Maximum WSN coverage in environments of heterogeneous path loss

Article *in* International Journal of Sensor Networks · January 2014

DOI: 10.1504/IJSNET.2014.066788

CITATIONS
10

READS
253

3 authors:

S.Hossein Mortazavi
University of Toronto
19 PUBLICATIONS **217** CITATIONS
SEE PROFILE

Mohammad Salehe
Sharif University of Technology
10 PUBLICATIONS **146** CITATIONS
SEE PROFILE

Mike H. MacGregor
University of Alberta
126 PUBLICATIONS **1,366** CITATIONS
SEE PROFILE

Maximum WSN Coverage in Environments of Heterogeneous Path Loss

S. Hossein Mortazavi
Computing Science Department
University of Alberta
Email: mortazavi@ualberta.ca

Mohammad Salehe
Computer Engineering Department
Sharif University of Technology
Email: salehe@ce.sharif.edu

M.H. MacGregor
Computing Science Department
University of Alberta
Email: mike.macgregor@ualberta.ca

***Abstract*—We study wireless sensor network (WSN) node placement in an environment where RF signal losses vary with position. This reflects real-world outdoor environments where vegetation and topography cause nonuniform path loss. Many techniques that solve for a variety of objective functions subject to various constraints have previously been proposed for node placement. However, many of these methods make simplifying assumptions such as all nodes having the same transmission range. Our goal is to take the insights and approaches of this previous work and extend it to real-world environments. The present work assumes we have a map that quantifies the path loss behaviour of the real environment. Based on this map, and a path loss model that accounts for spatial variations in the path loss exponent, we propose a node placement algorithm for two-tiered WSNs that maximizes the area covered by a specified number of relay nodes and sensor nodes.**

KEYWORDS

node placement; coverage; heterogeneous path loss; relay node; wireless sensor network.

I. INTRODUCTION

Over the past decade, Wireless Sensor Networks (WSNs) have been used in a wide range of application areas, including agriculture, transportation, home automation, search and rescue, industrial and environmental monitoring, military applications and medical care [2], [47], [52]. One category of applications is environmental monitoring. WSNs have been used in detecting forest fires, monitoring endangered habitats, watching over volcanoes, detecting soil erosion, etc. [7], [33], [42]. However designing efficient networks for outdoor environments is still a major challenge. Factors such as limited or intermittent connectivity, faulty connections, limited lifetime and the cost of the nodes must be confronted. In particular, in static networks where the location of the nodes is not changed frequently, node placement is a key concern and can affect many properties of the network, as well as the quality of the data sensed about the environment.

In difficult environments such as tropical dry forests or Arctic glaciers, it can be expensive and time-consuming to reach and access the area to be monitored. In these types of surroundings, it is impractical to consider adjusting the network to provide proper sensing or RF coverage *after* placing it on the ground [22]. Planning techniques are required that can predict ahead of time where best to locate sensor nodes (SNs), relay nodes (RNs) and other equipment. Otherwise, spatial constraints on node placement that must be respected to guarantee data integrity could easily be violated. Also, physical impact on the plants or animals being monitored may reduce the value of the data. Lastly, access restrictions or weather may limit the duration or timing of such work. Our goal is to contribute techniques that can make use of historically known and predictable environmental data that can be sensed remotely via satellite, aircraft or UAVs. These data can be used to characterize the RF environment of the network (e.g. predicting differences in RF signal propagation in different areas), as well as dictating how densely sensors must be placed to reliably gather data on the environmental variables of interest. Design techniques based on such data would be able to overcome the drawbacks of trying to plan and adjust networks on site, in harsh or delicate environments, and under the multiple pressures of time, cost and human frailty.

In this work, we assume the use of a two-tiered network consisting of SNs forwarding data to a base station through RNs. We further assume single-hop communication from each SN to a nearby RN, rather than multihop communication amongst SNs. The commercial hardware we are currently placing in the field operates in this manner, so this type of network is our first concern. Future work will address design algorithms for networks based on multihop forwarding.

Once the environmental variables of interest have been sensed by the SNs, these data need to be collected, aggregated and transmitted via a reliable route from each SN through the RNs to one or more base stations. In single-tiered networks, the SNs participate in the process of forwarding data to the base station. In multi-tiered networks additional nodes called relay nodes (RNs) are used [22], and SNs may or may not participate in forwarding the data. The RNs are more expensive than SNs because they have higher maximum transmission power than SNs. However, their use can reduce the frequency with which batteries need to be replaced in the SNs, thus significantly increasing network lifetime.

Although optimal node placement has been proven NP-hard in most conditions [14], [54], various optimal and sub-optimal techniques have been proposed for minimizing the number of nodes needed to create a network while respecting constraints such as connectivity [3], network lifetime [49], [50], reliable data transfer [5] and coverage [46]. These methods make various simplifying assumptions such as homogeneous

transmission range of nodes. They also often concentrate on optimizing either the number of SNs or the number of RNs separately [8], [11].

We propose a node placement algorithm for two-tiered WSNs that maximizes the area covered by a given number of RNs and SNs. We assume the area to be covered has a path loss characteristic that varies spatially, and that this variation has been mapped and expressed in terms of values of a path loss exponent. We specifically concentrate on static networks. Our main contributions are:

- **Considering heterogeneous signal path loss**: In this work we consider heterogeneous communications among nodes. This means that a signal that originates from a source may experience different amounts of loss when it reaches points at different directions from the source, but at the same distance r from the source. The amount of loss depends on the areas the signal propagates through.
- **Maximizing the area covered by a given number of RNs and SNs in an environment of heterogeneous path loss:** We solve the problem of maximizing the area that can be covered by a specific number of SNs and RNs. Rather than first placing the SNs, our Max-Cover 1-Connected algorithm first places RNs in locations that maximize the area in which SNs can be placed. We then modify a known SN placement algorithm to fully utilize a given number of SNs by placing them within that area.

The paper is organized as follows: in Section II we review related work, including the problem model, details of signal propagation, and the hardware we simulate. We describe the Max-Cover 1-Connected algorithm in Section III, and in Section IV we present our results and comparisons in detail. Conclusions and future work follow in Section V.

II. Related Work

Optimal node placement has been studied from the very introduction of WSNs [54]. Many methods have been proposed to address challenges such as network connectivity [3], [4], [10], [29], [55], network lifetime [48], [50], [51], network coverage [27], [45], [46], and fault tolerance [5], [57], [58], etc. mainly in 2D but also in 3D space [34], [35]. These papers often frame secondary challenges as constraints on the problem. While some of these algorithms concentrate on placing SNs [1], [11], [24], others focus on RN placement [8], [26]. In this section, we review some previous algorithms that optimize the number of SNs and RNs while providing specified network coverage or network connectivity.

A. Sensor node placement

We first categorize static SN placement algorithms based on the problem they solve. The methods used include linear programming, greedy algorithms, divide and conquer, simulated annealing, tabu search, etc. The problem that has attracted the most attention in the literature is known as the area coverage problem [15]. Here, the challenge is either maximizing the sensing area covered by a specific number of nodes, or minimizing the number of nodes needed to cover a specific amount or proportion of an area.

One study is that by Dhillon and Chakrabarty [11]. They present a probabilistic optimization framework for placing SNs. Their framework uses a probabilistic detection model for the SNs, and they consider noise and distance as parameters that can affect the information sensed by the node. Their Max-Min-Cov algorithm deploys SNs only at predefined grid points. This greedy, iterative algorithm considers a coverage miss probability for each cell in the grid. The miss probability of a cell is the probability that a cell is not covered by any sensor node. In each iteration an SN is deployed at the point where it has the maximum impact on minimizing the miss probabilities of the overall grid. After each SN is placed, the miss probabilities of the cells are updated. Dhillon and Chakrabarty also discuss similarities between this problem and the art gallery [30] problem.

A triangular-grid SN placement architecture was presented by Pompili et al. [34] for planning underwater WSNs in both two and three dimensions. The main objective of this study was to find the minimum number of SNs needed to achieve optimal sensing and communication coverage; this is done using tiling patterns. They propose placing the SNs on a triangular grid and prove that if the distance between neighbouring nodes is $\sqrt{3}$ times the sensing range, then the given area would be fully covered using their method.

Lin et al. [24] consider the problem of sensor placement for locating targets. They model the area to be sensed as a grid and propose a simulated annealing minimax optimization that ensures full coverage. They first place SNs on all grid points and gradually remove unnecessary nodes until a cost constraint is met. In each iteration an attempt is made to remove one node. If this would violate coverage, a node is selected and moved to a random position. The algorithm terminates once coverage and discrimination requirements are achieved. They compare their solution with the optimal solution in smaller cases and claim that although their algorithm may not always find the best possible result, the results are near-optimal.

