Cost-Efficient Deployment for Full-Coverage and Connectivity in Indoor 3D WSNs

Marc T. Kouakou,†1 Shinya Yamamoto,‡2 Keiichi Yasumoto†1 and Minoru Ito†1

In WSNs, coverage of the target field and wireless connectivity among sensor nodes are important criteria in terms of the performance required for WSN. In indoor environments, the target field is in general 3D space, thus WSNs deployed for such environments are called 3D WSNs. The sensor node deployment problem for 3D coverage and connectivity is NP-hard even without obstacles in the target field. Furthermore, no study has systematically examined the optimal 3D WSN deployment considering both obstacles and deployment cost. In this paper, we propose a new heuristic algorithm for computing a near optimal sensor node deployment that minimizes the cost for achieving the full coverage and node connectivity of a 3D target space with obstacles. First, we represent the monitoring area as well as the sensor node deployable area in the target 3D space, by a set of grid points. Our algorithm puts sensor nodes one by one on a grid point of the deployable area in the descendant order of the cost-performance value (i.e., how many monitoring space points are covered by the deployable area point per unit deployment cost) of the deployable area points. Then, the algorithm adds extra nodes to cover the shadow area of each node’s sensing region cut off by the obstacles. Moreover, to ensure the connectivity among all WSN nodes, the algorithm adds extra nodes and/or moves each unconnected sensor node one by one towards the closest connected sensor node in order to reduce the number of extra nodes. We implemented our proposed method in the UbiREAL simulator and evaluated the performance through simulations. As a result, we have confirmed that our proposed method can provide reliable and cost-efficient solutions for WSN deployment in indoor environments.

1. Introduction

The recent growth of context-aware applications in indoor environments has brought new research approaches in diverse fields including 3D WSNs, which have gained increasing interests due to their various uses in home automation and smart environment monitoring (location information collection, target’s position tracking, etc). One of the key challenges for 3D WSNs is how to achieve full coverage of the target monitoring space and sufficient network connectivity between sensor nodes while maintaining a low deployment cost. Crucially, we need to optimize the sensing performance of indoor 3D WSNs that characterizes how well each point in the sensing field (service area) is monitored by the WSN both for economic and technical reasons. Given a sensor network, the connectivity problem is to determine how well the sensor nodes communicate with each other. Due to the actual market cost of 3D sensor nodes and the requirements of their applications in real-world, the accuracy of the deployment has become a strict requirement.

The 3D WSN coverage problem is NP-hard since it includes the minimum set covering problem known to be NP-hard. Numerous researches have addressed this problem and some heuristic methods have been studied either by reducing the 3D geometry to the 2D [1–5] or by neglecting the obstacles existing in the target space [6–8]. Due to abstraction of the target space, these methods may lead to some deployment errors caused by the inaccuracy of the obtained results. We need to develop an optimal deployment method for 3D WSNs taking into account sensor node deployment cost different among installation locations and obstacles (walls, ceiling, appliances, furniture, etc) that may obstruct sensing region and wireless connectivity in the target area. Existing studies do not consider the above issues.

In this paper, we propose a heuristic method for determining the near optimal sensor nodes deployment while minimizing the cost for achieving the full coverage and the connectivity in a three dimensional indoor environment with obstacles. We first discretize the problem by reducing the 3D monitoring space to a set of grid points (monitoring points) and the deployable space to a set of grid points (deployable points). Given the cost of each deployable point, our algorithm computes the number of covered monitoring points per unit deployment cost and put sensor nodes one by one, until achieving the full-coverage, on deployable points with more monitoring points covered by those points than others. A
monitoring point is considered to be covered if this point is located within a node’s sensing range, and there is no obstacle that obstructs the monitoring point.

To reach the desired connectivity of the network, the algorithm computes the connectivity of each deployed sensor node and changes the location of each unconnected sensor node by moving it towards the closest connected sensor node. Then, if unconnected nodes still exist, it will add extra sensor nodes to ensure the connectivity of all WSN nodes.

We implemented our proposed algorithm in a smart-space simulator UbiREAL [9], designed for developing and testing ubiquitous applications. The function implemented in UbiREAL allows us to visualize the sensor nodes deployment computed by our method on a given indoor environment which includes home appliances and furniture.

The remainder of this paper is structured as follows. Section 2 outlines related work in WSN coverage and connectivity. After formulating the problem in Section 3, we describe our newly proposed method in Section 4. We discuss the implementation and evaluation of the proposed method in Section 5 and conclude the paper in Section 6.

