes must be increased from about 600 to about 800 to increase coverage from 95% to 98%; i.e., a 3% increase in coverage comes at the expense of a 33% increase in the number of deployed nodes. On the other hand, increasing the number of nodes from 400 to 600 is sufficient to increase coverage from 80% to 95%; i.e., a 50% increase in the number of nodes results in a 15% increase in coverage.

The value of $\bar{q}$ depends on the dispersion factor: it ranges from about 95% (low dispersion), to about 85% (high dispersion). As it could be expected, the shape of $R$ plays a role only when $\sigma_x \neq \sigma_y$: while the value of $n_{min}$ for a certain $q$ with $\sigma_x = 1$ and $\sigma_y = 2$ is significantly higher than the same value with $\sigma_x = 2$ and $\sigma_y = 1$ in case of a straight line, the two curves almost coincide in case of both the 90° and the 180° arc. This is due to the fact that symmetry plays a fundamental role in case of the curved lines, and the dispersion on either the $x$ or $y$ axis affects coverage similarly. Conversely, in a straight line the dispersion along the $y$ axis has a stronger effect on coverage: while dispersion along the $x$ axis might cause nodes to land forward or backward with respect to the drop point, dispersion along the $y$ axis might cause nodes to land too far from the monitored region, thus having several nodes not contributing at all to coverage. This explains why in case of a straight line the curve with $\sigma_x = 1$ and $\sigma_y = 2$ is very close to the curve with $\sigma_x = \sigma_y = 2$.

Observe that the phenomenon described above is mitigated if the sensing range is increased, since nodes which are further away from $R$ can still contribute to coverage. Indeed, we have observed the opposite phenomenon (i.e., the curve with $\sigma_x = 1$ and $\sigma_y = 2$ is *below* the curve with $\sigma_x = 2$ and $\sigma_y = 1$) if $r_s$ is chosen sufficiently high and the transmitting range is scaled properly (e.g.,

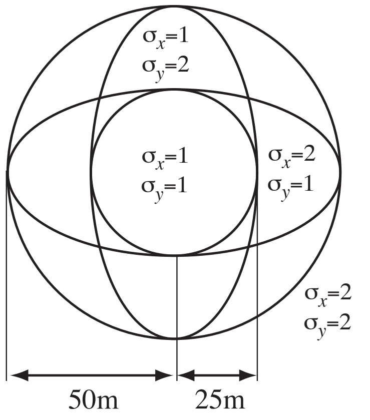

Figure 6: Landing regions of sensors correspondent to the various setting of $\sigma_x$ and $\sigma_y$. The probability of landing within the region is 0.95.

$r_s$ = 50m and $r$ = 120m): in this case, having a relatively high dispersion on the $y$ axis ($\sigma_y$ = 20m) is not detrimental, since most of the nodes contribute to coverage; also, nodes which do not contribute to coverage still contribute to connectivity, acting as bridges for the nodes that actually cover $R$. It then results that having a high dispersion along the $x$ axis is more detrimental, since network partitioning is more likely to occur.

| | | $\sigma_x = \sigma_y = 1$ | $\sigma_x = 1,\ \sigma_y = 2$ | $\sigma_x = 2,\ \sigma_y = 1$ | $\sigma_x = \sigma_y = 2$ |
|---|---|---|---|---|---|
| 80% cov. | straight | 2.71 | 3.56 | 3.24 | 4.04 |
| | 90° arc | 2.72 | 3.49 | 3.50 | 4.04 |
| | 180° arc | 2.68 | 3.54 | 3.42 | 4.00 |
| 95% cov. | straight | 2.95 | 5.61 | 3.44 | 5.92 |
| | 90° arc | 2.95 | 4.64 | 4.62 | 5.94 |
| | 180° arc | 2.95 | 4.60 | 4.65 | 5.92 |
| 99% cov. | straight | 4.58 | 9.20 | 5.12 | 9.79 |
| | 90° arc | 4.55 | 7.81 | 7.77 | 9.80 |
| | 180° arc | 4.56 | 7.65 | 8.05 | 9.79 |

Table 1: Minimal node densities for different coverage and dispersion factors.

Table 1 reports the minimal node density (expressed as number of sensor to be dropped every 10m) necessary to fulfill a certain DoC requirement, with different dispersion values. Note that the optimal density, which can be achieved in case of manual node deployment, is 0.5 nodes/10m (given that $r_s = 10m$). So, the data reported in Table 1 dictate that, for instance, deploying a number of

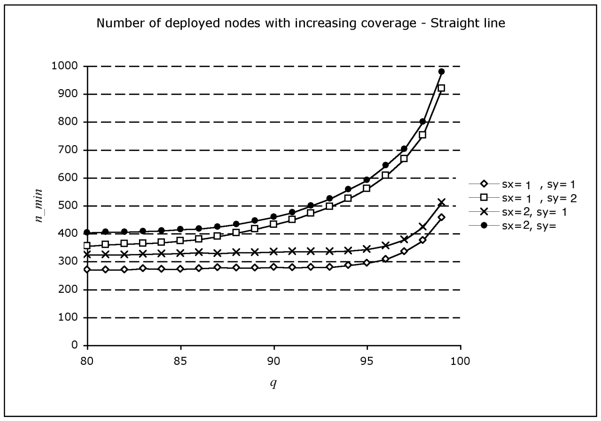

Figure 7: Minimum number of nodes to be deployed in order to fulfill $DoC(q, 0.95)$ with increasing values of $q$. $R$ is a line segment.

nodes which is about 6 times the minimal number of nodes needed with manual deployment is necessary to achieve 95% coverage in case of low dispersion. As for the effect of the the shape of the deployment region, we observe that similar sensor densities for the three shapes are needed when $\sigma_x = \sigma_y$, lower densities in the case of curved shapes are needed when $\sigma_x < \sigma_y$, and the opposite holds when $\sigma_x > \sigma y$.

When considering the best dropping strategy(ies), i.e. the dropping strategy(ies) used to successfully deploy $n_{min}$ nodes, we have observed that dropping one sensor at each drop point (i.e., having $n_{min}$ equally spaced drop points) is the more recurrent best strategy, especially when the dispersion is low. This occurs independently of the shape of $R$. In many cases, different dropping strategies resulted best performing: for instance, the best dropping strategies when $\sigma_x = \sigma_y = 1$, $q = 97$, and $R$ is a straight line are (1) dropping 2 sensors at 168 equally spaced drop points, (2) dropping 6 sensors at 56 equally spaced drop points, and (3) dropping 7 sensors at 48 equally spaced drop points, all resulting in $n_{min} = 336$. The number of sensors to be dropped at each drop point with the optimal strategy resulted always below 10 in case of low dispersion ($\sigma_x = \sigma_y = 1$), and resulted as high as 20 in case of high dispersion ($\sigma_x = \sigma_y = 2$). It is worth noting that, in evaluating $n_{min}$, a significant role is played by the inherent discretization in the number of dropped sensors.

In a second set of experiments, we have evaluated the effect of the dispersion factors on $n_{min}$, given the DoC(0.90,0.95) requirement. The results of these experiments are reported in Figure 10 for the straight line, and in Figure 11 for the 90° arc. The results obtained for the 180° arc are very similar to those obtained for the 90° arc and are not reported. From the figures it is evident the different effect of the dispersion along the $y$ axis in the two cases. As explained above, this is due to the fact that when $R$ is a curve symmetry plays a significant role.

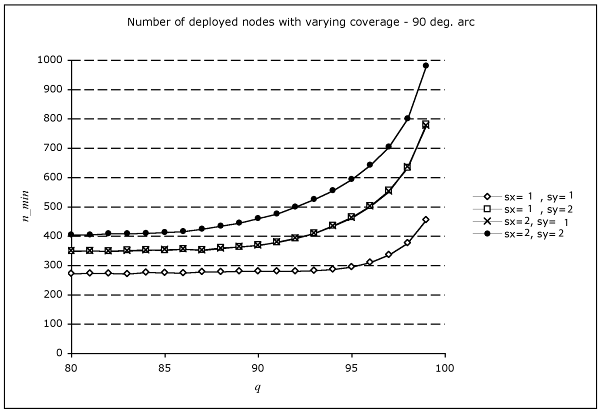

Figure 8: Minimum number of nodes to be deployed in order to fulfill $DoC(q, 0.95)$ with increasing values of $q$. $R$ is a 90° arc.