Zhang et al. [56] address the problem of node placement in situations where the required detection probability thresholds of various locations are different. Similar to the work by Dhillon and Chakrabarty, they assume probabilistic sensing by the SNs. They propose an iterative heuristic algorithm called DIFF-DEPLOY. The main drawback of their method is its $O(\frac{4}{3}n^6)$ computational complexity.

Aitsaadi et al. [1] propose a tabu search to achieve full coverage. They also consider a differentiated detection probability threshold for different areas and use a probabilistic event detection model for sensor nodes. Geographical characteristics of the monitored events are also considered. They claim that the number of SNs needed to fully cover a field is less than for previous methods, such as Max-Min-Cov and DIFF-DEPLOY.

The problem of evaluating coverage in a network was scrutinized by Meguerdichian et al. [27]. In their study of wireless ad-hoc sensor networks, Voronoi diagrams and Delaunay triangulation are used to find spaces between the nodes

that are not covered. They compute maximal breach paths and maximal support paths. The maximal breach path is a path where the shortest distance to any sensor is as large as possible. The maximal support path is a path whose greatest distance from the closest sensors is minimized. The polynomial time algorithm they present is centralized and is used to determine worst-case (breach) and best-case (support) coverage.

While connectivity is very important in WSNs, the previous studies only discuss the coverage constraint - connectivity among the SNs is not considered. These studies have been extended by including the connectivity constraint. This means that while coverage remains the main objective, the single or k-connectivity constraint among the SNs is satisfied as a side problem. To satisfy the k-connectivity constraint, every SN must be connected to k other SNs in the network.

To achieve both connectivity and coverage, Wang et al. [49] divide a sensing area into subareas. In each area, they deploy SNs row-by-row such that each row guarantees continuous coverage and connectivity, and such that adjacent rows ensure continuous coverage. Their solution allows arbitrary polygons as the sensing area, with possible existence of obstacles.

The relationship between connectivity and coverage in single-tier networks has been studied in detail by Zhang and Hou [55]. They use computational geometry to prove that with a radio transmission range of at least twice the sensing range, connectivity is implied by complete coverage of a convex area in the network. In their work they maintain coverage and connectivity by keeping a minimum number of SNs operational. They also study the problem of choosing the optimal number of working SNs from a dense network to obtain full coverage.

Biagioni and Sasaki [5] discuss various SN placement topologies. They aim to achieve coverage and also preserve connectivity in case of possible failures or battery depletion of a number of SNs. They study various deployment topologies such as circular, hexagonal, star-in-square, triangle and hexagonal grids. They conclude that the deployment depends on the sampling distance and the communication radius of the SNs, which means that each deployment should be customized to its specific location.

Other work includes the analytical study by Kumar et al. [20]. Their method covers each point of an area with at least k nodes, and provides analytical bounds on the number of nodes needed to do so. This property is mainly required in intrusion detection and security applications. They also develop the Randomized Independent Scheduling (RIS) algorithm, which maximizes network lifetime by using a probabilistic model to control which SNs are functional and which are sleeping.

B. Relay node placement

In two-tiered networks, the connectivity and fault tolerance of the network depends on the placement of the RNs. The main objective is to minimize the number of RNs needed to connect the SNs to the base station. This problem has been studied in wireless area networks [38], IEEE 802.16j [23] and WiMAX [53]. We assume k is the required connectivity between the RNs, and r and R are the respective radio ranges of the SNs and RNs. It has been proven that for $k = 1$ and $r = R$ the RN placement problem is NP-hard [25]. While some previous studies [18], [57] have a secondary objective of achieving k-connected networks where $k > 1$, we require only $k = 1$.

One motivation for the use of RNs is to increase network lifetime. Based on the application, different studies use different definitions for lifetime. It could be the time when the first node dies, the time by which a specific number of nodes have died, or the time before loss of coverage occurs [9]. Pan et al. [31] maximize the topological lifetime of a multi-tiered WSN by deploying base stations and hierarchical clusters with application nodes at specific locations functioning as RNs. By maximizing topological lifetime they mean maximizing the time from network initialization to the time when the WSN cannot maintain enough application nodes alive to continue its given mission. Additionally, they provide upper and lower bounds for the maximal topological lifetime.

Tang et al. [43] present a method for minimizing the number of RNs in networks where the SNs are distributed uniformly. The SNs are either one-connected or two-connected to an RN, and the RNs are either one-connected or two-connected to each other. Using r and R to denote the communication range of the SNs and RNs respectively, they study the case where $R \geq 4r$. Their solution is a 4.5-approximation algorithm for both one-connected and two-connected networks. That is, their solution is guaranteed to be within a factor of 4.5 of the optimal. They concentrate on guaranteeing connectivity and ensuring reliability in case of node failure, rather than increasing the total network lifetime.

Lloyd and Xue [26] present an improved algorithm that only has a constraint of $R \geq r$. Based on whether the SNs forward data or not, they propose two algorithms for single-tiered and multi-tiered networks, respectively. For the single-tiered case, their solution is based on finding a minimum spanning tree; a polynomial-time 7-approximation algorithm is proposed. For the two-tiered case where the RNs should be strongly connected, the algorithm finds a Steiner tree with a minimum number of Steiner points. The minimum geometric disk cover algorithm is used to connect the SNs to the RNs. The objective of the minimum geometric disk cover algorithm is to find the minimum number of unit disks whose union covers a given set of input points. The algorithm provided by Lloyd and Xue for the two-tiered case is a polynomial time $(5+\epsilon)$-approximation algorithm. Furthermore the NP-hardness of the second problem is proved.

Inspired by Lloyd and Xue's work, Zhang et al. [57] develop a polynomial 14-approximation algorithm for fault-tolerant networks that aims to deploy the minimum number of RNs while providing two-connectivity among RNs. They propose a $(20+\epsilon)$-approximation algorithm that includes placement of base stations.

Srinivas et al. [39] propose an improved two-stage approximation algorithm with the assumption that $R \geq 2r$. They

formulate this as the connected disk cover problem for mobile backbone networks. In their study, every regular node should be connected to a backbone node. In the first stage, a strip cover algorithm is used to associate SNs to RNs. In the second phase, they connect the RNs by creating a Steiner tree and using a minimum number of Steiner points.

In [10], Cheng et al. tackle the problem of relay placement. Their objective is to deploy RNs to maintain connectivity between every pair of SNs, under the assumption that the SNs can also communicate with each other. They assume the RNs have the same communication range as the SNs, i.e. $R = r$. They formulate the problem as a Steiner tree with a minimum number of Steiner points and bounded edge length (SMT-MSPBEL). This problem was previously proved NP-hard by Lin and Xue [25].

Chen and Cui [8] propose a polynomial time $(5 + \epsilon)$-approximation algorithm for the RN placement problem in WSNs with a base station. To find the positions of the RNs, they first turn the problem into a 1-geometric disk cover problem [17] where each SN needs to be covered by one RN. Then they design an algorithm similar to that used by Cheng et al. for solving the SMT-MSPBEL problem. The objective is to efficiently connect the RNs deployed during the previous stage to the base station.

Misra et al. [29] study two problems in the category of constrained RN placement in single-tiered networks where RNs can only be placed at a set of candidate locations. They propose polynomial-time approximation algorithms for solving the problem of connecting each SN to the base station using a bidirectional path through the RNs. They also consider the problem of connecting each SN through the RNs to at least two base stations by creating a Steiner tree with minimum Steiner points (STP-MSP).

The aforementioned studies assume homogeneous connections between nodes, and identical node transmission ranges. For more realistic networks, Han et al. [16] address the problem of deploying RNs to provide fault-tolerance with higher network connectivity in heterogeneous wireless sensor networks. They assume a transmission range of $[Tmin, Tmax]$ for SNs and a range of T_{relay} for RNs. They use both one-way and two-way links in their model. Their main result is an $O(k^3)$-approximation algorithm under the constraints that $R \geq r$ and $k \geq 2$.