2. Related Work

The coverage indicates how well information is tracked and collected by a WSN [6], while the connectivity reflects the quality of communication between the sensor nodes. Recently, many researches have addressed the deployment problem in WSN and various solutions have been proposed regarding the coverage and the connectivity. We classify the set of related existing studies in two categories.

2.1 Deployment problem in 2D

There have been several studies on the deployment problem in 2D space. Huang et al., in [1], proposed a polynomial time algorithm to efficiently solve the 2-dimensional coverage problem in terms of the number of deployed sensors. One closely related problem in the literature is the Art Gallery Problem in which we are to determine the minimum number of observers necessary to cover the whole gallery. Bai et al., in [4, 5], proposed an optimal deployment pattern to achieve both full coverage and 2-connectivity, and proved its optimality for arbitrary ratio of \( r_c/r_s \) where \( r_s \) and \( r_c \) mean the radius of each sensor node’s sensing range and communication range, respectively. In [10], a new set of optimal deployment patterns is proposed to address the mutation phenomenon occurring for certain ratios of \( r_c/r_s \). Though the sensor node deployment problem in the 3D space is NP-Hard, it can still be solved optimally in the 2D plane. The proposed solution determines whether a WSN provides full coverage of the monitoring area by investigating whether every point in the monitoring area is covered by at least \( k \) sensors (\( k \)-coverage).

An approach based on the Delaunay Triangulation has been used in [2] to solve the problem of coverage holes and plan the position of the sensors in an environment with obstacles.

Although it is possible to solve the deployment problem in 2D space, it may result in some inaccuracies that will lead to some deployment errors if we apply these 2D deployment methods to 3D coverage problem as is.

2.2 Deployment problem in 3D obstacle-free space

Alam and Haas studied the problem of coverage and connectivity for 3D spaces in [11]. Based on the Kepler’s conjecture, they suggested a deployment pattern that generates the Voronoi tessellation of truncated octahedron in 3D space. Moreover, in [8] and [7], Bai et al. designed several sets of lattice patterns to achieve full-coverage and \( k \)-connectivity in three dimensional space and proved their optimality under constraints of regularity. The above studies suppose that the sensor nodes can be placed at any point of the 3D space and do not consider the deployment cost at each point in the target space. Moreover, the suggested deployment patterns do not take into account the obstacles that may exist in the target space.

3. Problem Statement

In this section, we give assumptions in our proposed method and formulate the problem of sensor nodes deployment in a 3D indoor environment. We summarize
the symbols that are used in this section in Table 1.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A$</td>
<td>target area</td>
</tr>
<tr>
<td>$A_d$</td>
<td>sensor node deployable area ($A_d \subseteq A$)</td>
</tr>
<tr>
<td>$A_m$</td>
<td>monitoring area ($A_m \subseteq A$)</td>
</tr>
<tr>
<td>$\text{cost}(p)$</td>
<td>cost for deploying a sensor node at location $p$</td>
</tr>
<tr>
<td>$O$</td>
<td>set of obstacles</td>
</tr>
<tr>
<td>$S$</td>
<td>set of sensor nodes to achieve full coverage and connectivity</td>
</tr>
<tr>
<td>$s.pos$</td>
<td>location where sensor node $s$ is deployed ($s \in S$)</td>
</tr>
<tr>
<td>$r_s$</td>
<td>sensor node's sensing radius</td>
</tr>
<tr>
<td>$r_c$</td>
<td>sensor node's communication radius</td>
</tr>
<tr>
<td>$\text{sink}$</td>
<td>sink node that collects data from all deployed sensor nodes</td>
</tr>
</tbody>
</table>

### 3.1 Assumptions

We denote the target 3D space by $A$. In $A$, there are some objects called obstacles that may obstruct communication and/or sensing by sensor nodes. We denote the set of obstacles by $O$.

We call the subarea to be monitored in $A$, monitoring area and denote it by $A_m$. The subarea(s) of $A$ where sensor nodes can be installed is called deployable area and we denote it by $A_d$. The cost for installing a sensor node changes depending on the location in $A_d$\footnote{In general, installing on ceiling/wall will be cheaper than in the air.}. Let $\text{cost}(p)$ denote the cost for installing a sensor node at location $p \in A_d$.