Before ending this section, we observe that, in case the application scenario is border monitoring, the designer is in general interested at deploying a “barrier” of sensors [7], rather than covering a given (one- or two-dimensional) area. We have performed a set of experiments in which we fixed two points, $s$ and $d$, as the starting and ending points of the border to monitor, and we built a *coverage graph* $G$ in which two sensors are directly connected iff their covering circles intersect. Points $s$ and $d$ act as base stations in this scenario. With this setting, we have investigated the probability of having $s$ and $d$ connected by a path in $G$ when a certain number of sensors are dropped over the $\overline{sd}$ line segment. This problem is essentially that of 1-barrier covering an arbitrary, but sufficiently wide, belt region, i.e., a region in which the “east-west” bounding curves are distant enough from the line $\overline{sd}$ (see [7]). Here, however, we require that the connected path joins two well-defined points ($s$ and $d$) rather than two arbitrary points along the “north-south” bounding curves of the belt region.

We have computed the minimum number of nodes needed (on the average) to obtain $s - d$ connectivity in the coverage graph with probability at least 0.95, where the sensor dropping strategies are the same as in the previous experiments (i.e., sensors are dropped on the line segment connecting $s$ and $d$). The results of our experiments are reported in Table 2. As seen from the table, substantially less sensors can be dropped if the goal is to provide barrier coverage, instead of coverage of the line segment. Actually, line coverage implies barrier coverage, but not vice-versa (see Figure 12). The decrease in the number of deployed sensors can be as high as 61% when $\sigma_x = 1$ and $\sigma_y = 2$.

Summarizing, the main findings of our extensive simulation-based analysis of 1-SDP are the following:

– providing very strong coverage guarantees (say, prob. of coverage above 98%) with partially controlled sensor deployment is very expensive in

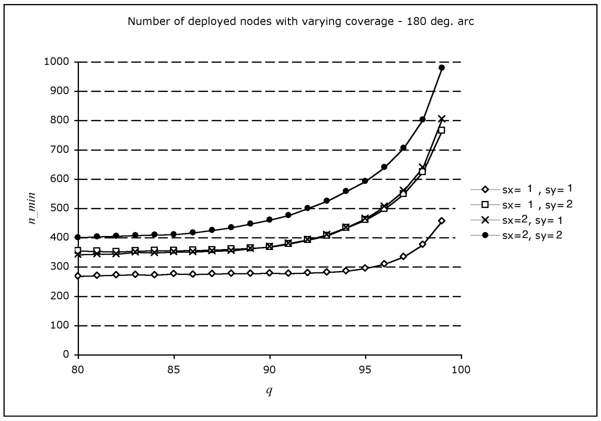

Figure 9: Minimum number of nodes to be deployed in order to fulfill $DoC(q, 0.95)$ with increasing values of $q$. $R$ is a 180° arc.

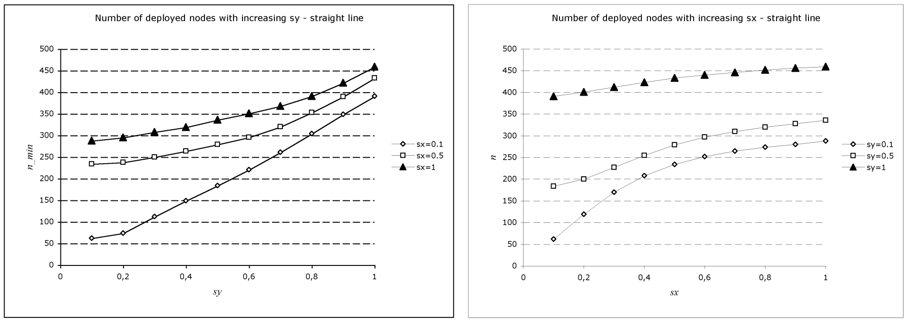

Figure 10: Minimum number of nodes to be deployed in order to fulfill $DoC(0.90, 0.95)$ with different values of dispersion. $R$ is a line segment.

terms of infrastructure cost (number of deployed sensors).

- the ‘finer’ deployment strategy (i.e., dropping one sensor at each drop point) is often the best performing strategy; however, this strategy might be very difficult to achieve or costly in a practical setting, and more ‘coarse’ dropping strategy might be considered.
- a considerable increase in sensor density (2.7-fold to 9.8-fold) with respect to the case of optimal (manual) sensor deployment is needed to provide strong coverage guarantees in case of partially controlled sensor deployment.
- the shape of the region to cover plays a significant role only when the dispersion factor is asymmetric ($\sigma_x \neq \sigma_y$);

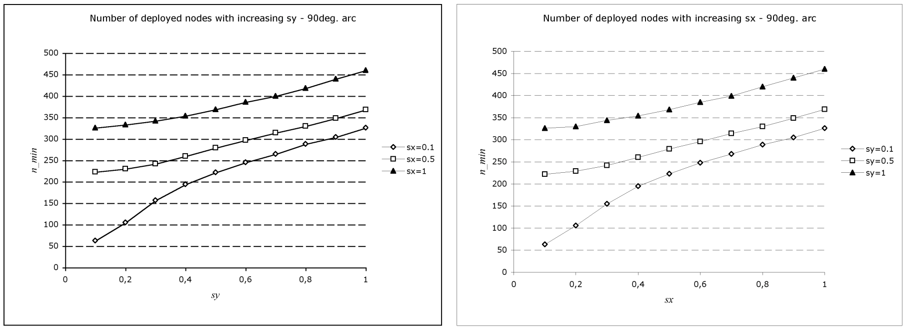

Figure 11: Minimum number of nodes to be deployed in order to fulfill $DoC(0.90, 0.95)$ with different values of dispersion. $R$ is a 90° arc.

| | | $n_{min}$ | |
|---|---|---|---|
| $\sigma_x$ | $\sigma_y$ | *barrier* | *line* |
| 1 | 1 | 420 | 725 |
| 1 | 2 | 555 | 1440 |
| 2 | 1 | 495 | 795 |
| 2 | 2 | 630 | 1510 |

Table 2: Minimum number of nodes to be deployed in order to obtain barrier coverage (*barrier* entry) and coverage of the line segment $\overline{sd}$ (*line* entry). The target probability is 0.95 in both cases.

– relatively few sensors are needed for guaranteeing barrier coverage instead of guaranteeing coverage of a straight line.

# 5 Two-dimensional case

In this section, we analyze the following two-dimensional sensor dropping problem:

**Definition 2** (Sensor Dropping Problem (SDP)). *We are given a square geographical region $R = [0, 1]^2$ and a DoC requirement $DoC(s, q, p)$. Sensor nodes are dropped from a moving vehicle. We assume that the node spatial distribution generated when sensors are dropped at a certain point $(x, y)$, with $0 \le x, y \le 1$, is accurately approximated by a probability density function (pdf) $\mathcal{F}$. Sensors can be dropped at arbitrary locations within R. Given R, $DoC(s, q, p)$ and $\mathcal{F}$, which is the optimal drop strategy, i.e., the strategy such that the DoC requirement is met and the total number of deployed sensors is minimum?*

Solving this problem in the most general formulation is very difficult (mainly because of the uncountable number of different drop strategies to consider), and we leave it as an open problem. In this paper, we simplify the above problem

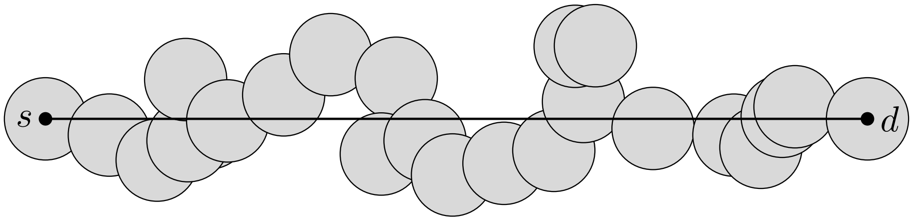

Figure 12: Barrier and line coverage. There is a connected $s - d$ path in the coverage graph, even though the $\overline{sd}$ segment is not covered.