Many of the studies discussed above are analytical, and the methods proposed have not been implemented or tested, either in real networks or by simulation. This means that aspects such as the communication quality among nodes in different scenarios, or the network lifetime that results from the use of particular techniques have not been studied. Also, all of these studies assume that all nodes of the same type have the same transmission range. Assuming heterogeneous radio ranges for SNs and RNs makes the problem significantly more difficult but also more realistic. In our framework, we allow for transmission range to vary as a result of varying propagation conditions in the area being sensed. Furthermore we aim to improve the efficiency of the network by placing both SNs and RNs, rather than considering just one of these problems. We also determine the communication quality between nodes.

Fig. 1. Sample map

C. Problem model

None of the studies discussed in the previous section consider the problem of SN or RN placement in areas with non-uniform communications for both types of nodes. However, several studies that actually measured the RF path loss in environments such as forests [6], [28], [36] have found highly variable and heterogeneous conditions. Caldeirinha et al. find that the receiving signal is affected by trees and vegetation that add attenuation and also scatter the signal in various ways. Dias at el. studied the path loss of links in a Brazilian rain forest [[12]. These measurement studies motivate us to consider spatially-varying RF conditions in the area of interest.

We use a grid Γ to subdivide the geographical area of interest, and then we attribute a value of a path loss exponent α to each cell in Γ. In the simplest case, Γ is a uniform rectangular grid. However any arbitrary tessellation of the geographical area of interest can be used by our algorithm. A sample map of a boreal forest area in Alberta [13] is depicted in Figure 1, along with a simple rectangular grid. Using maps derived from remote sensing, the number, type and density of trees and other vegetation in each cell can be used to select the value of α for each cell. More accurate values for α result in better modelling of the area and a more realistic version of the map. For our current experiments, we use synthetically-generated values and maps; the use of real data is proposed as future work. Similar models that divide an area based on environmental characteristics have previously been proposed in [37] where a region in Australia with different vegetation characteristics was divided into 2000 zones.

In general, increasing the number of the cells increases

the accuracy of the algorithm we propose in the following. However splitting the map into smaller cells results in more computation and requires more detailed knowledge of the area.

D. Signal propagation model

Spatial variation of the path loss exponent has previously been used in empirical models such as Lee et. al's [21] path loss prediction method for flat terrain, as well as in Grimlund and Gudmundson's empirical street corner path loss model [40]. Inspired by these models we have extended the basic model for free space path loss [32] to account for the attenuation of the transmitted signal as it propagates through cells with varying values of α.

The basic equation for free space path loss is:

$$P_r(d) = P_t \times G_1 G_2 (\frac{\lambda}{4\pi d})^2 \quad (1)$$

where P_t is the transmitted power in mW, P_r is the received signal strength in mW, d is the distance from the transmitter in meters, G_1 and G_2 are the gain of the transmit and receive antennas and λ is the wavelength in meters. We define the constant $K = G_1 G_2 (\lambda/4\pi)^2$, so that:

$$P_r(d) = K P_t / d^2 \quad (2)$$

Following this approach, we calculate the path loss for a signal crossing portions of two cells as:

$$P_r(d) = K P_t \times \frac{1}{{d_1}^{\alpha_1} {d_2}^{\alpha_2}} \quad (3)$$

where d_1 is the distance the signal travels within the first cell and d_2 is the distance the signal travels in the second cell. α_1 and α_2 are the values of the path loss exponent in the first and second cell, respectively. K appears only once, rather than as K^2, because it expresses the effects of transmit and receive antenna gains, and transmit frequency. The signal is not being received and retransmitted when it crosses between cells, nor is the frequency changing, so there is no motivation for repeating the gain or frequency terms in the product. We further note that in the generalization from Eq. 2 to Eq. 3 we have changed the fundamental nature of the path loss model from a mechanistic explanation of path loss in free space, to an empirical model of path loss in a general environment. In particular, once the value of α is changed to any value other than the integer 2, the physical units on the right-hand side are different from those on the left. This highlights the empirical nature of this and similar models [21], [40].

To calculate the signal loss along a path that travels through k cells, we further generalize Eq. 3 to the following:

$$P_r(d) = K P_t \times \Pi_{i=1}^{k} 1/{d_i}^{\alpha_i} \quad (4)$$

Here $\sum d_i$ is the distance from the transmitter to the receiver. The example in Figure 2 shows node A transmitting a signal with a power of 100 mW at a frequency of 2.4 GHz. The power of the received signal at B is:

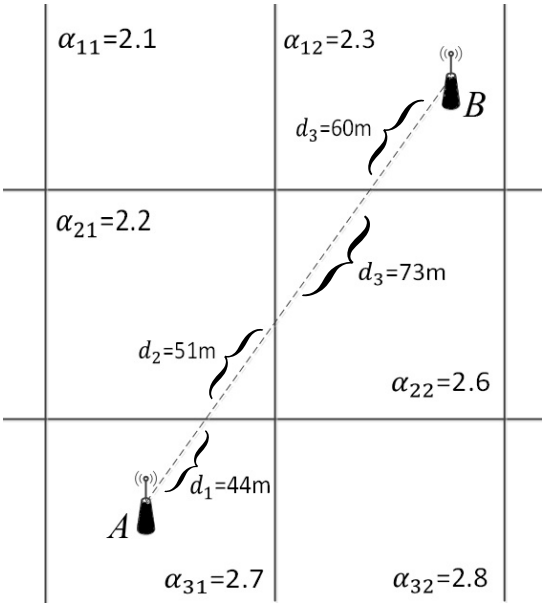

Fig. 2. Path loss example

$$\begin{aligned} &100 \times (1/44^{2.7}) \\ &\times (1/51^{2.2}) \times (1/73^{2.6}) \\ &\times (1/60^{2.3}) = 7.44 \times 10^{-16} \\ &\simeq -151 dBm. \end{aligned}$$

Also, we rely on the following definitions:

- RN to RN communication: If we consider a receiving sensitivity threshold P_{tr} for RN B, then RN A can connect to RN B if the power of the signal received from A at B is at least P_{tr}.
- SN to RN communication: SN S is connected to RN A if the signal originated at S has a minimum power of P_{ts} when it reaches A.

E. Simulated hardware

For the communications between nodes, we use the transmit power and receive sensitivity of the Digi Xbee Pro R1 802.15.4 radios. These transmit at a frequency of 2.4 GHz, which is 0.125 m wavelength, and have a receive sensitivity of -70 dBm. The SNs have a transmit power of 10 dBm, while the RNs have a transmit power of 20 dBm.

III. Max-Cover 1-Connected Algorithm

In the previous section we noted that previous node placement algorithms do not account for heterogeneous RF losses. Here, we solve the problem of maximizing the area covered by a given number m of SNs and n RNs in a heterogeneous RF environment. We propose a three-stage solution for placing both SNs and RNs. Our algorithm aims to fully utilize each node so that sensing coverage is maximized while connectivity

is maintained. We call our proposed algorithm max-cover 1-connected for reasons that will become clear in the following. The algorithm is composed of three stages:

- greedy RN placement: First, we use iterative greedy placement of RNs. In each round, an RN is placed where it maximizes the number of *candidate* locations for SNs and RNs in the next round. Each newly placed RN must be connected to those previously deployed. The candidate locations we maximize for SN placement are those where an SN could communicate with an RN, and the candidate RN locations are those where an RN could communicate with another RN, or with a base station.
- improve RN placement with simulated annealing: Once all the RNs have been initially deployed, we try to improve the placement via simulated annealing. As the objective is to maximize the area sensed by the given number of SNs, any effective algorithm for adjusting a given RN deployment must maximize the number of candidate locations for SN deployment. To be effective, the algorithm must also preserve connectivity between the RNs and the base stations.
- SN placement: Last, we deploy SNs in locations made feasible by the RN placement. We utilize a modified version of the method presented by Pompili et al. [34]. The previous two steps guarantee that SNs placed in any of the candidate locations will have the ability to establish a connection to the base station through some sequence of RNs.

A. Greedy RN placement

Unlike other node placement algorithms, our max-cover 1-connected method deploys both RNs and SNs, and does so to create a two-tiered network deployed in an area of heterogeneous path loss. To explain the first stage of our algorithm, we need two definitions:

- RN-to-RN communication area: is the area in which RN A can connect to RN or base station B. This means both A and B should be able to receive signals from each other that have a power greater than P_{tr}. In Figure 3, node A can reach the base station. Nodes A and B can communicate, but node B cannot reach the base station directly. However, B can reach the base station indirectly through A. If we define the communication areas of nodes A and B as $X(A)$ and $X(B)$ then the communication area of this pair of nodes is $\{X(A) \cup X(B)\}$. This is the area within which it is feasible to locate SNs, and have their data routed to the base station through the RNs.
- RN-to-SN communication area: similarly, the RN to SN communication area is the area where SN S can connect to RN or base station A. The SN must be within the communication area of the RN, and the RN must be within the communication area of the SN. In Figure 3, the shaded areas show where the SNs can connect to one of the RNs, or to the base station.
- SN sensing area: we assume a disk covering model for the sensing area covered by the SNs. The radius of the disk is S_r. The full details of how this affects SN placement is disclosed below, in Sec. III-C.

