Development of a Simulator for Stochastic Deployment of Wireless Sensor Networks

Carlos E. Otero
Florida Institute of Technology/Electrical and Computer Engineering, Melbourne, FL, USA
Email: cotero@fit.edu

Ivica Kostanic and Luis D. Otero
Florida Institute of Technology/Electrical and Computer Engineering, Melbourne, FL, USA
Email: {kostanic, lotero}@fit.edu

Abstract—This paper presents an open framework and detailed procedures for creating simulators that address application-specific deployment of Wireless Sensor Network (WSN). The presented framework and developed models are combined in a deployment simulator. The simulator assists decision-makers in selecting among different WSN deployment alternatives. The outlined software framework serves as a blueprint for creating deployment simulators that take into account application specific factors and may be used for optimization of WSN stochastic deployments.

Index Terms—Wireless Sensor Networks, Stochastic Deployment, RF Propagation, Connectivity, Coverage, Simulation

I. INTRODUCTION

Recent advances in micro electro-mechanical systems (MEMS) have led to the development of tiny low-power devices that are capable of sensing the world and communicating with each other. Such devices may be deployed in vast numbers over large geographical areas to form wireless sensor networks (WSN). WSN provide the means for autonomous monitoring of physical events in areas where human presence is not desirable or even possible. Therefore, they are expected to facilitate many existing applications and bring into existence entirely new ones. A few proposed applications of WSN include [1]:

- Disaster Relief
- Environmental Control
- Military Applications
- Border Security

In each application, the sensor nodes are deployed over the area of interest. They are usually tasked with sensing the environment and communicating with each other in a multi-hop fashion. The sensing information is transmitted from the sensors back to a base station, also known as the information sink [2]. From the sink, the information is collected and typically relayed to a central location, where it is processed and analyzed.

For the most part, WSN are highly application-dependent. This means that details such as node design, form-factor, processing algorithms, network protocols, network topology, and deployment scheme are customized for the proposed application. Among these, the network deployment scheme is considered extremely important, since it directly influences parameters such as complexity, connectivity, cost, and lifetime.

Generally speaking, small-scale WSN applications typically allow deterministic deployment schemes and do not experience problems such as lack of connectivity or inadequate area coverage. In these scenarios, the deployment scheme consists of precise installations performed in such way that the connectivity is guaranteed and the coverage is optimal. This reduces network complexity by eliminating the need for complex protocols or additional processing that manages the network and ensures its operation. Deterministic deployment schemes are optimal; however, they are impractical and sometimes impossible for large-scale WSN applications. In large-scale applications, WSN may be dropped from a plane, delivered in an artillery shell, rocket or missile, or catapulted from a shipboard [2]. In these cases, the deployed WSN has the utmost challenge of guaranteeing connectivity and proper area coverage [3]. This requires additional complex protocols that ensure efficient network operation, maximize network lifetime, and decrease frequency of re-deployment are required.

Delivery of large-scale WSN presents a major problem in the application of WSN technology, since they rely mainly on stochastic deployment schemes. Achieving acceptable network efficiency (e.g., connectivity, area coverage, lifetime, cost, etc.) in both deterministic and stochastic deployment schemes is formally referred to as the deployment problem [1].

The deployment problem has been the topic of much research work; however, the majority of the work concentrates on carefully positioning of nodes to meet application requirements [4, 5, 6, 7, 8, and 9]. Insufficient work has been done on the analysis of random deployment of WSN. Furthermore, most of the work provides solutions that optimize one or two parameters at the expense of other network parameters. These approaches fail to provide decision-makers with...
holistic views of deployment strategies that supports them in the decision-making process and allows them to make decisions based on application-specific parameters.

Accurate modeling of specific WSN deployment environment is necessary before any deployment decision can be made. Since WSN are application-specific, it is impractical to expect that the same solution can be used to address deployment in all environments [10-12]. Therefore, simulation engines must be developed and customized in detail to model specific environmental and deployment issues (e.g., RF propagation, terrain) affecting network operations.