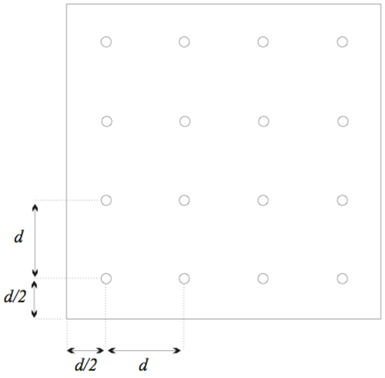

Figure 13: Grid strategy of parameter $d = 0.25$. Drop points are represented by circles. There are $4 \times 4 = 16$ drop points in total.

formulation by restricting the class of possible drop strategies to *grid strategies*, which we now define.

**Definition 3** (Grid drop strategy)**.** *The grid drop strategy of parameter $d$, with $0 < d \leq 1$, corresponds to releasing sensors at coordinates $(\frac{d}{2} + i \cdot d, \frac{d}{2} + j \cdot d)$, for $i, j = 0, \ldots, \left\lfloor \frac{1}{d} \right\rfloor - 1$. The total number of drop points in the grid strategy of parameter $d$ is $\left\lfloor \frac{1}{d} \right\rfloor^2$.*

The grid strategy of parameter $d = 0.25$ is reported in Figure 13. Figure 14 shows the subdivision of $R$ into cells when $s = 0.1$, and four different grid strategies.

In our analysis, we assume that $\mathcal{F}$ is the two-dimensional Normal distribution of parameter $\sigma$ centered at the drop point. More formally, the pdf of the $x$-coordinate of a sensor dropped at point $(\bar{x}, \bar{y})$ is accurately approximated by $\mathcal{N}(\bar{x}, \sigma)$, while the $y$-coordinate is accurately approximated by $\mathcal{N}(\bar{y}, \sigma)$.

Parameter $\sigma$ models the expected geographical dispersion of sensor nodes when dropped at a certain point. For this reason, $\sigma$ is called the *dispersion factor*. For the sake of simplicity, we assume that $\sigma$ does not depend on the particular drop point. In other words, we assume that the pdf that models the

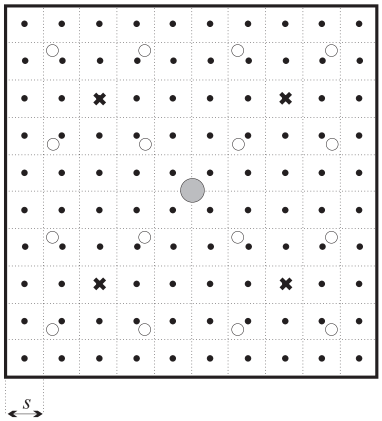

Figure 14: Subdivision of $R$ into square cells when $s = 0.1$. Grid strategies of parameter $d = 0.1$ (black circle), $d = 0.25$ (white circle), $d = 0.5$ (cross) and $d = 1$ (shaded circle) are also shown.

node spatial distribution is the same at each drop point. Finally, we assume that the dispersion along the $x$-axis is the same as the dispersion along the $y$-axis, and that the two measures are independent. Again, the goal of this assumption is only making the presentation of our results clearer. Our techniques can be easily extended to account for different dispersions along the two axes.

We are now ready to define the constrained version of SDP which will be analyzed in this section.

**Definition 4** (Grid Sensor Dropping Problem (GSDP))**.** *We are given a square geographical region $R = [0, 1]^2$ and a DoC requirement $DoC(s, q, p)$. Sensor nodes are dropped from a moving vehicle. We assume that the node spatial distribution generated when sensors are dropped at a certain point $(x, y)$, with $0 \leq x, y \leq 1$, is accurately approximated by the two-dimensional Normal distribution of parameter $\sigma$ centered at $(x, y)$. Given R, $DoC(s, q, p)$ and $\sigma$, which is the best performing grid drop strategy, i.e., the grid strategy such that the DoC requirement is met and the total number of deployed sensors is minimum?*

## 5.1 Theoretical analysis

At first sight, the restricted version of the SDP introduced in the previous section could be investigated using occupancy theory, an important branch of Probability Theory (see, e.g., [6]). The classical occupancy model of randomly throwing $n$ balls into $N$ boxes has a number of well-known applications, and it has been used also in ad hoc networks to investigate properties such as connectivity [13], and the performance of cell-based energy-conservation strategies [1] and of clustering protocols [21]. In our case, we can quite naturally view the sensing cells as the boxes and the sensors dropped as the balls thrown into

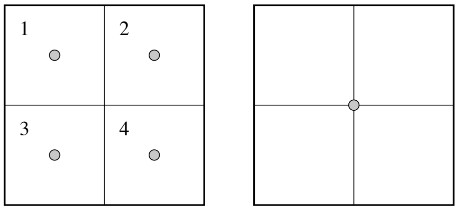

Figure 15: One vs four launches: the small circles indicates the (projection of the) point from which sensors are dropped.

them[2]. However, there are some difficulties that prevent us from using the most studied occupancy models. Consider the launch of a sensor and suppose that $p_i$ is the probability that it lands in cell $i$. Occupancy theory gives us models for the case of equiprobable allocations (which means that the probability of a ball going into box $i$ is the same for all $i$, i.e., $p_i = \frac{1}{N}$, where $N$ is the number of boxes) as well as nonequiprobable allocations (i.e., possibly $p_i \neq p_j$, for $i \neq j$). However, in our case the balls (sensors) are thrown from different places; this makes the elementary probabilities $p_i$ also dependent on a further variable, say $t$, that can be interpreted as a time variable. In other words, we must replace $p_i$ with $p_i(t), t = 1, 2, \ldots$, where $p_i(t)$ is the probability that the $t$-th sensor dropped goes into cell $i$. Moreover, the $p_i(t)$ are not given for free; in fact, each $p_i(t)$ has to be computed by integrating, over the area represented by cell $i$, the probability density function that governs the $t$-th launch. At the very least, for a large number of cells and many drop points, the computation of the basic probabilities $p_i(t)$ can be time consuming.

For the above mentioned reasons, we do not further pursue this line of attack; rather, we devise, using “special purpose” methods, an analytical result that applies to one specific, yet interesting, instance of GSDP.

Suppose we are faced with relatively large values of $\sigma$ with respect to the sensing granularity. Under this circumstance, a simple question that turns out to be analytically tractable is the following: is there any advantage from dropping sensors at the intersection of four cells with respect to dropping (a less number of) sensors over each of the four cells (see Figure 15)?

Consider a square region $R$ of side $4u$ and suppose it is divided in four square cells of side $2u$, for some $u \geq 0$. We consider the following two experiments: (a) four sensors are dropped, one at the center of each cell, and (b) four sensors are dropped at the center of $R$ (Figure 15). We compute the probability that cell 1 receives at least one sensor under (a) and (b); because of an easy symmetry argument, such probability is the same for all the cells. In all cases we assume that the pdf governing the dispersal of the sensors is Normal with mean centered at the drop point (which is always assumed to be at $(0, 0)$ without loss of generality) and standard deviation $\sigma$ along both axes, denoted $N_{O,\sigma}$. We also

[2] The surface outside the deployment region $R$ can be regarded as an additional box collecting the probability of a sensor not landing into $R$.