Sensor Nodes: We consider only static nodes to be deployed in the 3D WSNs: they cannot move from their location after being deployed. Each sensor node has wireless communication capability. Its sensing and communication ranges are represented by spheres of some radius with the node at its center, respectively. We assume that all the sensor nodes have identical sensing radius $r_s$ and communication radius $r_c$. Sensor nodes within distance $r_c$ can communicate with each other (no obstacles case). We denote the sensing range of sensor node $s$ by $s.range$. Sensing range and communication range of each sensor node can be obstructed by obstacles. As shown in Fig. 1, we assume that the sensor node cannot sense the information from the shadow area in the sensing region generated by obstacles. Moreover, we assume that two sensor nodes within distance $r_c$ can communicate with each other only if the line of sight (LoS) exists between them, that is, no obstacle obstructs the straight line between the nodes.

### 3.2 Problem Formulation

Given a three dimensional indoor area $A$, a set $O$ of obstacles in $A$, sensor node deployable area ($A_d \subseteq A$) with cost function $\text{cost}(p)$, the monitoring area ($A_m \subseteq A$), a sink node $\text{sink}$ and its position $\text{sink}.pos$, our target problem is to determine the number of sensor nodes and their installing positions that guarantee the full-coverage of $A_m$ and the connectivity of all sensor nodes to $\text{sink}$ while minimizing the total sum of deployment cost.

Let $S$ denote the set of sensor nodes for deployment. Each sensor node $s \in S$ must be deployed in $A_d$. When we denote the deployed position of sensor node $s$ by $s.pos$, the following equation holds.

$$\forall s \in S, s.pos \in A_d$$ \hspace{1cm} (1)

**Coverage of $A_m$:** We say that each monitoring point $m$ ($m \in A_m$) is covered by the sensor node $s$ only if it is located within the sensing range $s.range$ and not in the shadow area.

We say that $m$ is $k$-covered when it is covered by at least $k$ sensor nodes. For $k$-coverage of the monitoring area $A_m$, the following equation must hold.

$$\forall m \in A_m, \vert \{ s \in S \cap m \in s.range \setminus \text{Shadow}(s, O) \} \vert \geq k$$ \hspace{1cm} (2)

Here, $\text{Shadow}(s, O)$ represents the shadow area in $s.range$ by obstacles $O$ (as shown in Fig. 1).

**Connectivity:** We say that a sensor node $s$ is connected when it has a connected path to the sink node $\text{sink}$. To ensure the connectivity of the whole WSN, the following equation must hold.
∀s ∈ S, connected(s, sink) = true

where

\( connected(s, s') \) \( \overset{\text{def}}{=} \ | \ (s, s') | \leq r_c \wedge LoS(s, s', O) = true \)
\( \vee \exists s_1, \ldots, s_i \in S(1 \leq i) \)
\( | (s, s_1) | \leq r_c \wedge LoS(s, s_1, O) = true \)
\( \wedge | (s_1, s_2) | \leq r_c \wedge LoS(s_1, s_2, O) = true \wedge \ldots \)
\( \wedge | (s_i, s') | \leq r_c \wedge LoS(s_i, s', O) = true \)

Here, \( LoS(s, s', O) \) is the boolean function that becomes true only if no obstacles of \( O \) obstruct the straight line between \( s \) and \( s' \).

Using the above constraints, the objective function of our target problem is given by the following equation:

\[
\text{minimize } \sum_{s \in S} \text{cost}(s, \text{pos}) \\
\text{subject to constraints (2) – (4)}
\] (5)

4. The Proposed Solution

4.1 Preliminaries

In order to simplify the computation in the proposed algorithm, first, we discretize the monitoring area and the deployable area by a set of grid points, respectively.

The set of \( \text{monitoring points} \ M \) is the set of points to be covered by the wireless sensor network. As shown in Fig. 2, the monitoring points are represented by a set of grid points in the monitoring area \( A_m \).

The set of \( \text{deployable points} \ D \) is the set of points on which sensor nodes can be eventually put. As shown in Fig. 2, the deployable points are represented by a set of grid points in the deployable area \( A_d \).

We assume that the target 3D space is sufficiently covered if all the monitoring points \( M \) are sufficiently covered.

Our algorithm consists of two phases: \( \text{initial deployment} \) and \( \text{node selection} \). Also, we assume that the sensor node’s sensing radius is no more than half of its communication radius: \( r_s \leq \frac{1}{2} r_c \). We show the pseudo code of the algorithm in Algorithm 1.