This paper presents an open framework and detailed procedure for creating simulators that address application-specific deployments. The presented framework and models are combined in a deployment simulator, which serves as blueprint for creating future deployment simulators that may be used as main decision-making tools in stochastic deployment of WSN. The presented framework is focused on the WSN deployment and addresses specific issues of connectivity, coverage, cost and lifetime. Implementation of complex operations (e.g., protocols) required for network operation after deployment are deemed to be out of the simulation scope.

The remainder of the paper is organized as follow. Section II provides a summary of related work in stochastic deployment techniques. Section III describes the process for creating simulation engines for WSN deployments. Specifically, it covers in details the aspects related to WSN deployments and provides the methods for creating a deployment simulation engine. Section IV provides results and analysis for supporting deployment decisions. Finally, Section V provides conclusions and identifies future research.

II. BACKGROUND WORK

Wireless Sensor Networks are composed of small, custom-designed computers equipped with sensors and radio technology. These custom-designed computers are commonly referred as wireless sensor nodes. Wireless sensor nodes come in different size, shape, and form. However, at the core, they share many similar features. Firstly, they provide cheap, memory constrained computing with sensing and radio technology. Secondly, they are small and lightweight. Thirdly, they are battery-operated and consume small amounts of energy. These features allow for WSN operation and deployment in environments that are not suitable for general purpose computers.

A. WSN Applications

WSN are frequently used in military applications. The military community is interested in deploying WSN to provide battlefield surveillance, reconnaissance, targeting, and battle damage assessment [2]. In these applications, WSN are deployed on-demand to monitor critical terrains and obtain timely information about the enemy activities. In addition, military WSN can be used to replace land-mine systems. With current land-mine systems, anyone moving through the area, friendly or not, is affected. Moreover, long after conflicts are over, these land-mines are still active and deadly [13]. According to a UNICEF report, over the last 30 years, land mines have killed or harmed more than one million people, many whom are children [13]. These deadly systems can be replaced by deploying thousands of wireless sensor nodes equipped with a magnetometer, a vibration sensor, and a GPS receiver [13].

Other work proposes deploying WSN for nuclear, biological, and chemical attack detection [2]; disaster relief operations, forest fire detection, and flood detection [14, 15]. In all of these applications, accurate prediction of network operation upon deployment is an essential component.

B. Deployment Challenge

WSN offer unique systems for creating communications infrastructures on-demand. However, their use is dependent on the effective deployment of these systems to areas of interest. There are several challenging issues involved in the deployment of WSN, mostly due to their small size and large number of nodes required to establish proper operation. For the most part, initial deployment must (1) reduce installation cost, (2) promote self-organization, and (3) eliminate the need for pre-organization [2]. Reducing installation cost is attained by minimizing the number of nodes deployed in a given WSN. However, promoting self-organization and eliminating the need for pre-organization may require high number of nodes to ensure proper connectivity.

Higher density systems provide greater number of independent measurements and ability to put nodes to sleep for long periods to extend network lifetime [16]. However, higher density of nodes increases network cost. An alternative solution consists of deploying fewer nodes, while transmitting over longer distances to ensure network connectivity. This approach minimizes cost at the expense of network lifetime, since transmission power increases with distance. This poses one last requirement for initial deployment; that is, initial deployment must (4) reduce radio transmission power to increase network lifetime. Collectively, network cost, connectivity, area coverage, and lifetime define the WSN network efficiency and are at the forefront of research in stochastic deployments of WSN.

B. Stochastic Deployments

Deterministic deployment of WSN typically results in optimal efficiency; however, due to node size and node density required to provide appropriate network coverage in large geographical areas, careful positioning of nodes becomes impractical. Furthermore, several applications of WSN are expected to operate in hostile environments [17]; this makes deterministic deployment in some cases impossible. Consequently, stochastic deployments become the only feasible alternative [1].

In [19], the authors focus on determining the number of randomly deployed nodes required to carry out target detection in a region of interest. In their work, they identify path exposure as a network efficiency metric and define it as the measure of the likelihood of detecting a
target traversing the region using a given path. The
decision tradeoffs in the study lie between path exposure
(i.e., area coverage) and deployment cost. Similarly, in
[3, 20], the authors study ways to maximize area
coverage in randomly deployed wireless sensor networks.