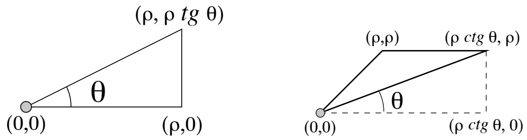

Figure 16: Integration domains within the definition of $C(\rho, \theta)$ and $S(\rho, \theta)$.

assume that dispersions along the $x$ and $y$ axis are uncorrelated. With these assumptions, in polar coordinates we have:

$$N_{O,\sigma}(\rho, \theta) = N_{O,\sigma}(\rho) = \frac{\rho}{2\pi\sigma^2} \mathrm{e}^{-\frac{\rho^2}{2\sigma^2}}. \tag{1}$$

**Case (a).** Assume the four cells are numbered 1 through 4 as in Figure 15. Let $p_{ij}$ denote the probability that a sensor dropped at the center of cell $i$ lands somewhere in cell $j$, $i, j = 1, \dots, 4$, and let $q_{ij} = 1 - p_{ij}$. The probability $p_1$ that cell 1 receives at least one sensor is $p_1 = 1 - q_{11}q_{21}q_{31}q_{41}$. By symmetry, $p_{ij}$ only depends on the relative positions of cells $i$ and $j$. Hence, for instance, $p_{21} = p_{31} = p_{12}$ and $p_{41} = p_{14}$, and we are simply lead to compute $p_{11}, p_{12}$, and $p_{14}$. To this end, for $\theta \in \left[0, \frac{\pi}{4}\right]$ and $\rho \geq 0$, we define the following two functions:

$$\begin{aligned} C(\rho, \theta) &\triangleq \int_0^{\theta} \int_0^{\frac{\rho}{\cos\theta}} N(\rho', \theta') \mathrm{d}\rho' \mathrm{d}\theta' = \\ &= \frac{\theta}{2\pi} - \frac{1}{2\pi} \int\limits_0^{\theta} \mathrm{e}^{-\frac{\rho^2}{2\sigma^2 \cos^2\theta'}} \mathrm{d}\theta', \end{aligned} \tag{2}$$

and

$$\begin{aligned} S(\rho, \theta) &\triangleq \int_{\theta}^{\frac{\pi}{4}} \int_0^{\frac{\rho}{\sin\theta}} N(\rho', \theta') \mathrm{d}\rho' \mathrm{d}\theta' = \\ &= \frac{1}{8} - \frac{\theta}{2\pi} - \frac{1}{2\pi} \int\limits_{\theta}^{\frac{\pi}{4}} \mathrm{e}^{-\frac{\rho^2}{2\sigma^2 \sin^2\theta'}} \mathrm{d}\theta'. \end{aligned} \tag{3}$$

By definition, $C(\rho, \theta)$ (respectively, $S(\rho, \theta)$) is the probability that a sensor dropped at $(0, 0)$ lands somewhere in the triangle $(0, 0), (\rho, 0), (\rho, \rho \tan\theta)$ (respectively, $(0, 0), (\rho, \rho), (\rho \operatorname{cotan}\theta, \rho)$) – see Figure 16.

By combining $C(\rho, \theta)$ and $S(\rho, \theta)$, we can compute the probabilities $p_{11}$, $p_{12}$, and $p_{14}$ as follows.

- $p_{11} = 8C(u, \frac{\pi}{4})$, where $C(u, \frac{\pi}{4})$ accounts for the probability over the area depicted in Figure 17.
- $\frac{1}{2}p_{12} = 2\left(C(3u, \bar{\theta}) + S(u, \bar{\theta}) - C(u, \frac{\pi}{4})\right)$, where $\bar{\theta} = \operatorname{arcos}\left(\sqrt{10}/3\right)$, by the area decomposition suggested in Figure 18.

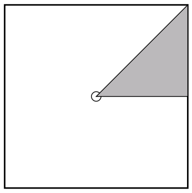

Figure 17: Area that receives $C(u, \frac{\pi}{4})$ of the total probability. Sides have length $2u$.

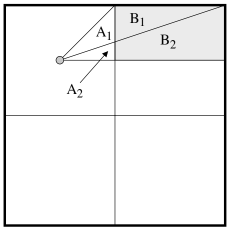

Figure 18: The probability over the light gray shaded area $(B_1, B_2)$ is computed as the probability over $(A_1, B_1)$ $(S(u, \bar{\theta}))$, plus the probability over $(A_2, B_2)$ $(C(3u, \bar{\theta}))$, minus the probability over $(A_1, A_2)$ $(C(u, \frac{\pi}{4}))$.

- $p_{14} = 2\left(C(3u, \frac{\pi}{4}) - C(u, \frac{\pi}{4})) - p_{12}\right)$. In fact, $p_{14}$ can be computed by following the area decomposition suggested in Figure 19.

Substituting the single pieces into $p_1 = 1 - q_{11}(q_{21})^2 q_{41}$ and doing the algebra, we get:

$$
\begin{aligned}
p_1 = 1 - &\left( \frac{4}{\pi} \int_0^{\frac{\pi}{4}} \mathrm{e}^{-\frac{u^2}{2\sigma^2 \cos^2\theta}} \mathrm{d}\theta \right) \cdot \\
&\left( 1 - \frac{1}{\pi} \left( \int_0^{\frac{\pi}{4}} \mathrm{e}^{-\frac{u^2}{2\sigma^2 \cos^2\theta}} \mathrm{d}\theta - \int_0^{\bar{\theta}} \mathrm{e}^{-\frac{9u^2}{2\sigma^2 \cos^2\theta}} \mathrm{d}\theta + \right.\right. \\
&\left.\left. - \int_{\bar{\theta}}^{\frac{\pi}{4}} \mathrm{e}^{-\frac{u^2}{2\sigma^2 \sin^2\theta}} \mathrm{d}\theta \right)\right)^2 \cdot \\
&\left( 1 - \frac{1}{\pi} \left( \int_{\bar{\theta}}^{\frac{\pi}{4}} \mathrm{e}^{-\frac{u^2}{2\sigma^2 \sin^2\theta}} \mathrm{d}\theta - \int_{\bar{\theta}}^{\frac{\pi}{4}} \mathrm{e}^{-\frac{9u^2}{2\sigma^2 \cos^2\theta}} \mathrm{d}\theta \right)\right)
\end{aligned}
$$

**Case (b).** Suppose we drop four sensors at the center of $R$. The probability the any of these goes into cell 1 is $\hat{p}_1 = 1 - (\hat{q}_1)^4$, where $\hat{q}_1$ is the probability

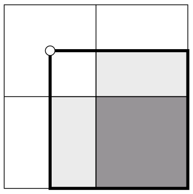

Figure 19: The probability over the dark gray shaded area is computed as the probability over the square with thick sides ($2C(3u, \frac{\pi}{4})$), minus the probability over the light gray shaded areas ($p_{12}$), minus the probability over the inner white square ($2C(u, \frac{\pi}{4})$).

that any given sensor does not land in cell 1. In turn, $\hat{q}_1 = 1 - 2C(2u, \frac{\pi}{4})$, so that (after simple algebra) we obtain

$$\hat{p}_1 = 1 - \left( \frac{3}{4} + \frac{1}{\pi} \int_0^{\frac{\pi}{4}} \mathrm{e}^{-\frac{u^2}{2\sigma^2 \cos^2 \theta}} \, \mathrm{d}\theta \right)^4 .$$

**Comparison.** In Figure 20 we report the probability of cell 1 receiving at least one sensor in the cases (a) and (b) discussed above. The probability is plotted against the ratio $\frac{\sigma}{4u}$, i.e., the dispersion factor is normalized with respect to the side of the deployment region. Clearly, for symmetry reasons, this is also the probability that any other cell receive at least one sensor. The plot confirms the intuition that there must be a value of $\frac{\sigma}{4u}$ (approximately 0.206) beyond which the probability of having at least a sensor in a given cell with the single launch option is higher.

Let us now consider the expected number of sensors that must be dropped in cases (a) and (b) in order to have at least one sensor *in each cell.* We recall that, for independent trials (i.e., launches), each one having the same success probability $p$, the number of attempts that must be made before the first success occurs is a geometric random variable, i.e., the first success occurs at trial $i$ with probability $p(1-p)^{i-1}$. The expectation of a geometric random variable of parameter $p$ is $\frac{1}{p}$.