4.2 Initial Deployment

As shown in lines 1-2 of Algorithm 1, the algorithm first computes the 3D Euclidean positions of the monitoring points \( M \) and the deployable points \( D \), given \( A_m, A_d \), and the grid interval \( \Delta \). Here, the grid interval \( \Delta \) is no more than the node’s communication range \( r_c (\Delta \leq r_c) \) and the smaller \( \Delta \) will produce the more accurate deployment although the computation cost will also increase.

4.3 Node Selection

In the next step, the algorithm selects the deployable point with the highest \( dc \) ratio. The function to calculate the \( dc \) ratio is defined by:

\[
dc_{\text{ratio}}(d) = \frac{| \{ m \in M \wedge m \in \text{sphere}(d, r_s) \setminus \text{Shadow}(d, r_s, O) \} |}{\text{cost}(d)}
\] (6)

where \( \text{sphere}(d, r_s) \) means the sphere with radius \( r_s \) centered at a deployable
point \(d\) and \(\text{Shadow}(d, r_s, O)\) means the shadow area generated by obstacles \(O\) in \(\text{sphere}(d, r_s)\).

At first, the \(dc\) ratio is computed for each deployable point \(d \in D\) and the algorithm selects \(d^*\) with the highest \(dc\) ratio as shown in lines 5-11 of Algorithm 1.

A sensor node \(s\) is then placed on \(d^*\) (line 12). The coverage level of the monitoring points newly covered by \(s\) is updated, and those with a sufficient coverage level are removed from the set of monitoring points (lines 14-15). We then once again compute the \(dc\) ratio at each empty deployable point and repeat the above steps until all the monitoring points are sufficiently covered.

After reaching the required level of coverage of the 3D environment, the algorithm focuses on the connectivity problem.

Let \(s_u\) be the sensor node element of unconnected sensor nodes \(U\) under consideration. The algorithm selects a node \(s_c\) in the connected sensor nodes \(C\) that is the closest to \(s_u\) and computes the new "virtual" position \(p'\) of \(s_u\) by moving it toward \(s_c\) as long as the set of monitoring points initially covered by \(s_u\) remains unchanged. If the sensor node \(s_u\) is still unconnected after changing its position, extra nodes are put on the line segment with endpoints \(s_u\) and \(s_c\) so that \(s_u\) and \(s_c\) are connected, as shown in Fig. 3.

**Algorithm 1** Node Selection for 3D Coverage and Connectivity

1: \(D = \text{computeDeployablePoints}(A_d, \Delta)\)
2: \(M = \text{computeMonitoringPoints}(A_m, \Delta)\)
3: \(S = \emptyset, \text{maxdc} = 0\)
4: while \(M\) is not \(k\)-covered do
   5:      for each \(d \in D\) do
   6:         \(dc = \text{coverMonitoringPoints}(d) / \text{deploymentCost}(d)\)
   7:            if \(\text{maxdc} < dc\) then
   8:               \(\text{maxdc} = dc\)
   9:                  \(d^* = d\)
  10:           end if
  11:      end for
  12:      \(s:pos = d^*\)
  13:      \(S = S \cup \{s\}\)
  14:      \(D = D \setminus \{d^*\}\)
  15:      \(\text{updateCoverageLevel}(M, S)\)
  16:  end while
17: \(C = \text{connectedSensors}(S)\)
18: \(U = S \setminus C\)
19: for each \(s_u \in U\) do
   20:      \(s_c = \text{closestSensor}(C, s_u)\)
   21:      \(s_u:pos = \text{moveTowards}(s_c)\)
   22:         if not \(\text{connected}(s_u, s_c)\) then
   23:            \(\text{addMoreSensors}(S, s_u, s_c)\)
   24:         end if
   25:      \(U = U \setminus \{s_u\}\)
   26:      \(C = C \cup \{s_u\}\)
27: end for

**4.4 Example**

In the following example, we illustrate the proposed algorithm step by step.
We are given an indoor environment in which we have to deploy sensor nodes to ensure the full-coverage and the connectivity of the WSN. A sink node is placed in the 3D space at an arbitrary position. The deployment cost is different among the positions in the sensor node deployable area.