In [21], the authors present interesting contributions to
the deployment problem by attempting to maximize
coverage and connectivity in randomly deployed WSN.
They study random deployment using three different
statistical distributions: simple diffusion, constant
placement, and R-random placement. The simple
diffusion distribution models sensor nodes deployed from
an air vehicle. The constant placement distribution
models sensor networks with constant node density and
random positioning within the area of interest. Finally,
the R-random distribution models deployments where
nodes are uniformly scattered in terms of the radius and
angular direction from the center, which coincides with
the sink. Their study involved 250 deployed nodes, each
with a fixed sensing range of 60 meters, and a radio
transmission range fixed at 100 meters. Using these fixed
parameters, they maximize coverage and connectivity
using the three different deployment distributions.

In [22, 23], the authors point out the lack of research
towards the WSN deployment problem and state that
“While WSN design, architecture, protocols and
performance have been extensively studied, only a few
research efforts have studied the device deployment
problem”. Furthermore, the authors point out flaws in
recent publications by stating that “most of these works
tackle the deployment problem only from a perspective of
coverage and/or connectivity. The significance of
deployment on lifetime is mostly overlooked”. Their
work proposes three deployment strategies, namely
connectivity-oriented deployments, lifetime-oriented
deployments, and hybrid deployment; which addresses
the concerns of both connectivity and lifetime in sensor
network deployments. However, their work uses fixed
network parameters, which limits the results to their
particular research environment.

In [24, 25, 26], attempts to decrease node density (i.e.,
cost) have been made in random deployments. For
example, in [24], significant reduction of network cost
occurs by devising application-specific deployment
architecture for sensor networks in perimeter security. It
is the most up to date work found in the literature.

Perhaps the most complete approach to the WSN
deployment problem is proposed in [27]. In their work, a
simulator-platform is developed to support the ad hoc
deployment of WSN. Their methodology begins with a
stepwise deployment of sensor nodes in a simulation
environment. Once the first deployment is complete, area
coverage and connectivity is computed; if desired
coverage and connectivity is not attained, a second
iteration deployment occurs until satisfactory coverage
and connectivity is attained. An important factor in
computing connectivity is radio propagation. Their work
uses the free-space radio model, which assumes clear,
unobstructed line of sight between sender and receiver.
For WSN, this assumption can lead to misleading results.

Most of the work done so far in stochastic deployment
of WSN use optimization approaches to make
deployment decisions that maximizes partial network
efficiency metrics. Not enough work has been done on
providing holistic views to allow decision-makers to
consider all variables present in WSN deployments.
Furthermore, important methods, such as RF propagation
models, are either omitted or inadequate to model
different environments. In this paper, we extend the work
presented in [27] by providing detailed procedures and
models for creating customized simulators for studying
stochastic deployments of WSN.

III. WSN DEPLOYMENT SIMULATION

This section presents the simulation framework used
for development of the Simulator for Wireless Sensor
Networks Research (SIMWISER). SIMWISER is a
Microsoft’s Windows application developed to study the
WSN deployment problem. The simulation framework
used by SIMWISER provides placeholders that support
multiple deployment distributions, different levels of
terrains obstructions, multiple propagation models,
variable radio range, variable sensing range, and easy
customization for data collection and analysis. These
capabilities are designed at the core of the framework
through three different domains, namely the Node
Domain, Manager Domain, and Air Interface Domain.
The Node Domain serves as compartment for all code
required to simulate the wireless sensor node. The
Manger Domain is where modeling of specific
phenomena encountered in the WSN deployment
problem is performed. Finally, the Air Interface Domain
provides a common repository for computations
involving deployed nodes in the WSN, such as
connectivity between nodes and area coverage. These
domains work together to provide a generalized approach
for evaluating WSN efficiency in different deployments.
An overview of SIMWISER’s architectural framework is
presented in Figure 1.