Let us start with scenario (a). In this scenario, the random experiment is the launch of four sensors, one for each of the four drop points. Let $X_i$, for $i = 1, \ldots, 4$, be the random variable that denotes the minimum number of experiments (each dropping four sensors) after which cell $i$ contains at least one sensor. It is immediate that $X_i$ has geometric distribution of parameter $p_1$, where $p_1$ is defined as above. Note that the $X_i$s are not independent, since the fact that a certain cell is occupied (or empty) has the effect of decreasing (or increasing) the probability that another cell is occupied. However, they have negligible correlation, as the following simple argument shows.

Let $C$ be the random variable that denotes the number of experiments after which *all* the cells are covered with at least one sensor. It is immediate that

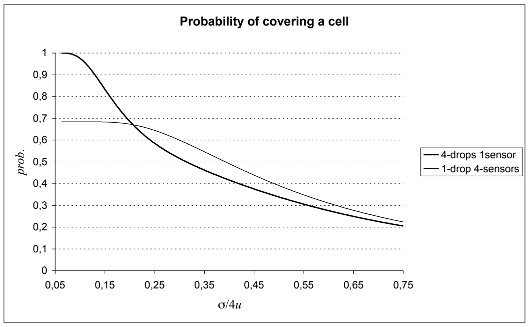

Figure 20: Probability of having at least one sensor in cell 1 under cases (a) and (b) as a function of $\frac{\sigma}{4u}$.

$C = \max_{i=1,\ldots,4} X_i$. Suppose, without loss of generality, that the maximum is achieved at cell 1 and consider the conditional probability $P(X_1 \geq k | X_i \leq k, i = 2, 3, 4)$. How is the distribution of $X_1$ affected by knowing that $X_i \leq k$, for $i = 2, 3, 4$? Actually, this latter information amounts only to saying that 3 out of $4k$ sensors launched have not landed in cell 1, and clearly this information tends to be negligible as $k$ grows. Actually, small values of $k$ are likely to be obtained for small dispersion factors, which in turn imply low correlation as well (but for a different reason: nodes are highly concentrated around the drop points).

The above observation has been confirmed by the simulation results. First, we have derived a formula for $\mathrm{E}[C]$ under an independence assumption of the $X_i$'s. Then, we have generated a large number of data sets and sampled the true expectation. The results are reported in Figure 21.

To derive the formula for the expectation we have first determined the cumulative distribution function of $C$ as follows:

$$Prob(C < k) = 1 - Prob(C \geq k) =$$

$$1 - Prob(\exists i : X_i \geq k) = 1 - (1 - Prob(\forall i : X_i < k))$$

Under the stated independence assumption, we can then write:

$$Prob(C < k) \approx \prod_{i=1}^{4} Prob(X_i < k) \ .$$

Since the $X_i$s are geometric random variables of parameter $p_1$, we have:

$$Prob(C < k) \approx \left( \sum_{j=1}^{k-1} p_1 (1 - p_1)^{j-1} \right)^4 \ . \tag{4}$$

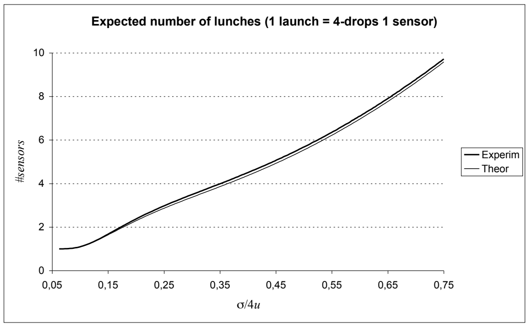

Figure 21: Expected number of launches to have one sensor in each cell under scenario (a) with our formula (Theor) and as resulting from simulations (Exper), as a function of $\frac{\sigma}{4u}$.

Using the cumulative distribution function, we can compute the probability that $C$ equals $k$ as follows:

$$Prob(C = k) = Prob(C < k + 1) - Prob(C < k) \ .$$

It follows that the expectation of the random variable $C$ is:

$$\mathrm{E}[C] \approx \sum_{k=1}^{+\infty} k \cdot Prob(C = k) \ .$$

Plugging equation (4) into the expression above and simplifying the resulting formula (we have used Mathematica$^{\mathrm{TM}}$), we obtain:

$$\mathrm{E}[C] \approx \frac{-25 + 69p_1 - 85p_1^2 + 58p_1^3 - 22p_1^4 + 4p_1^5}{p_1(-2 + p_1)(3 - 3p_1 + p_1^2)(2 - 2p_1 + p_1^2)} \ . \tag{5}$$

Note that the value above must be multiplied by four to obtain the number of expected sensors to drop, since each experiment in this scenario consists of dropping four sensors. So, $\mathrm{E}[X_a] = 4 \cdot \mathrm{E}[C]$, where $X_a$ is the random variable that denotes the number of sensors to be dropped in scenario (a) to have every cell covered.

Turning to scenario (b), we observe that it is not necessary always to drop $4k$ sensors (for some $k > 0$), and a finer analysis is required. The problem can be viewed as a coupon collecting task (see, e.g., [11]): there are four coupons (the four cells) that we want to collect, called *valid* coupons, and one which we are not interested in, corresponding to the region outside $R$. Actually, we may assume that the latter is a fifth coupon that we have already collected. The number of attempts that we have to make before collecting all the valid coupons is the sum of the expected values of four geometric variables, $X_1$ through $X_4$, characterized by decreasing probability of success. More precisely, let $p$ the probability of collecting any of the four valid coupons during one attempt. Note that in our case $4p < 1$, since the coupon we are not interested in has nonzero probability of occurrence. For the first attempt we are happy if we get any of the four valid coupons, and this happens with probability $4p$. For the second attempt, we look for one of only three valid coupons, with corresponding total probability $3p$. By iterating the reasoning it is not difficult to see that the expected number of attempts before collecting all the valid coupons under scenario (b) is

$$\begin{aligned} \mathrm{E}[X_b] &= \mathrm{E}[X_1 + X_2 + X_3 + X_4] = \sum_{i=1}^{4} \mathrm{E}[X_i] \\ &= \sum_{i=1}^{4} \frac{1}{ip} = \frac{25}{12p}, \end{aligned} \tag{6}$$

where $p = 1 - \hat{q}_1$.

Figure 22 shows the plot of the expected values $\mathrm{E}[X_a]$ and $\mathrm{E}[X_b]$: for $\frac{\sigma}{4u} < \bar{c} \approx 0.1875$ , dropping sensors at the center of each cell is the best choice (in expectation); for $\frac{\sigma}{4u} > \bar{c}$, the situation is reversed, and dropping sensors at the center of the region is better. We have thus proved the following theorem, which, similarly to the one-dimensional case studied in Section 4.1, shows that the best performing sensor deployment strategy depends on the environmental conditions.

**Theorem 2.** *Assume R, with side l, is divided into four cells, and that we want to deploy the minimum number of sensors such that all the cells are covered, and consider the two deployment strategies* (*a*) *and* (*b*) *defined above. If the dispersion factor* $\sigma$ *is such that* $\frac{\sigma}{l} < \bar{c}$*, then the best performing strategy is to drop sensors at the center of each cell (strategy* (*a*)*); otherwise, the best performing strategy is to drop sensors at the center of R (strategy* (*b*)*).* □

Since $\mathrm{E}[X_a]$ and $\mathrm{E}[X_b]$ are not integers, in general, we also report the rounded values in Table 3, whose entries are computed from the expected values as follows: under scenario (b) the nearest largest integer; under scenario (a) the nearest largest integer divisible by 4. If you consider two consecutive rows of Table 3, say rows $i$ and $i + 1$, then row $i + 1$ gives the number of sensors to be dropped when $\frac{\sigma}{4u}$ is strictly greater than the value in row $i$ and not greater than the value in row $i + 1$.

## 5.2 Simulation study

In this section we report the results of the extensive simulations performed to investigate GSDP.