Note that. with these definitions, the communication ranges of the SNs and RNs are not fixed constants. Both depend on the transmit power of an SN or RN, the receive sensitivity, the value(s) of the path loss exponent in the cell(s) a signal traverses, and the distance traversed in each cell. Further, based on the path loss map Γ and our communication model, the shapes of the communication areas e.g. $X(A)$ of individual nodes are not necessarily circular. In fact circles are created only when the path loss exponent is equal in all directions. This is not the case when we have spatially varying values of the path loss exponent. Given these definitions, the algorithm operates as follows.

First a preprocessing step finds the places where SNs and RNs can be deployed. We create a rectangular communication grid Ψ of the same spatial extent as Γ, but with the number of cells in Ψ typically larger than the number of cells in Γ. That is, the spatial resolution of Ψ is typically finer than Γ. The number of cells in Ψ is chosen depending on the accuracy desired. Then, based on the nodes' radio ranges, two sets of points are obtained for each vertex of Ψ – one for the RN-to-RN communication area and one for the SN-to-RN communication area. We number the vertices of Ψ from 1 to N, as $j = r \times c$ where r is the row number of the vertex and c is the column number of the vertex, with r and c both starting at 1. Assuming we place an RN at vertex j, the set of vertices at which another RN would be within communication range is called $\rho(j)$. With an RN placed at that same vertex j, the set of vertices from which SNs would be able to reach it is called $\sigma(j)$. We note that $\rho(j)$ and $\sigma(j)$ are both functions of the underlying, spatially varying path loss, as captured by the grid Γ.

The complexity of preprocessing is $O(N^2M)$ where N is the number of vertices in Ψ and M is the number of cells in Γ. This is because we have to calculate the RF path loss between every pair of vertices in Ψ, and each path traverses at most M cells in Γ. There is no multiplicative term for the complexity of finding the path between each pair of vertices because we know the RF signal between any two vertices must follow the shortest Euclidean path. This preprocessing is of relatively high complexity, but we will need these results many times in the following iteration, so the overall complexity of our method is decreased.

Following preprocessing, we begin greedily placing RNs. In each iteration, one RN is placed on the vertex in Ψ that maximizes the incremental number of vertices where SNs could be placed and communicate with an RN. The newly placed RN must be within the communication area of at least one of the base stations, or at least one other RN.

Formally, the algorithm is given Ψ, m and n, and the locations on Ψ of the q base stations. It is also given the sets $\rho(j)$ and $\sigma(j)$ for each vertex j in Ψ. Define:

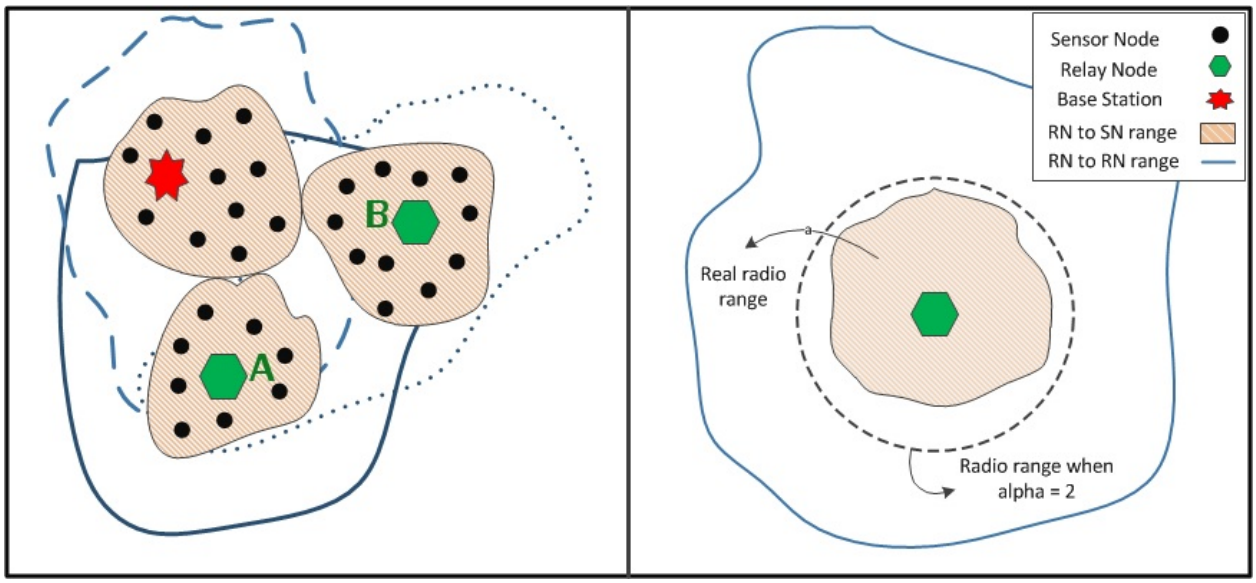

Fig. 3. Radio range in a real environment

$$S = s_1, s_2, ..., s_m$$
$$R = r_1, r_2, ..., r_n$$
$$B = b_1, b_2, ..., b_q$$

where S is the set of SNs, R is the set of RNs and B is the set of base stations.

Without loss of generality, denote the RN placed in iteration k as r_k. Also without loss of generality, denote the vertex where r_k is placed as $j(k)$. Then $\sigma(j(1))$ is the set of SN locations made feasible by placing r_1 at vertex j. We place r_1 at a vertex in Ψ that maximizes $\Upsilon(1)$, and r_k is placed at a vertex that maximizes the set difference:

$$\sigma(j(k)) - \{\cup_{i=1}^{k-1}\sigma(j(i))\}$$

The side constraint requiring connectivity with at least one previously placed RN or base station can be expressed as:

$$\Upsilon(R_k) \cup \Upsilon(B_k) \neq \emptyset$$

where:

$$\Upsilon(R_k) = \{\rho(r_1) \cup \rho(r_2) \cup \cdots \rho(r_{k-1})\} \cap \rho(r_k)$$
$$\Upsilon(B_k) = \{\rho(b_1) \cup \rho(b_2) \cup \cdots \rho(b_q)\} \cap \rho(r_k)$$

The algorithm continues until all n RNs have been placed. Each step in the iteration has complexity $O(N^2)$ because all the unoccupied vertices in Ψ must be examined to determine the incremental number of candidate SN locations enabled by placing an RN there. The first factor of N appears because, although only those vertices within the communication area of the base stations and the already placed RNs must be examined, there are $O(N)$ of these in the worst case. The second factor of N appears because checking the incremental number of candidate SN locations added has complexity $O(N)$.

B. Simulated annealing

Because of the greedy nature of the first stage, it is possible for it to miss RN deployment locations that increase the sensing area. For example if the method reaches a region that has a high path loss exponent, it will prefer to go around it rather than attempting to place RNs there.

To improve the performance of greedy RN placement, we follow it with a Simulated Annealing (SA) stage [19], [44]. Simulated annealing can deliver near-optimal solutions on combinatorial optimization problems. It is suitable for our problem because of the large solution space. Solving problems with SA is relatively simple to formulate and has low memory requirements [41].

One of the most important properties of SA-based algorithms is their ability to escape from local maxima and minima by sometimes choosing worse moves. The probability of accepting a worse move depends on the current state of the system and the change in the cost function. SA algorithms select a random move from the available neighbouring moves rather than necessarily choosing the best available move. If the move is better than the current position, then the move is certainly taken. If not, worse moves are selected based on the following probability [41]:

$$p(x) = e^{-(\Delta E(x)/T)} > R(0,1) \quad (5)$$

$\Delta E(x)$ is the change in the evaluation function as a result of choosing move x, $\mathcal{T}$ is the current temperature of the system and R is a random number uniformly distributed between 0

and 1. We start from a relatively high temperature, $T = 0.9$, and then use the following cooling function:

$$F(T) = c/(1 + log(1 + k)) \quad (6)$$

where $c = 10$ is a constant and k is the current iteration number. In each step $T = F(T)$. Using the cooling function, the temperature decreases as the algorithm progresses.