SIMWISER’s design and capabilities provide an
efficient platform for studying the WSN deployment
problem. First, by segmenting SIMWISER’s code,
changes to specific models, techniques, and algorithms
can be made easily without affecting the rest of the code.
Second, unlike the popular ns2 simulator [10],
SIMWISER does not require users to have knowledge of
high-level languages, specific programming
environments, or scripting languages to engage
simulations. This is because the core simulation engine
provides a configurable interface that allows users to
customize and create unattended simulations with user-
desired parameters. Finally, SIMWISER provides built-in
capabilities for data management and filtering of large
volumes of data collected from the simulations. The
output results are easily exported in formats that are easy
to analyze with external tools such as Microsoft’s Excel,
R [28], and Minitab [29].
A. Node Domain

The Node domain consists of objects that model the wireless sensor node. Specifically, the Node Domain is composed of the Graphic Wireless Node (GWN) and the Wireless Node (WN). The GWN serves as interface for users to monitor and control the WN; therefore, there is a one to one relationship between the GWN and WN objects. This means that deployed WN objects in simulations have a uniquely associated GWN object responsible for representing its state, capabilities, and behavior in graphical format. This process is illustrated in Figure 2.

As seen, the user interfaces with the GWN to turn on sensing capabilities; the GWN request from the WN the sensing range; the WN provides a response containing the defined sensing range for the ongoing simulation; and finally, the GWN presents to the user the graphical display of the sensing range relative to the node’s position. The WN is where actual node capabilities are implemented, such as radio range, sensor range, transmission frequency, and receiver’s sensitivity among others. Typical capabilities are supported by the framework’s general WN model, which include radio ranges from 50 to 100 meters [30-32], sensor ranges from 10 to 60 meters [33], receiver sensitivity from -90 to -100 dBm [34-37], and transmission frequency of 2.4 GHz. Although the receiver’s sensitivity values are derived from common commercially available wireless sensor nodes, new values can be computed using:

\[ RxSens = 10 \log_{10}(kTB) + F + C/N \]  

where \( kT \) is the power spectrum density of thermal noise, \( B \) is the bandwidth of the receiver, and \( F \) is the noise figure expressed in dB and \( C/N \) is the required carrier to noise ratio. Once specific WN characteristics have been identified for specific deployment application, the general WN model can be modified easily to reflect real node characteristics without affecting any other part of the framework.

B. Manager Domain

The Manager Domain consists of the Simulation Manager, Deployment Manager, RF Manager, and File Manager. Managers work together to provide simulation capabilities for modeling wireless transmissions, distribution of sensors, managing data collection and formatting. The following sections describe in detail the available managers.

1) Simulation Manager

The Simulation Manager (SM) is responsible for configuration and coordination of experimental deployment scenarios. Specifically, it provides capabilities to configure deployment strategies under varying levels of terrain obstructions. Each strategy consists of specific number of nodes, radio range, and sensor range. In addition, the SM provides capabilities of replication and batch simulation, which provides for automated simulation and data collection of multiple deployment strategies. The SM’s graphical user interface is presented in Figure 3.
The response outputs provided by the SM are connectivity, coverage and transmission power. The SM can operate in the following modes: static mode, dynamic mode, and hybrid mode. When operating in the static mode, the SM uses fixed input parameters and uses replication to compute responses. In the dynamic mode, the SM uses replication while varying all input factors to provide responses for each level of each factor at each level of all other factors. In hybrid mode, the SM operates at fixed level of specific factors while varying the levels of all other factors. This mode is essential in capturing the effects of specific factors on responses. Once the SM is configured, it creates a list of experimental scenarios that is used as an input to the core simulation engine.

2) Deployment Manager

The Deployment Manager (DM) provides a central repository for management of the deployment area and support of statistical distributions for node deployment. As in most of the literature, the deployment area is assumed to be two dimensional and rectangular in shape [1]. The DM provides functionality to configure and modify the rectangular deployment area before or during ongoing simulations. In addition, the DM couples the deployment distribution from the rest of the code. This way, new distributions can be added without affecting the simulation environment. The current deployment distribution supported by the DM is uniform [0-1], which is a popular distribution for WSN research [1]. However, other distributions found in [21] can be added with a little effort. The DM’s graphical user interface is presented in Figure 4 and a sample deployment is shown in Figure 5.