The simulator that we have designed has the following input parameters:

- the required sensing granularity $s$, with $0 < s \leq 1$;

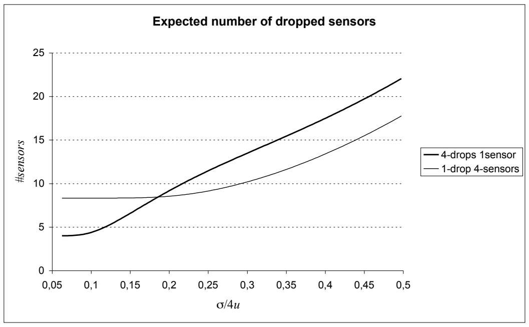

Figure 22: Expected number of dropped sensors to have one sensor in each cell under cases (a) and (b) as a function of $\frac{\sigma}{4u}$.

| $\sigma/4u$ | 4-drops | 1-drop | $\sigma/4u$ | 4-drops | 1-drop |
|---|---|---|---|---|---|
| $< 0.175$ | 8 | 9 | 0.3875 | 20 | 13 |
| 0.24 | 12 | 10 | 0.415 | 20 | 14 |
| 0.2625 | 12 | 10 | 0.4375 | 20 | 15 |
| 0.29 | 16 | 10 | 0.455 | 20 | 15 |
| 0.3275 | 16 | 11 | 0.46 | 24 | 16 |
| 0.36 | 16 | 12 | 0.48 | 24 | 17 |
| 0.3625 | 16 | 12 | 0.4975 | 24 | 18 |

Table 3: Expected number of sensors to be dropped for having one sensor in each cell under (a) and (b).

– the parameter $d$ of the grid drop strategy, with $0 < d \leq 1$;

– the transmitting range $r$;

– the dispersion factor $\sigma$;

– the number of sensors $n_d$ to be dropped at the single drop point. The total number of deployed sensors is then $n = n_d \cdot \left\lfloor \frac{1}{d} \right\rfloor^2$.

In a single experiment, $n_d$ sensors are dropped at each drop point according to the Normal distribution of parameter $\sigma$. Once all the $n$ sensors have been deployed, the communication graph $G$ is generated by connecting with an edge all the pairs of nodes that are within distance $r$. Then, the largest connected component $LCC_G$ of $G$ is calculated. Finally, the fraction of sensing cells covered by at least one node in $LCC_G$ is computed and recorded in the output file.

Note that, since node positions are generated according to a pdf with unbounded support, sensors may lay outside $R$. These "out of bound" nodes

| Parameter | Settings |
|---|---|
| $s$ | 1/8, 1/9, 1/10 |
| $d$ | 1, 1/2, 1/3,...,1/10 |
| $r$ | 0.15, 0.25 |
| $\sigma$ | from 0.005 to 0.05 in steps of 0.0025 |
| | from 0.06 to 0.2 in steps of 0.01 |
| | from 0.225 to 0.5 in steps of 0.025 |
| $n_d$ | from 1 to 1000 |

Table 4: Settings of the input parameters used in our experiments. The values of $s$, $d$, $r$ and $\sigma$ are normalized with respect to the side of the deployment region $R$.

are useless for the purpose of coverage, but can be used as bridges to increase network connectivity.

In our simulations, we have considered about 49000 different settings of the input parameters and, for each setting, we have performed a number of experiments varying from 250 to 1000. The settings of the various parameters used in our simulations are shown in Table 4.

An observation concerning $n_d$ is in order. In order to keep the simulation time reasonable, we have imposed an upper bound of 1000 to the total number $n$ of deployed nodes. Thus, depending on the value of $d$, the maximum value of $n_d$ considered is much smaller than 1000. For instance, when $d = 0.1$ the maximum value of $n_d$ considered in our experiments is 10.

We have performed simulations with two values of the transmitting range $r$ (0.15 and 0.25). Although these values are not sufficient to ensure network connectivity as a consequence of coverage in the worst-case (we recall that we need $r \geq 2r_s \geq 2\sqrt{2}s$ for this), the simulation results have shown that setting $r = 0.15$ is already sufficient to provide connectivity on the average with the DoC requirements used in our experiments. For this reason, in the following we show only the results obtained when $r = 0.15$.

### 5.2.1 Evaluation of drop grid size

In the first set of experiments, we have evaluated the best grid size as the dispersion factor $\sigma$ increases. Given the value of $\sigma$, of $r$, and a certain DoC requirement $DoC(s, q, p)$, we have calculated the value of $d$ corresponding to the best drop strategy (among the ones considered), i.e. the value of $d$ such that $DoC(s, q, p)$ is satisfied and $n$ is minimum. This calculation has been performed by post-processing the huge amount of output data generated by our experiments.

The simulation results for $r = 0.15$, $s = 0.1$, $p = 95$, and different values of $q$ are reported in Figure 23. Figure 24 reports the same results when $s = \frac{1}{8} = 0.125$. In the figures, 75-95 (and the other plots) must be interpreted as follows: in at least 95% of the cases, at least 75% of the cells is covered by at least one sensor in $LCC_G$.

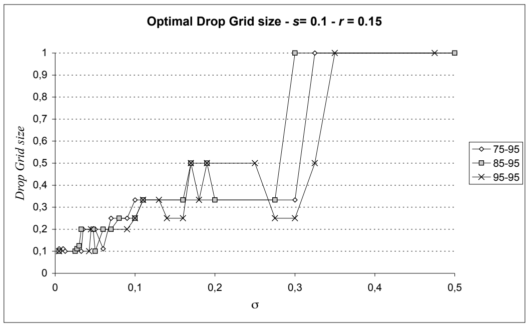

Figure 23: Optimal grid size for increasing values of $\sigma$. The sensing granularity is $s = 0.1$, and the transmitting range is $r = 0.15$.

From figures 23 and 24 it is seen that as the dispersion factor $\sigma$ increases, the size of the best grid strategy increases as well. When $\sigma$ is around 0.3 or above, the best drop strategy is to drop all the sensor at the center of the deployment region $R$, independently of the DoC requirement. Note that in this case it is likely that the uncovered cells reside on the boundary of $R$. On the other hand, when the expected dispersion is very small ($\sigma = 0.03$ and below), the grid strategy of minimum size 0.1 is always best. In between these two values of $\sigma$, we are in an intermediate regime, where we can observe an increasing trend of the drop grid size. This trend is not perfectly linear, due to the considerable effect of the discretization which is inherent in the definition of grid deployment strategy. It is also important to observe that *the effect of the DoC requirement on the best performing grid size is only marginal.* We believe this is due to the fact that which deployment strategy performs best mainly depends on the probability of covering cells in the vicinity of the drop point, which heavily depends on the dispersion factor (recall Section 5.1).

### 5.2.2 Number of deployed nodes

In the second set of experiments, we have evaluated how the total number of deployed sensors varies as a function of the dispersion factor $\sigma$. We have considered four different grid drop strategies ($d = s$, $d = \frac{1}{6} = 0.166$, $d = \frac{1}{3} = 0.333$ and $d = 1$), with various DoC requirements. The transmitting range is set to 0.15.

The results of our experiments for different *DoC* requirements are reported in figures 25–28. The figures also report the number of sensors needed to obtain the same *DoC* requirements under optimal (manual) and random uniform sensor deployment. We make the following observations.

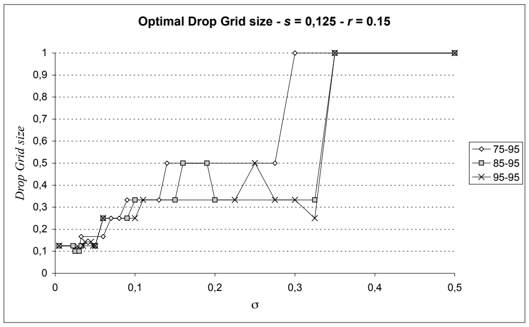

Figure 24: Optimal grid size for increasing values of $\sigma$. The sensing granularity is $s = 0.125$, and the transmitting range is $r = 0.15$.