In each step the SA algorithm selects one RN at random and moves it to a random neighbouring vertex. Then we examine the change in the energy of the system. Maximizing the number of points in the SN-to-RN communication areas of all RNs and base stations is our objective. Therefore, we define the current energy of the system E as the number of points covered by the SN-to-RN communication area of the RNs. The difference in E that results from choosing a move to state x is $\Delta E(x)$. If the answer is better than the best result yet achieved, then we update the best achieved result and move on.

Maintaining connectivity among RNs and from the RNs to the base stations is an important requirement. If connectivity is broken in a particular step, re-establishing it might require many further steps. Therefore, we do not allow RNs to move to locations that would result in a loss of connectivity. In each step we check connectivity by running a depth-first search from the base stations. We also calculate the change in the objective function by calculating the number of points covered by the RNs.

The SA algorithm terminates when a specific number of iterations have been performed, and the system has reached a specific temperature T_d. Other constraints can also be used to limit the runtime, such as continuing until a specific level of sensing coverage is achieved.

The complexity of each iteration of the SA algorithm is $O(n^2 + N^2 n)$, where n is the number of RNs placed and N is the number of cells in the grid Ψ. n^2 appears because we have to check the connectivity between the RNs and the base station. The term $N^2 n$ is for calculating the SN candidate location set. We present the pseudocode for the SA algorithm in Figure 4.

Once the SA algorithm is finished, we have obtained a set of RNs that are connected to the base station and that maximize the SN-to-RN communication area. In the next stage, we must place the SNs. We have a set of candidate locations that are suitable for SN placement and we need to choose among these locations.

```
1: B ← BaseStation {B the set of BNs}
2: R ← RelayNodes {R the set of initially placed RNs}
3: T ← CurrentTemperature
4: E_old ← 0
5: while true do
6:    r ← RN
7:    SelectMove(r) {Move a relay node}
8:    if CalculateNextEnergy() > 0 then
9:       E ← CalculateNextEnergy
10:      ΔE ← E − E_old
11:      if ΔE > 0 then
12:         Move(r)
13:      else
14:         if e^{−ΔE/T} > p_move then
15:            Move(r)
16:      E_old ← E
17:      T ← CurrentTemperature
18: FUNCTION PreProcess
19: for all n ∈ Cells do
20:    n_RelayCover ← covering relay points
21:    n_SensorCover ← covering sensor points
22: for all r ∈ RelayNodes do
23:    Move r to closest grid point
24: END FUNCTION
25: FUNCTION CalculateNextEnergy
26: if any relay node is not connected to a base station then
27:    return −1
28: else
29:    return Calculate number of covered points
30: END FUNCTION
```

Fig. 4. The Simulated Annealing Algorithm

C. SN placement

At this point, we have a connected deployment of RNs and base stations. In addition, we have maximized the number of vertices at which SNs can be deployed and be within reach of one or more RNs or base stations. If the number of available SNs m is too small to make use of all these potential locations, we should place the SNs sparsely to maximize the fraction of the area sensed. If the number of available SNs is large relative to the number of potential locations, we should place the SNs more densely, to improve the sensing coverage of the area. The third stage of our algorithm makes this trade-off.

We assume a disk covering model for the SNs. This means that the SNs either fully sense an area, or do not sense it at all. One alternative is to assume a probabilistic model where the probability of correctly sensing the variable of interest decreases as a given function of the distance from the SN [11], [56].

We modify the SN placement algorithm proposed by Pompili et al [34]. Their method is presented for underwater WSN applications, and is proposed for both two and three dimensional space. They assert that if the distance d between adjacent nodes in the grid is $\sqrt{3}S_r$, where S_r is the sensing range of the SNs, then full sensing coverage of an area can be achieved. Figure 5 illustrates such a deployment.

If the distance between the adjacent SNs in the grid is $2S_r$ or more, then the sensing disks of neighbouring SNs will not overlap, and the area covered by each SN will be maximized. Figure 6 depicts a deployment where $d = 2S_r$.

Our goal is to find the distance d between adjacent SNs that yields the densest possible placement of the available number of SNs, while maximizing the sensing coverage. We test values between $d = \sqrt{3}S_r$ and $d = 2S_r$ and select the smallest

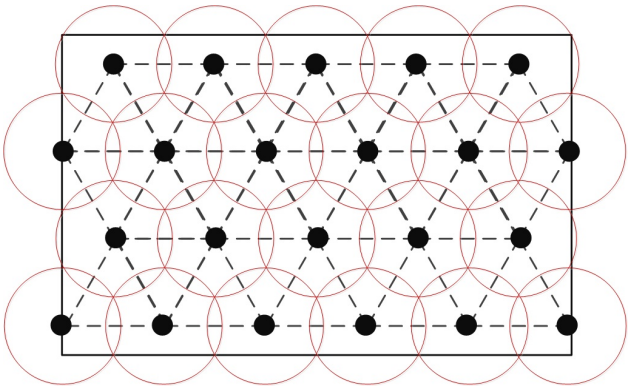

Fig. 5. Full coverage at $d = \sqrt{3}S_r$

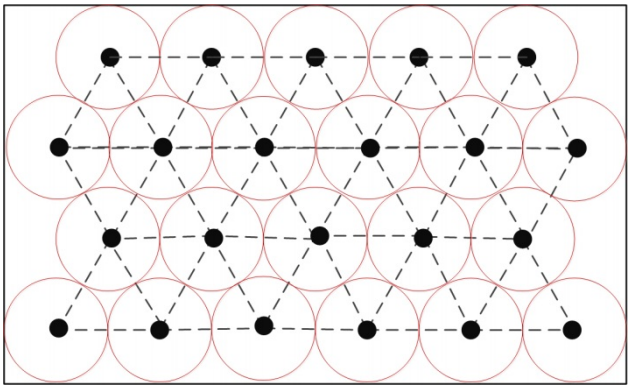

Fig. 6. Coverage is maximized at $d = 2 \times S_r$

value that maximizes the area covered by the SNs. If the step size is w, then the total number of times that we need to test a deployment is $(2-\sqrt{3})S_r/w$. In each iteration, our SN placement algorithm works as follows:

- Creating a grid: In iteration i we create a triangular sensor grid $\mathcal{T}$ and set the distance of adjacent vertices to $\sqrt{3}+(w \times i)$. We align $\mathcal{T}$ with the communication grid Ψ used for RN placement.
- Placing SNs: Based on the locations where RNs were deployed by the previous stage, and the total number of candidate locations thus created for SN placement, we place SNs. If a vertex of the grid $\mathcal{T}$ is in the area covered by the RNs, we select the vertex for SN deployment. All candidate locations provided by the simulated annealing step are guaranteed to be connected to at least one RN or a base station, so additional side constraints can be used to set a precedence order on the candidate locations for SN placement. This process continues until all m SNs have been deployed, or we run out of candidate locations.
- Examining the placement: Based on the deployed SNs, we calculate the total sensing area covered by the SNs. This is done by creating another grid τ that has smaller cell sizes to increase the accuracy of the measurement. We examine whether a point is in the smaller grid τ or not. If a point is covered by two or more SNs, we only consider that point covered once.

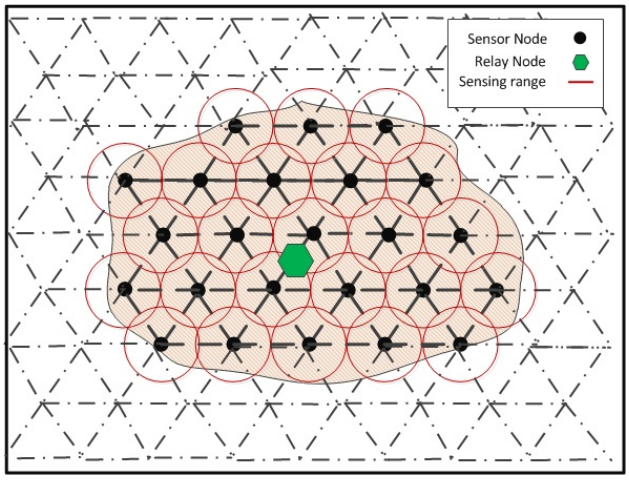

Fig. 7. Example SN placement

We keep the placement with the smallest value of d that maximizes the sensing area covered, and return that placement plus the locations of the RNs as the output of the max-cover 1-connected algorithm. If there is not enough room to place all m SNs, even with $d = \sqrt{3}S_r$, then the algorithm returns the number of surplus SNs. Figure 7 depicts an example placement. The complexity of placing the SNs in each iteration above is $O(V)$ where V is the number of candidate vertices in Ψ where SNs can be placed.