3) The RF Manager

The RF Manager is responsible for simulation of the radio frequency (RF) signal propagation. RF signals are subject to physical phenomena that distort the original transmitted waveform and introduce uncertainty at the receiver side [1]. The three main phenomena affecting RF propagation are reflection, diffraction, and scattering [38]. These mechanisms introduce a distance-dependent, frequency selective, and time-variable loss of the transmitted signal. This loss is referred to as path-loss. Path loss models provide estimates of signal loss as function of distance between transmitter and receiver. The mechanisms governing path-loss are dependent on deployment area; therefore a single unique model cannot describe radio propagation for a variety of environments [39]. The main path-loss models associated with WSN research are: free space, two ray, and log-distance [10].

The free-space path loss model predicts received signal strength as function of distance when the transmitter and receiver have clear and unobstructed line-of-sight path [38]. This model does not include effects of terrain, obstacles or fading. Furthermore, in most realistic environments, the transmitted signal reaches receivers through different paths. Therefore the free-space model is not well suited for most WSN applications.

The two-ray model is a useful propagation model that considers both direct path and ground reflected propagation path between transmitter and receiver [38]. However, similar to the free space model, the two ray model does not take into account environmental obstructions affecting signal strength. Depending on the environment, the received signal strength for the same transmission distance will be different [39]. This variation due to location is referred to as the log-normal shadowing and may be modeled using the log-distance propagation model [38]. The log-distance model indicates that the average received signal power decreases linearly as a function of log of distance. Furthermore, the model accounts for environmental obstructions present between transmitter and receiver. For example, the measured received signal strength between transmitter and receiver may vary greatly in environments with higher obstruction characteristics. This variation due to log-normal shadowing can be modeled using a zero-mean Gaussian distributed random variable with standard deviation $\sigma$ [38]. The log-normal shadowing model is supported by the RF Manager’s general propagation model. However, more accurate models can be added to the RF manager once proper characterization of the deployment area has been made. The log-normal shadowing model is shown in (2) [38].

$$L_p = L_0 + 10\alpha \log_{10}(d) + X \quad (2)$$

where $\alpha$ is referred to as the path loss exponent, $d$ is the distance, and $X$ is a zero-mean Gaussian random variable with standard deviation $\sigma$. Typical values for $\alpha$ in terrestrial environment are 3-5. Typical values for $\sigma$ are shown in Table 2 [10].
The allowable path loss ($L_p$) for specific transmitter-receiver separation distance is a function of the power transmitted ($P_{tx}$) and the receiver’s sensitivity ($P_{rx}$). This path loss may be calculated as

$$L_p = 10 \log(P_{tx}) - 10 \log(P_{rx}) \quad (3)$$

Therefore, the log distance model from (2) may be used to compute the required power in accordance with

$$10 \log(P_{tx}) - 10 \log(P_{rx}) = L_0 + 10 \alpha \log(d) + X,$$

$$10 \log(P_{tx}) = 10 \log(P_{rx}) + L_0 + 10 \alpha \log(d) + X \quad (4)$$

where, $L_0$ is the path loss at the reference distance (e.g., 1 meter) and may be computed as function of transmission frequency (in MHz) and distance (in km), as

$$L_0 = 32.44 + 20 \log(f) + 20 \log(d) \quad (5)$$

Finally, a two-step process is used to compute the log-normal shadowing effect ($X$). First, the Box-Muller transformation method [40] is used to transforms two independent and uniformly distributed random variables on the interval [0, 1] into two standardized normal random variates, as seen in (6) and (7).

$$y_1 = \left[-2 \log(x_1)\right]^{1/2} \sin(2\pi x_2) \quad (6)$$

$$y_2 = \left[-2 \log(x_1)\right]^{1/2} \cos(2\pi x_2) \quad (7)$$

Once in standard form, de-standardization is accomplished to obtain a normal random variate $X$ with mean $\mu = 0$ and standard deviation $\sigma$, as seen in (8) [41].