- *For any of the DoC requirements considered*, the behavior of $n$ as $\sigma$ increases depends on the drop strategy: if the drop grid is very fine ($d = s$), we have a linear increase of $n$ with $\sigma$ (up to the discretization induced on $n$ by the fact that $n_d$ is an integer). For relatively coarser grids, $n$ decreases with $\sigma$ quite sharply initially; then, it increases with a relatively modest slope. When the dispersion factor is large enough ($\sigma$ around 0.35 and above), $n$ increases with $\sigma$ independently of the grid size. This fact is quite intuitive: as the dispersion factor increases, the node deployment becomes more and more difficult to control, and the total number of sensors to be deployed to meet the DoC constraint increases.

- *For any of the DoC requirements considered*, we can distinguish three regimes in the plots: for small values of $\sigma$ ($\sigma$ around 0.1 or below), dropping the sensors using the finest grid ($d = s$) is a good choice, that often minimizes $n$. For large values of $\sigma$ ($\sigma$ around 0.3 and above), the best option is always to deploy sensors using a unique drop point located at the center of $R$. For intermediate values of $\sigma$, setting $d = 0.333$ seems to be the best choice.

- the ‘finer’ grid deployment strategy ($d = 0.1$) always resulted, if not the best performing, very close to the best performing strategy. This finding, which is in accordance with the results for the one-dimensional case, suggests that dropping sensors at the center of each cells might be the best choice in a practical setting.

- The DoC requirement has a dramatic effect on $n$. For instance, when $\sigma = 0.1$, less than 200 nodes are sufficient to satisfy $DoC(0.1, 75, 95)$, while more than 400 nodes are necessary to satisfy $DoC(0.1, 95, 95)$. In words, increasing the required percentage of covered cells by 20% results in a more than 100% increase in the number of deployed sensors. This means, under certain environmental conditions, requiring 100% coverage of the monitored region incurs a huge infrastructure cost. Changing the constraint on $s$ has a less significant impact on $n$: when $\sigma = 0.1$ and the requirement is $DoC(0.125, 75, 95)$, the number of nodes to be deployed is around 125. Thus, a 25% increase in $s$ results in a decrease in the order of 37% of the number of deployed sensors (we recall that a larger value of $s$ results in a less stringent DoC requirement). Note that the situation is very different when $\sigma$ is very small ($\sigma$ below 0.025); in this case, up to 200 nodes are sufficient to satisfy both $DoC(0.1, 75, 95)$ and $DoC(0.1, 95, 95)$. This is due to the fact that with these values of $\sigma$ node deployment can be accurately controlled, and dropping 1 or 2 sensors at the center of every sensing cell is sufficient to meet basically all the DoC requirements. In these situations, requiring 100% coverage of the monitored region is a realistic goal, and techniques similar to those presented in [18, 20, 22] can be used to increase network lifetime.

– when the dispersion factor is low ($\sigma < 0.05$), partially controlled sensor deployment is more effective than random uniform deployment in achieving the desired $DoC$ requirement, especially when the coverage requirement is quite stringent. For instance, when $DoC = (0.1, 0.95, 0.95)$, at least 357 sensors are needed in cased of random uniform deployment, while $\leq 300$ are needed with partially controlled deployment when $\sigma < 0.05$. This should be compared with 95 sensors which are needed in case of optimal sensor deployment. On the other hand, partially controlled deployment is less effective than random uniform deployment when the dispersion factor is high. This is also due to the fact that, under random uniform deployment, we are actually *guaranteed* that all the deployed sensors land into $R$, which is hardly the case in case of partially controlled deployment with high dispersion factor. This observation discloses an interesting insight, i.e., that, contrary to what typically assumed in the literature, random uniform deployment of nodes might not be a sort of 'worst-case scenario' for studying coverage problems in WSNs.

Before concluding this section, we present sample sensor deployments that help us clarifying why the apparently trivial strategy of dropping all the sensors at the center of $R$ is optimal for large values of $\sigma$. We consider the two extreme drop strategies: having a drop point in the center of every sensing cell ($d = s$), and having a unique drop point at the center of $R$. Figure 29-a shows sample deployments obtained with the two dropping strategies when $\sigma = 0.04$ with $n = 200$ nodes. Figure 29-b shows similar samples when $\sigma = 0.4$ and $n = 300$. In case of small $\sigma$, the finer drop strategy results in a far better coverage: 94% of the cells are covered, as compared to only 10% covered cells when $d = 1$. The situation is reversed when $\sigma = 0.4$: in this case, the cell coverage is 70% with $d = s$ and 82% with $d = 1$. The better coverage of the coarser drop strategy is due to the fact that when all the sensor are dropped at the center of deployment

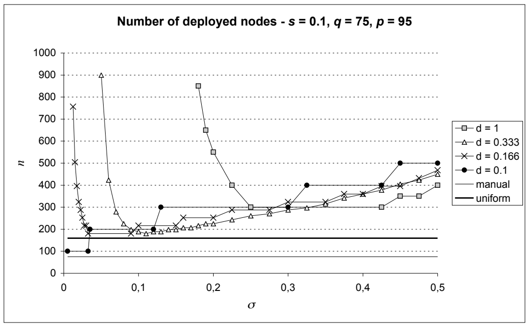

Figure 25: Total number of deployed sensors with increasing values of $\sigma$ and different grid strategies. The DoC requirement to be met is $DoC(0.1, 75, 95)$.

region, less sensors are expected to fall outside the boundaries of $R$: the number of "out of bound" sensors is 56.7% when $d = s$, and only 35% when $d = 1$.

# 6 Final remarks

In this paper we have investigated in detail a wireless sensor dropping problem that can arise when the WSN is used for monitoring vast geographical regions. We have considered a target spatial DoC requirement, and studied both theoretically and experimentally the relation between the environmental conditions that influence node dispersal (the dispersion factor(s)) and the best performing deployment strategy. In particular, we have shown that, in many practical cases, *the best performing deployment strategy depends only on the environmental conditions, and not on the particular DoC requirement*. This observation is important, since it eases the task of identifying the best performing deployment strategy.

We believe that one of the main contributions of this paper is the definition of a simple yet powerful methodology to tackle the problem of finding a 'good' WSN deployment strategy: 1) define a quantity that can be used to measure QoS (in our case, sensing coverage) 2) define a pdf that models the node dispersal at the single drop point (the Normal distribution in this paper) as a function of the environmental conditions (modeled by the dispersion factor(s) in our study) 3) define a reasonable set of drop strategies (such as equally spaced drop points in the one-dimensional scenario, or grid drop strategies in the two-dimensional case) 4) once the above parameters are set, the best deployment strategy can be identified.

As the example of barrier coverage considered in Section 4 has shown, the methodology described above can be used to study node deployment strategies

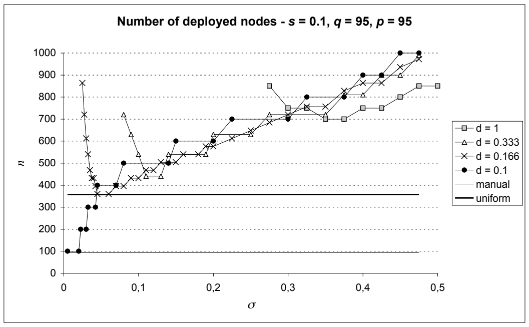

Figure 26: Total number of deployed sensors with increasing values of $\sigma$ and different grid strategies. The DoC requirement to be met is $DoC(0.1, 95, 95)$.

in other application scenarios, such as target detection. For instance, it is clear that sensing granularity and exposure (see [17]) are related measures: the finer the granularity, the easier it is for the WSN to detect a target moving within the monitored region. An in-depth investigation of the relation between these two important QoS measures is the subject of ongoing studies.

Another way to extend the work presented in this paper is to analyze sensor deployment problems in which the QoS requirement accounts also for *temporal* accuracy, such as the latency with which events are reported to the base station. We are currently working on this topic, and on the related topic of deploying a network with a minimal guarantee on the operational lifetime. We believe that a viable approach to tackle these more complex problems is to combine our methodology with techniques used in the analysis of topology control, clustering and energy-conserving protocols.