One of the major benefits of our algorithm is that it considers RF path loss in both RN and SN placement. Theoretically, there should not be any errors in communication between nodes. Another benefit is that once the RNs have been placed, a good estimate of the maximum number of SNs that can be usefully placed can be developed. In other words, while we currently solve the problem of using a given number of SNs to maximize sensing coverage, the technique can easily be modified to find the minimum number of SNs that achieve a specified level of sensing coverage.

IV. Experimental Results

In the previous section, we presented the max-cover 1-connected algorithm. This algorithm takes given numbers of SNs and RNs as input and maximizes the sensing area covered by the SNs while maintaining connectivity of the SNs to the base stations through the RNs. We presented a three-stage algorithm that consists of greedy RN placement, a simulated annealing step that improves the results of the first stage and finally SN placement. We implemented our proposed algorithm, and in this section we illustrate its performance.

The first stage of our method places RNs in a greedy, iterative manner. This behaviour leads the algorithm to select points that result in the maximum SN-to-RN communication area. Figure 8 shows a 1000m by 1000m map, with the shaded area in the middle representing a zone with a higher path loss ($\alpha = 3$). The rest of the map has a path loss exponent of $\alpha = 2$. A single base station is located at the top left corner. In this figure, the dots represent the candidate locations of

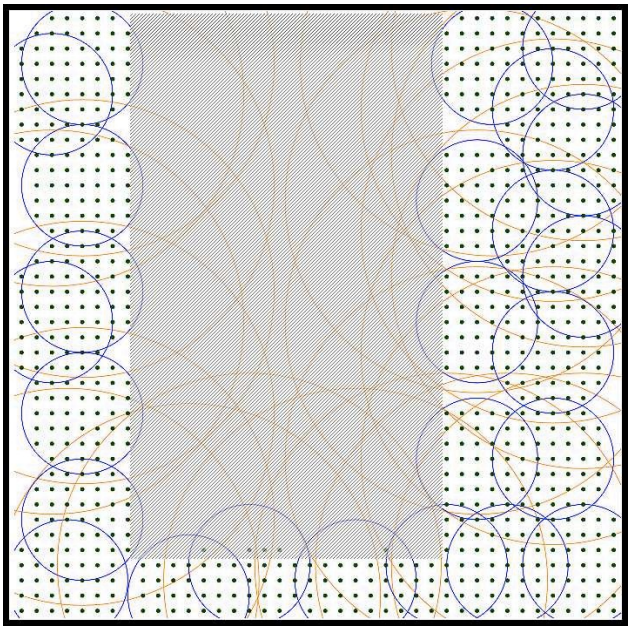

Fig. 8. Greedy RN placement

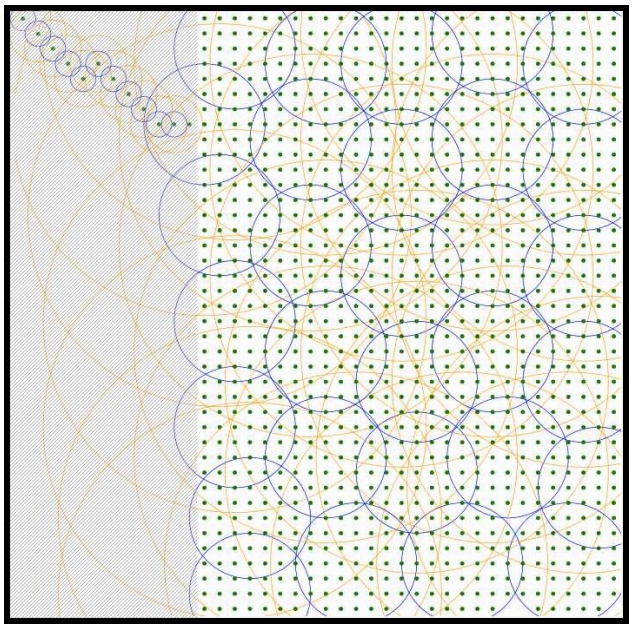

Fig. 9. SA improving RN placement

SNs, and the small circles show the SN-to-RN communication range of the SNs. All the dots within a given small circle can communicate to the RN at the centre of that circle. The large circles show the RN-to-RN communication range. Note that the radio ranges that cross the border between the high-loss and low-loss zone are not accurately depicted due to the change in α. However, our algorithm accounts for the change in α between zones.

As illustrated in Figure 8, the greedy algorithm selects points that maximize the SN-to-RN communication area so that a maximum number of SNs can be deployed. This leads the algorithm to go around the shaded area and connect regions that have lower path loss exponent.

The simulated annealing (SA) stage overcomes this behaviour, and reaches through the high path-loss region to connect it with the rest of the map. As shown in Figure 9, the SA easily finds a route across the high path-loss region. The beginning of this path is illustrated in the top-left corner of the figure. Note that the SN-to-RN communication ranges are much smaller in the high loss region, as are the RN-to-RN communication ranges. Using the same number of RNs, the area provided for SN placement is significantly increased.

Once the SA algorithm has placed the RNs, candidate locations are available for SN placement. We place SNs on a triangular grid. Changing the distance d between vertices on this grid may result in different deployments. We test different values for d and select the smallest value that maximizes the sensing area covered.

A. Comparing greedy alone with greedy plus SA

To determine the benefits of adding the SA step, we compare the area covered by the deployed SNs, with and without SA. To do so, we vary the number of RNs available from 10 to 190 in steps of ten, and run the two variations on a 1000m by 1000m map. The geographic grid Γ is a 10x10 grid, where each cell has a value of α randomly chosen between 2 and 3. The grid Ψ used by the SA and greedy algorithms is 50x50. In these experiments the SA algorithm improves the results of the greedy algorithm by 3.15% on average (see Figure 10).

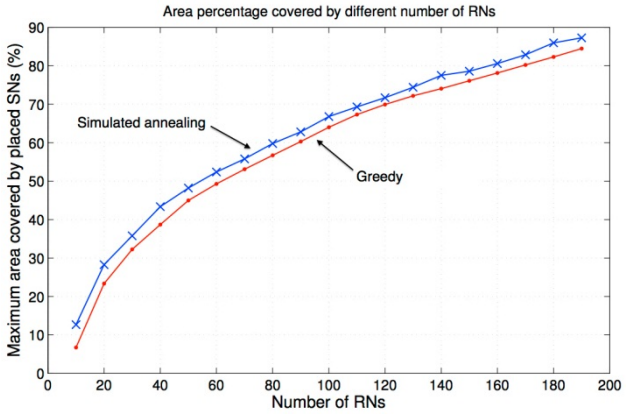

Fig. 10. SA improves on greedy

B. Comparison with the optimal solution

We also compare the results of max-cover 1-connected with an exhaustive backtracking search that examines all possible placements of the RNs and returns the best result. The purpose of this test is to compare the greedy and greedy plus SA algorithms with the optimal solution. Because of the huge search space, the exhaustive search can only be tested on smaller Ψ grids. The results are presented in in Table I. In these tests, the number of RNs is varied from 4 to 7, and the cells of Γ have values of α randomly selected between 2 and

TABLE I
COMPARING MAX-COVER 1-CONNECTED WITH EXHAUSTIVE SEARCH

Map area	Ψ size	RNs	Optimal	SA	Greedy
200×300	6×8	4	39.4%	39.4%	37.01%
200×300	6×8	5	44.3%	44.3%	38.0%
200×300	6×8	6	48.2%	48.2%	43.6%
200×300	6×8	7	54.7%	54.7%	48.2%
250×250	8×8	5	58.5%	58.5%	56.5%
250×250	8×8	6	62.6%	62.6%	62.6%
250×250	8×8	7	69.0%	69.0%	67.2%
300×300	9×9	5	46.5%	46.5%	46.5%
300×300	9×9	6	54.9%	54.9%	45.2%
300×300	9×9	7	62.5%	61.2%	51.3%

3. We examine the performance of the three algorithms with different sizes of the communication grid Ψ.

Greedy plus SA achieves the optimal result in all but one case. In this case, exhaustive search gives a somewhat better result, but it needs 37 minutes to do so. The SA algorithm only requires 10 seconds to run, and by increasing the number of iterations, we achieve the optimal result in this case also.