$$X_i = \mu + \sigma y_i = \sigma y_i \quad (8)$$

4) The File Manager

The File Manager is responsible for collecting, formatting, and providing output of simulated data. There are several configurations available for data output, namely, raw output, summarized output, and customized output. Raw output provides the data as they were collected from the simulation experiment. Summarized output analyses the data, identifies equal deployment simulations, computes average values, and provides these averages as outputs. Customized output provides means for configuring specific simulation outputs. All of the File Manager’s outputs are compatible with the Microsoft’s Excel format, which allows data analysis with common statistical packages such as Excel and Minitab.

C. Air Interface Domain

The Air Interface Domain (AID) is the central repository where deployed nodes are managed. Specifically, the AID serves as central compartment to keep track of each deployed node, connectivity between nodes, and coverage achieved by each node. The two main components of the AID are the Node Database data structure and the Air Interface object. In each simulation, nodes are deployed using the Deployment Manager and stored in the Node Database structure. Once in the database structure, the Air Interface object computes both connectivity among all deployed nodes and area coverage. The following sections describe these processes in detail.

1) Node Connectivity

Node connectivity is a measure of how many nodes are directly or indirectly connected in a particular deployment. The Air Interface object computes connectivity using the following steps.

a) Create connectivity matrix (CMatrix)

The CMatrix is an $n \times n$ adjacency matrix that keeps track of connectivity throughout the WSN. Initially, for each row $i$ and column $j$, all elements $a_{ij} = 1$, and all elements $a_{ii} = 0$, as seen in (9). This initial configuration indicates zero connectivity.

$$CMatrix = \begin{bmatrix} 0 & 1 & \ldots & n \\ 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ n & 0 & \ldots & 1 \end{bmatrix} \quad (9)$$

b) Determine CMatrix values

Once the CMatrix data structure is created, the Air Interface object polls the Node Database for all deployed nodes, determines direct and indirect connections, and updates the connectivity matrix accordingly. Direct connections are determined by computing the Euclidean distance between two nodes and comparing it with the transmission distance, as determined by the RF propagation model. In two-dimensional planes, the distance between two points $(x_i, y_i)$ and $(x_j, y_j)$ is the length of the path connecting them and can be obtained from (10) [42].

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (10)$$

Therefore, assuming transmission distance ($d_{tx}$) node $i$ with position $(x_i, y_i)$, and node $j$ with position $(x_j, y_j)$, the $CMatrix$ values for single-hop case can be computed using the algorithm in Figure 6.
For the multi-hop case, the single-hop communication algorithm is used recursively to determine indirect connections between nodes.

c) Compute Connectivity

Once the CMatrix values are determined, the overall WSN connectivity is determined by traversing the CMatrix and computing the ratio of connected nodes and total number of deployed nodes, as

\[
\text{Connectivity} = \left( \frac{\sum_{i=0}^{n} \sum_{j=0}^{n} C_{ij}}{n^2 - n} \right) \times 100
\]  

(11)

2) Area Coverage

Area coverage is defined as the ratio of points sensed by the WSN in the deployment area to the total number of points in the deployment area. This metric explains how much of the area of interest is covered after deployment of the WSN. The Air Interface domain uses the Boolean Sensing Model [1] to determine coverage. In the Boolean Sensing Model, all sensors have a common sensing range; events within this sensing range are detected reliably, and events outside this range are not detected at all. In addition, as in most sensing models used in the literature [43-45], the sensor range is assumed omnidirectional with no random variations. The Boolean sensor model used by the Air Interface Domain is shown in (12) [1].

\[
S(p, q) = \begin{cases} 
\alpha : \|p - q\|_2 \leq r \\
0 : \text{otherwise} 
\end{cases}
\]