Finally, it is interesting to study variations of the deployment strategies considered in this paper, for instance strategies in which drop points are subject to a perturbation error, or in which the sensing regions are not uniform.

# References

[1] D.M. Blough, P. Santi, "Investigating Upper Bounds on Network Lifetime Extension for Cell-Based Energy Conservation Techniques in Stationary Ad Hoc Networks", *Proc. ACM Mobicom 02*, pp. 183–192, 2002.

[2] K. Chakrabarty, S.S. Iyengar, H. Qi, E. Cho, "Grid Coverage for Surveillance and Target Location in Distributed Sensor Networks", *IEEE Trans. on Computers*, Vol. 51, n. 12, pp. 1448–1453.

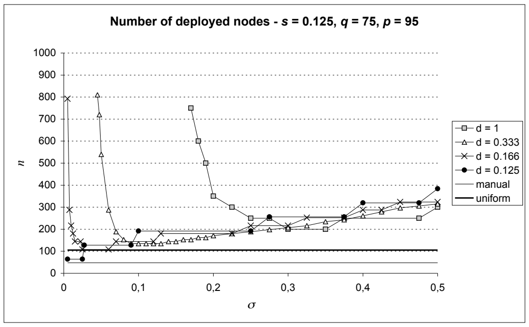

Figure 27: Total number of deployed sensors with increasing values of $\sigma$ and different grid strategies. The DoC requirement to be met is $DoC(0.125, 75, 95)$.

[3] T. Clouqueur, V. Phipatanasuphorn, P. Ramanathan, K.K. Saluja, "Sensor Deployment Strategy for Target Detection", *Proc. ACM Workshop on Sensor Networks and Applications (WSNA)*, pp. 42–48, 2002.

[4] A. Howard, M.J. Mataric, G.S. Sukhatme, "An Incremental Self-Deployment Algorithm for Mobile Sensor Networks", *Autonomous Robots*, Vol. 13, n. 2, pp. 113–126, 2002.

[5] A. Howard, M.J. Mataric, G.S. Sukhatme, "Mobile Sensor Network Deployment using Potential Fields: A Distributed, Scalable Solution to the Area Coverage Problem", *Proc. Int. Conference on Distributed Autonomous Robotic Systems (DARS)*, pp. 299–308, 2002.

[6] V.F. Kolchin, B.A. Sevast'yanov, V.P. Chistyakov, *Random Allocations*, V.H. Winston and Sons, Washington D.C., 1978.

[7] S. Kumar, T. Lai, A. Arora, "Barrier Coverage with Wireless Sensors", *Proc. ACM Mobicom 2005*, pp. 284-298, 2005.

[8] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, J. Anderson, "Wireless Sensor Networks for Habitat Monitoring", *Proc. ACM WSNA 02*, pp. 88–97, 2002.

[9] S. Meguerdichian, F. Koushanfar, G. Qu, M. Potkonjak, "Exposure in Wireless Ad-Hoc Sensor Networks", *Proc. ACM Mobicom*, pp. 139–150, 2001.

[10] S. Meguerdichian, S. Slijepcevic, V. Karayan, M. Potkonjak, "Localized Algorithms in Wireless Ad Hoc Networks: Location Discovery and Sensor Exposure ", *Proc. ACM MobiHoc*, pp. 106–116, 2001.

[11] R. Motwani and P. Raghavan, *Randomized Algorithms*, Cambridge University Press (1995).

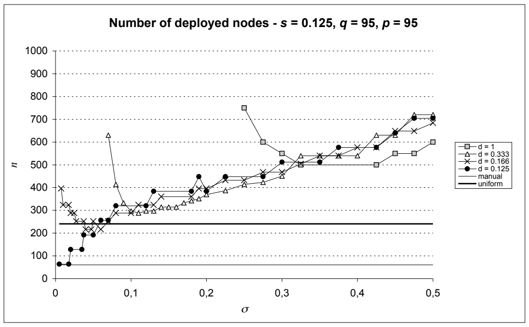

Figure 28: Total number of deployed sensors with increasing values of $\sigma$ and different grid strategies. The DoC requirement to be met is $DoC(0.125, 95, 95)$.

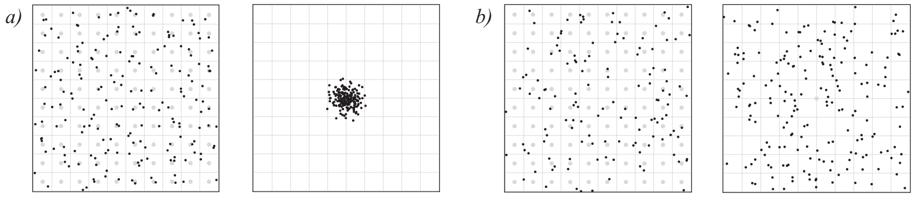

Figure 29: Sample sensor deployments when $\sigma = 0.04$ ($a$)) with different dropping strategies: $d = s$ ($a$)-left) and $d = 1$ ($a$)-right). Black circles represent sensors, and shaded circles represent the drop points. Only sensors that lay within $R$ are shown. Deployments when $\sigma = 0.4$ are shown in $b$): $d = s$ ($b$)-left) and $d = 1$ ($b$)-right).

[12] G.J. Pottie, W.J. Kaiser, "Wireless Integrated Network Sensors", *Communications of the ACM*, Vol. 43, n. 5, pp. 51–58, 2000.

[13] P. Santi, D.M. Blough, "The Critical Transmitting Range for Connectivity in Sparse Wireless Ad Hoc Networks", *IEEE Transactions on Mobile Computing*, vol. 2, no. 1, pp. 25–39, January-March 2003.

[14] L. Schwiebert, S. Gupta, J. Weinmann, "Research Challenges in Wireless Networks of Biomedical Sensors", *Proc. ACM Mobicom*, pp. 151–165, 2001.

[15] D.C. Steere, A. Baptista, D. McNamee, C. Pu, J. Walpole, "Research Challenges in Environmental Observation and Forecasting Systems", *Proc. ACM Mobicom*, pp. 292–299, 2000.

[16] S. Tilak, N.B. Abu-Ghazaleh, W. Heinzelman, "Infrastructure Tradeoffs for Sensor Networks", *Proc. ACM Workshop on Sensor Networks and Applications (WSNA)*, pp. 49–58, 2002.

[17] G. Veltri, Q. Haung, G. Qu, M. Potkonjak,“Minimal and Maximal Exposure Path Algorithms for Wireless Embedded Sensor Networks”, *Proc. ACM SenSys*, pp. 40–50, 2003.

[18] X. Wang, G. Xing, Y. Zhang, C. Lu, R. Pless, C. Gill, “Integrated Coverage and Connectivity Configuration in Wireless Sensor Networks”, *Proc. ACM SenSys*, pp. 28–39, 2003.

[19] A.F. Winfield, “Distributed Sensing and Data Collection Via Broken Ad Hoc Wireless Connected Networks of Mobile Robots”, *Distributed Autonomous Robotics Systems*, Vol. 4, pp. 273–282, 2000.

[20] F. Ye, G. Zhong, S. Lu, L. Zhang, “Peas: a Robust Energy Conserving Protocol for Long-Lived Sensor Networks”, *in Proc. IEEE ICDCS*, pp. 28–37, 2003.

[21] O. Younis, S. Fahmy, “Distributed Clustering in Ad-hoc Sensor Networks: A Hybrid, Energy-Efficient Approach”, *Proc. IEEE Infocom 04*, 2004.

[22] H. Zhang, J. Hou, “Maintaining Sensing Coverage and Connectivity in Large Sensor Networks”, *NSF International Workshop on Theoretical and Algorithmic Aspects of Sensor, Ad Hoc Wireless, and Peer-to-Peer Networks*, 2004.

[23] Y. Zou, K. Chakrabarty, “Sensor Deployment and Target Localization Based on Virtual Forces”, *Proc. IEEE Infocom*, 2003.