C. SN placement

We also present results for the third step in our algorithm, of adjusting SN placement on a triangular grid. Table II summarizes the results for different values of d for a sample deployment of 10 RNs in a $1000m \times 1000m$ map with random path loss exponents for its cells. S_r is the sensing radius of each SN, which we assume is 10m in this test.

Table II shows that, in this particular case, the maximum coverage is achieved when $d = \sqrt{3}S_r$. We can deploy slightly fewer SNs by using a vertex spacing of $1.78S_r$. This requires 603 SNs and covers only 0.1% less of the area than the maximum coverage placement of 626 SNs. The relatively low coverage, about $1/6$ of the total area, is the result of the availability of only ten RNs for the large 100 ha. area. This example also provides a very good illustration of some of the non-obvious effects of adjusting d. In this case, increasing d decreases the sensing coverage by decreasing the number of candidate positions for SN placement more than it increases the area covered by the SNs placed. The 15% increase in d results in nearly a 24% decrease in the number of candidate positions. Using this approach, the maximum coverage is highest at the lowest value of d, but of course a larger number of SNs is required for this larger coverage. An alternative design approach is to keep the number of SNs fixed at the smallest value found, i.e. 478. In this case, coverage behaves as intuitively expected, and increases as a function of d.

D. Summary

In this section we presented the performance of our max-cover 1-connected algorithm. The purpose of our algorithm is to maximize the sensing area covered by SNs in a two-tiered network by placing the RNs and SNs. The greedy stage of the algorithm was augmented with simulated annealing, and the benefits of adding SA were tested. We evaluated the overall performance of our method by comparing it to the optimal solution from exhaustive search. In the relatively small cases where exhaustive search could find the optimum, our method achieved results nearly identical to the optimal solution.

V. CONCLUSIONS AND FUTURE WORK

In this paper we study the problem of sensor and relay node placement in an environment of heterogeneous RF path loss. We suggest a simple model for signal propagation to account for the attenuation of the signal in areas with different path loss exponents. We model the area of interest as a geographic grid Γ with a value for the path loss exponent as a property of each cell in the grid.

We solve the problem of maximizing sensing coverage while also maintaining connectivity from SNs to the base stations through the RNs in the environment of heterogeneous loss. We propose a three-stage algorithm that consists of greedy RN placement, simulated annealing improvement and then SN placement. We show how the simulated annealing stage improves greedy RN placement in terms of the candidate locations provided for the SN placement stage. We also compare the performance of our three-stage algorithm with the optimal results from exhaustive search. Our algorithm has identical results compared to the exhaustive search for all our test cases. Our results show that our method can find a high-quality solution quickly.

We suggest the following improvements as future work:

- Based on real environments, real maps can be developed with values for the path loss exponent in each cell based on the properties of the vegetation and terrain in that cell.
- An RF loss model with parameter values based on the types of vegetation in an area could be used to predict how RF signals propagate through different types of terrain.
- Other cooling functions can be tested for the SA stage. In our work, the cooling function depends on the number of iterations. If we increase the number of iterations, more states are examined at higher temperatures. Also, the constant in our cooling function can be tuned based on the size of the communication grid Ψ and number of RNs.

TABLE II
EFFECT OF VARYING GRID SIZE, d

d	# SNs	% Coverage	Coverage at 478 SNs
$\sqrt{3} \times S_r$	626	16.80%	12.83%
$1.78 \times S_r$	603	16.66%	13.21%
$1.83 \times S_r$	563	16.30%	13.84%
$1.88 \times S_r$	520	15.56%	14.30%
$1.93 \times S_r$	503	15.30%	14.53%
$2.0 \times S_r$	478	14.86%	14.86%

- Our current method can be improved if we create the sensor-placement grid $\mathcal{T}$ at different starting positions. This means that rather than aligning the origins of our Ψ and $\mathcal{T}$ grids, we can shift the triangular grid $\mathcal{T}$ by $r/2$ so that other SN deployments based on the shifted grid can be examined too.
- Rather than using thermal functions, quantum fluctuations can be utilized to find the global minima of our system. Because of our discrete search space, using this method may lead to faster and better results.
- To better escape local minima, stochastic tunnelling based on Monte Carlo sampling can be used. In this way, instead of only moving to neighbouring positions, we would be able to jump to more distant states.
- Rather than always keeping the step size of one, dynamic step sizes can be used for moving to neighbouring nodes. If we are moving towards a better answer, we decrease the step size and if we are far from a good solution, we increase the step size.

REFERENCES

[1] N. Aitsaadi, N. Achirt, K. Boussetta, and G. Pujolle. A tabu search approach for differentiated sensor network deployment. In *IEEE Consumer Communications and Networking Conf.*, CCNC '08, pages 163–167, 2008.

[2] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: A survey. *Computer Networks*, 38(4):393 – 422, 2002.

[3] X. Bai, S. Kumar, D. Xuan, Z. Yun, and T. H. Lai. Deploying wireless sensors to achieve both coverage and connectivity. In *Proc. 7th ACM Int'l Symp. on Mobile Ad Hoc Networking and Computing*, MobiHoc '06, pages 131–142, New York, NY, USA, 2006. ACM.

[4] X. Bai, D. Xuan, Z. Yun, T. H. Lai, and W. Jia. Complete optimal deployment patterns for full-coverage and k-connectivity ($k \leq 6$) wireless sensor networks. In *Proc. 9th ACM Int'l Symp. on Mobile Ad Hoc Networking and Computing*, MobiHoc '08, pages 401–410, New York, NY, USA, 2008. ACM.

[5] E. Biagioni and G. Sasaki. Wireless sensor placement for reliable and efficient data collection. In *Proc. 36th Annual Hawaii Int'l Conf. on System Sciences*, HICSS '03, pages 127.2–, Jan. 2003.

[6] R. Caldeirinha and M. Al-Nuaimi. A novel FDTD based model for prediction of bistatic RCS of single leaves and trees. In *Eleventh Int'l Conf. on Antennas and Propagation*, volume 2, pages 783–786, 2001.

[7] A. Cerpa, J. Elson, D. Estrin, L. Girod, M. Hamilton, and J. Zhao. Habitat monitoring: Application driver for wireless communications technology. *SIGCOMM Comput. Commun. Rev.*, 31(2 supplement):20–41, Apr. 2001.

[8] G. Chen and S. Cui. Relay node placement in two-tiered wireless sensor networks with base stations. *J. Combinatorial Optimization*, pages 1–10, 2012. 10.1007/s10878-012-9451-5.

[9] Y. Chen and Q. Zhao. On the lifetime of wireless sensor networks. *IEEE Comm. Letters*, 9(11):976 – 978, Nov. 2005.

[10] X. Cheng, D.-Z. Du, L. Wang, and B. Xu. Relay sensor placement in wireless sensor networks. *Wireless Networks*, 14:347–355, 2008. 10.1007/s11276-006-0724-8.

[11] S. S. Dhillon and K. Chakrabarty. Sensor placement for effective coverage and surveillance in distributed sensor networks. In *Proc. IEEE Wireless Communications and Networking Conf.*, WCNC '03, pages 1609–1614, 2003.

[12] M. H. C. Dias, A. Rotava, F. G. Andrade, R. A. Alem, M. A. K. Melo, and J. C. A. Santos. Path loss measurements of HF/VHF land links in a Brazilian Atlantic rainforest urban site. *IEEE Antennas and Wireless Propagation Letters*, 10:1063–1067, 2011.

[13] Google Earth V4.3.7284.3916 (beta), July 2008. Mandy Lake, Alberta, 12U 367550m E 5907750m N, Eye alt 1.6 km, DigitalGlobe 2013.

[14] A. Efrat, S. Har-Peled, and J. Mitchell. Approximation algorithms for two optimal location problems in sensor networks. In *2nd Int'l Conf. on Broadband Networks*, BroadNets 2005, pages 714 – 723 Vol. 1, Oct. 2005.

[15] S. Gabriele and P. Di Giamberardino. Mobile sensors networks under communication constraints. *WSEAS Trans. on Systems*, 7(3):165–174, Mar. 2008.

[16] X. Han, X. Cao, E. Lloyd, and C.-C. Shen. Fault-tolerant relay node placement in heterogeneous wireless sensor networks. *IEEE Trans. on Mobile Computing*, 9(5):643 –656, May 2010.

[17] D. S. Hochbaum and W. Maass. Approximation schemes for covering and packing problems in image processing and VLSI. *J. ACM*, 32:130–136, 1985.