(12)

where \(p\) is the position of the sensor node, \(r\) is the sensing range, \(q\) is the position of event of interest, \(\|\|_2\) is the Euclidean distance between two points, and \(\alpha\) is the sensor output. Using the sensor model, the total number of covered points \(c\) in the deployment area is computed using

\[
c = \sum_{p \in P} f(p), \text{ where } f(p) = \begin{cases} 
1 : p \in C \\
0 : \text{otherwise} 
\end{cases}
\]

(13)

where \(p\) is a point in the deployment area, \(P\) is the set of all points in the deployment area, and \(C\) is the set of all points covered by the WSN. Finally, the formula for computing the percentage of area coverage is given as

\[
\text{AreaCoverage} = \sum_{p \in P} c \times 100 \tag{14}
\]

IV. DEPLOYMENT ANALYSIS

This section provides results obtained by simulation and in support of the decision-making process for uniformly distributed WSN deployments. The scenario assumes a typical rectangular deployment area measuring 500 m x 500 m, 50 to 100 nodes available for deployment with onboard radio capable of transmitting between 40 to 100 meters and sensors capable of covering between 30 to 60 meters. In addition, signal degradation due to terrain obstacles can vary from low to high. Figures 3 and 4 display SIMWISER’s configuration for this case study. The following sections provide detailed results.

A. Connectivity

Using the presented framework, connectivity is computed and results are presented in Figure 7.

As seen, the simulation results provide insight into the dynamics involved in the deployment problem. Specifically, it provides a full picture of the effects of varying radio range and number of nodes on overall connectivity. More importantly, the results suggest that connectivity for deployment strategies using radio ranges of 40 to 50 meters do not appear significantly different. Similarly, deployment strategies containing 70 to 100 nodes and varying radio ranges of 80 to 100 meters show non-significant differences in connectivity. In these scenarios, decision-makers can reduce network cost and increase lifetime by using deployment strategies containing fewer number of nodes and decreased radio range.

B. Coverage and Tx Power

The results obtained from simulation for area coverage and transmission power are presented in Figures 8 and 9.
network efficiency. The effects of number of nodes, radio range, sensor range, and terrain obstruction on overall connectivity and coverage profile, this information provides decision-makers with full characterization of the deployment scenario have been presented. The results obtained from simulation suggest that several deployment strategies vary slightly from one another in terms of connectivity and coverage. These variations require proper analysis to determine the statistical effects that number of nodes, radio range, and sensor range have on connectivity and coverage. Future work is on the way to add placeholders to the framework for relevant statistical tests. These tests would enable decision makers to attach a statistical significance to the results that are produced by the simulator.

V. CONCLUSIONS AND FUTURE WORK

An open framework for using simulation as the main decision-making tool for stochastic deployment of WSN has been presented. The framework serves as blueprint for decision-makers to create and use application-specific simulations with application-specific models that are representative of the desired deployment scenario. In addition, analysis and results of specific WSN stochastic deployment scenario have been presented. The results show how simulation provides a full view of deployments and how different levels of deployment parameters affect the overall deployment efficiency.

Although the presented framework serves well to provide guidance into building deployment simulators, several avenues for improvement have been identified. The most important ones are listed as follows.

A. Deployment distributions

Perhaps the most challenging problem present in stochastic deployments of WSN is the accurate characterization of deployment distribution. Most of the work in the literature assumes a uniform distribution [1]. However, others have presented interesting distributions [21] that may occur as results of different deployment methods. Deployment simulators should include a comprehensive list of deployment distributions and provide decision-makers with different views of network efficiency obtained using different deployment methods.

B. RF Models

Accurate RF propagation modeling is essential when studying the WSN deployment problem to prevent misleading conclusion from simulation results. An extensive list of RF models addressing various application scenarios is presented in [38, 39]. Deployment simulators should include access to these models to improve accuracy of results.

C. Statistical Analysis

The results obtained from simulation suggest that several deployment strategies vary slightly from one another in terms of connectivity and coverage. These variations require proper analysis to determine the statistical effects that number of nodes, radio range, and sensor range have on connectivity and coverage. Future work is on the way to add placeholders to the framework for relevant statistical tests. These tests would enable decision makers to attach a statistical significance to the results that are produced by the simulator.

REFERENCES