After the initial deployment (generation of the monitoring points and the deployable grid points), the $dc$ ratio is computed at each deployable grid point as shown in Fig. 4(a). Then, the algorithm puts a sensor node on the grid point having the highest $dc$ ratio ($0.85$). The two previous steps are iteratively executed until all the monitoring points are sufficiently covered as shown in Fig. 4(b). The connectivity of each deployed sensor node is then computed and all the unconnected sensor nodes are moved towards the nearest connected nodes, respectively, as shown in Fig. 4(c). In this case, unconnected sensor nodes still remain, new nodes are added to ensure the wireless connectivity in Fig. 4(c). The final result of our deployment method is shown in Fig. 4(d).

5. Implementation - Performance Evaluation

We implemented our proposed solution in the UbiREAL simulator that allows us to easily visualize and simulate the behavior of home appliances. We then conducted simulation-based experiments, to show the performance of our algorithm.

5.1 Simulation Configuration

For the simulation, we are given an indoor environment in which we are to deploy wireless sensor nodes for full-coverage and 1-connectivity. The simulation
configuration parameters are shown in Table 3.

Table 2: Simulation Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target space</td>
<td>10m x 10m x 3m</td>
</tr>
<tr>
<td>Node’s sensing range $r_s$</td>
<td>1.5m</td>
</tr>
<tr>
<td>Node’s connectivity range $r_c$</td>
<td>5m</td>
</tr>
</tbody>
</table>

Fig. 5: Initial Configuration (Front View)

The monitoring area $A_m$ is a plane located at mid height of the target space ($h = 1.5m$) as shown in Fig. 5. Moreover, we divided the deployable area in horizontal (ceiling deployment) and vertical (wall deployment) regions with a variable deployment cost. We set the deployment cost for each region as shown in Table 3.

We then set the grid spacing $\Delta$ to 0.5m in the deployable and the monitoring grids.

The program outputs both the position and the cost of all the sensor nodes required to fully cover the defined monitoring area while ensuring the nodes’ connectivity. We applied the same conditions to the triangular lattice pattern proposed in [4] and compared the obtained deployment cost of the method in [4] with our proposed method.

Table 3: Deployment Cost

<table>
<thead>
<tr>
<th>Regions</th>
<th>cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Horizontal Regions</td>
<td></td>
</tr>
<tr>
<td>h=3.0m</td>
<td>5</td>
</tr>
<tr>
<td>h=2.5m</td>
<td>10</td>
</tr>
<tr>
<td>h=2.0m</td>
<td>15</td>
</tr>
<tr>
<td>h=1.5m</td>
<td>20</td>
</tr>
<tr>
<td>Vertical Regions</td>
<td>5</td>
</tr>
</tbody>
</table>

5.2 Result of the comparison

The experimental results are shown in Table 4. By using the triangular lattice pattern method, that has been proved to be an optimal deployment method, we achieve the full-coverage and 1-connectivity of the target area with 22 sensor nodes scattered in a regular pattern. However, since all the deployed positions of sensor nodes are in the same horizontal plane ($h=1.5m$), the same deployment cost is applied to all of them, i.e. 20 as shown in Table 3. The total deployment cost is then 440. On the other hand, our proposed method achieved the deployment for the same monitoring area with a total deployment cost of 270. Although, the number of deployed sensor nodes is much larger than the number obtained with the triangular pattern, the total deployment cost is about 40% smaller.

We confirmed that our proposed method provides better cost-efficient deployment results than existing methods.

Table 4: Comparative results of the two methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Number of sensors</th>
<th>Total cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cost-efficient deployment</td>
<td>39</td>
<td>270</td>
</tr>
<tr>
<td>Triangular Lattice</td>
<td>22</td>
<td>440</td>
</tr>
</tbody>
</table>
6. Conclusion

In this paper, we studied the problem of sensor nodes deployment in three-dimensional WSNs and proposed a heuristic algorithm that provides a near optimal deployment that ensures full-coverage and connectivity at minimal cost. Our method considers the obstacles residing in the 3D target space and computes the semi-optimal sensor node deployment taking into account their influence on the coverage and the connectivity among sensor nodes. Based on this method, we implemented a WSN simulator using the UbiREAL simulator and evaluated the performance of the algorithm through visualization.

As part of the future work, we will evaluate the proposed method under more practical conditions. This will allow us to improve the deployment algorithm to maintain better performances while minimizing the cost. We will complete our studies by investigating the case where \( r_s > \frac{1}{3} r_c \). In parallel, we will conduct experiments on a WSN testbed and extend the WSN simulator to simulate realistic sensing/communication ranges depending on types of sensors/wireless communication standards and actual influence by obstacles.

References