[18] A. Kashyap, S. Khuller, and M. A. Shayman. Relay placement for higher order connectivity in wireless sensor networks. In *IEEE 25th Int'l Conf. on Communications*, INFOCOM 2006, pages 1–12, 2006.

[19] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220, 4598:671–680, 1983.

[20] S. Kumar, T. H. Lai, and J. Balogh. On k-coverage in a mostly sleeping sensor network. In *Proc. 10th Annual Int'l Conf. on Mobile Computing and Networking*, MobiCom '04, pages 144–158, New York, NY, USA, 2004. ACM.

[21] W. C. Y. Lee. *Mobile Communications Design Fundamentals*. John Wiley & Sons, Inc., New York, NY, USA, 2nd edition, 1992.

[22] J. Li, L. L. Andrew, C. H. Foh, M. Zukerman, and H.-H. Chen. Connectivity, coverage and placement in wireless sensor networks. *Sensors*, 9(10):7664–7693, 2009.

[23] B. Lin, P. Ho, L. Xie, and X. Shen. Relay station placement in IEEE 802.16j dual-relay MMR networks. In *IEEE Int'l Conf. on Communications*, ICC '08, pages 3437–3441, 2008.

[24] F. Lin and P. Chiu. A near-optimal sensor placement algorithm to achieve complete coverage-discrimination in sensor networks. *IEEE Comm. Letters*, 9(1):43 – 45, Jan. 2005.

[25] G. Lin and G. Xue. Steiner tree problem with minimum number of Steiner points and bounded edge-length. *Information Proc. Letters*, 69:53–57, 1999.

[26] E. L. Lloyd and G. Xue. Relay node placement in wireless sensor networks. *IEEE Trans. on Computers*, 56:134–138, 2007.

[27] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M. B. Srivastava. Coverage problems in wireless ad-hoc sensor networks. In *IEEE 20th Int'l Conf. on Communications*, volume 3 of *INFOCOM 2001*, pages 1380–1387, 2001.

[28] Y. S. Meng, Y. H. Lee, and B. C. Ng. Study of propagation loss prediction in forest environment. *Progress in Electromagnetics Research B*, 17:117–133, 2009.

[29] S. Misra, S. D. Hong, G. Xue, and J. Tang. Constrained relay node placement in wireless sensor networks to meet connectivity and survivability requirements. In *IEEE 27th Int'l Conf. on Computer Communications*, INFOCOM 2008, pages 281 –285, Apr. 2008.

[30] J. O'Rourke. *Art Gallery Theorems and Algorithms*. Oxford University Press, Inc., New York, NY, USA, 1987.

[31] J. Pan, L. Cai, Y. T. Hou, Y. Shi, and S. X. Shen. Optimal base-station locations in two-tiered wireless sensor networks. *IEEE Trans. on Mobile Computing*, 4:458–473, 2005.

[32] J. D. Parsons. *The Mobile Radio Propagation Channel, 2nd Edition*. Wiley, 2 edition, Nov. 2000.

[33] J. Polastre, R. Szewczyk, A. Mainwaring, D. Culler, and J. Anderson. Analysis of wireless sensor networks for habitat monitoring. *Wireless Sensor Networks*, pages 399–423, 2004.

[34] D. Pompili, T. Melodia, and I. F. Akyildiz. Deployment analysis in underwater acoustic wireless sensor networks. In *Proc. 1st ACM Int'l Workshop on Underwater Networks*, WUWNet '06, pages 48–55, New York, NY, USA, 2006. ACM.

[35] V. Ravelomanana. Extremal properties of three-dimensional sensor networks with applications. *IEEE Trans. on Mobile Computing*, 3(3):246–257, July 2004.

[36] N. C. Rogers, A. Seville, J. Richter, D. Ndzi, N. Savage, R. F. S. Caldeirinha, A. K. Shukla, M. O. Al-nuaimi, K. Craig, E. Vilar, and J. Austin. A generic model of 1-60 GHz radio propagation through vegetation. Technical report, Radiocommunications Agency, 2002.

[37] S. Shukla, N. Bulusu, and S. Jha. Cane-toad monitoring in Kakadu national park using wireless sensor networks. In *Proc. of APAN*, 2004.

[38] A. So and B. Liang. Enhancing WLAN capacity by strategic placement of tetherless relay points. *IEEE Trans. on Mobile Computing*, 6:522–535, 2007.

[39] A. Srinivas, G. Zussman, and E. Modiano. Mobile backbone networks: Construction and maintenance. In *Mobile Ad Hoc Networking and Computing*, pages 166–177, 2006.

[40] G. Stüber. *Principles of Mobile Communication*. Springer Verlag, 2011.
[41] B. Suman and P. Kumar. A survey of simulated annealing as a tool for single and multiobjective optimization. *J. Operational Research Society*, 57:1143–1160, 2006.
[42] R. Szewczyk, E. Osterweil, J. Polastre, M. Hamilton, A. Mainwaring, and D. Estrin. Habitat monitoring with sensor networks. *Commun. ACM*, 47(6):34–40, June 2004.
[43] J. Tang, B. Hao, and A. Sen. Relay node placement in large scale wireless sensor networks. *Comput. Commun.*, 29(4):490–501, Feb. 2006.
[44] E. Teller, N. Metropolis, and A. Rosenbluth. Equation of state calculations by fast computing machines. *J. Chem. Phys*, 21(13):1087–1092, 1953.
[45] D. Tian and N. D. Georganas. A coverage-preserving node scheduling scheme for large wireless sensor networks. In *Mobile Computing and Networking*, pages 32–41, 2002.
[46] B. Wang. Sensor placement for complete information coverage in distributed sensor networks. *J. Circuits, Systems, and Computers*, 17:627–636, 2008.
[47] N. Wang, N. Zhang, and M. Wang. Wireless sensors in agriculture and food industry: Recent development and future perspective. *Computers and Electronics in Agriculture*, 50(1):1 – 14, 2006.
[48] Q. Wang, K. Xu, H. Hassanein, and G. Takahara. Minimum cost guaranteed lifetime design for heterogeneous wireless sensor networks (WSNs). In *IEEE 24th Int'l Performance, Computing, and Communications Conf.*, IPCCC 2005, pages 599 – 604, Apr. 2005.
[49] Y.-C. Wang, C.-C. Hu, and Y.-C. Tseng. Efficient placement and dispatch of sensors in a wireless sensor network. *IEEE Trans. Mobile Comput.*, 7(2):262–274, 2008.
[50] K. Xu, H. Hassanein, G. Takahara, and Q. Wang. Relay node deployment strategies in heterogeneous wireless sensor networks: Single-hop communication case. In *IEEE Global Telecomm. Conf. 2005*, volume 1 of *GLOBECOM'05*, pages 5–pp. IEEE, 2005.
[51] K. Xu, Q. Wang, H. Hassanein, and G. Takahara. Optimal wireless sensor networks (WSNs) deployment: Minimum cost with lifetime constraint. In *IEEE Int'l Conf. on Wireless And Mobile Computing, Networking And Communications 2005*, volume 3 of *WiMob '05*, pages 454 – 461 Vol. 3, Aug. 2005.
[52] N. Xu. A survey of sensor network applications. *Survey Paper for CS694a*, 2002.
[53] D. Yang, X. Fang, G. Xue, and J. Tang. Relay station placement for cooperative communications in WiMAX networks. In *IEEE Global Telecomm. Conf. 2010*, GLOBECOM '10, pages 1–5, 2010.
[54] M. Younis and K. Akkaya. Strategies and techniques for node placement in wireless sensor networks: A survey. *Ad Hoc Networks*, 6(4):621–655, June 2008.
[55] H. Zhang and J. C. Hou. Maintaining sensing coverage and connectivity in large sensor networks. *Ad Hoc & Sensor Wireless Networks*, 1, 2005.
[56] J. Zhang, T. Yan, and S. Son. Deployment strategies for differentiated detection in wireless sensor networks. In *IEEE 3rd Sensor and Ad Hoc Communications and Networks*, volume 1 of *SECON '06*, pages 316 –325, Sept. 2006.
[57] W. Zhang, G. Xue, and S. Misra. Fault-tolerant relay node placement in wireless sensor networks: Problems and algorithms. In *IEEE 26th Int'l Conf. on Computer Communications*, INFOCOM 2007, pages 1649 – 1657, May 2007.
[58] X. Zhang and S. B. Wicker. How to distribute sensors in a random field? In *Proc. 3rd Int'l Symp. on Information Processing in Sensor Networks*, IPSN '04, pages 243–250, New York, NY, USA, 2004. ACM.