

Breach Path Reliability for Directional Sensor Networks

Mohammed Elmorsy
Department of Computing Science
University of Alberta
Edmonton, T6G 2E8, Canada
E-Mail: elmorsy@ualberta.ca

Ehab S. Elmallah
Department of Computing Science
University of Alberta
Edmonton, T6G 2E8, Canada
E-Mail: elmallah@ualberta.ca

Abstract—Wireless Sensor Networks (WSNs) equipped with directional communication and sensing devices provide a high level of tunability needed in optimizing their performance in critical applications. Such devices and nodes, however, remain prone to failure when operating in the field. In this paper we formalize a problem, called directional breach path detection reliability (DIR-BPDREL), that quantifies the ability of such networks to jointly detect and report unauthorized traversal through a network when communication and sensing devices fail independently of each other. We adopt a framework for deriving lower and upper bounds on exact reliability solutions, and develop efficient algorithms for optimizing the computations using pathset and cutset structures of the given network. The algorithms process separate communication and sensing graphs to ensure joint detection and reporting of intrusion events from multiple possible entry-exit sides. The obtained numerical results give insight into the effect of various design parameters on network wide performance.

I. INTRODUCTION

Recent advances in designing wireless sensor nodes have enabled the production of low cost modules capable of achieving spatial communication and sensing selectivity through the use of directional communication and sensing devices. The use of such technologies enable nodes to tune their communication and sensing ranges and directions while incurring acceptable levels of power consumption. Such technologies push the frontier of using wireless sensor networks (WSNs) in critical areas such as area surveillance applications (see, e.g., [1], [2]). Nodes, however, remain prone to failure due to operation in harsh environments.

In this paper, we consider a problem on assessing the likelihood that a WSN employing directional communication and sensing devices can successfully detect and report intrusion events across a geographical area guarded by the network. The problem is called *directional breach path detection* (DIR-BPDREL) reliability problem. In the problem, the guarded area is defined by a polygon, where each side is a potential entry or exit side for an intruder wishing to cross the area. Sensor nodes in the network fail randomly with known probabilities.

For a brief literature review of some recent related work, we recall that directional WSNs are being used for both target detection and target tracking, especially for camera sensor networks. Examples of work on target detection include the

work of [3]–[10]. Our work here complements the above work by considering WSNs whose nodes are subject to random failure.

Results closely related to our work here include the work of [11]–[13] that investigates a restricted version of the DIR-BPDREL problem, denoted BPDREL. Nodes in the BPDREL problem are assumed to be omnidirectional. In addition, the restricted problem models the given WSN using a single graph G whose links are required to satisfy joint communication and sensing requirements. Moreover, the restricted problem utilizes a 2-state node reliability model where each node can either operate or fail.

In contrast, the more general DIR-BPDREL problem is investigated by decoupling the communication graph G_{com} from the sensing graph G_{sense} , so as to harness the flexibility and power offered by directional devices. Additionally, nodes in our current problem utilizes a 3-state node reliability model that separates communication failure from node failure.

The work in [11], mentions that the BPDREL problem is #P-hard. The work presents a dynamic programming algorithm for computing exact solution of the problem on grid networks. The algorithm runs in polynomial time on $W \times L$ grids, $W \leq L$, for networks with constant width W .

In [12], the dynamic program in [11] is extended to handle networks of arbitrary topologies with more flexibility in placing the sink node. The used methodology involves embedding a subgraph of the given WSN into a *block* grid network. The embedding, however, may result in omitting some links from the original network. Thus, the obtained results serve as lower bounds on the exact solutions.

In our previous work [13], we have presented a framework for computing lower and upper bounds on the exact solution of the BPDREL problem. In this paper, we extend the framework used in [13] to the more complex DIR-BPDREL problem.

Our main contribution in the paper is the formulation of DIR-BPDREL problem where the communication graph is decoupled from the sensing graph, and the adoption of a 3-state node reliability model for increased accuracy. We develop a unified framework for computing lower and upper bounds for the generalized problem at hand.

II. SYSTEM MODEL

In this section, we introduce the needed concepts and notation to explain our bounding approach and algorithms. Restricted versions of these concepts appear in [11] and [13].

A. Network and Directionality Model

Throughout the paper, a WSN is modelled by an undirected communication graph $G_{com} = (V \cup \{s\}, E_{com})$, and a sensing graph $G_{sense} = (V \cup \{s\}, E_{sense})$ on a set V of directional sensor nodes, and a distinguished sink node s . The adoption of two graphs in the model generalizes the single graph model used previously in [13].

In G_{com} , a link (x, y) exists if both nodes can reach each other. In G_{sense} , a link (x, y) exists if every point on the line segment (x, y) can be sensed by node x and/or node y . This latter property is needed to guarantee detection of any crossing of the line segment (x, y) .

The exact links in G_{com} and G_{sense} are determined by directionality parameters, denoted $DIR_{com}(x)$ and $DIR_{sense}(x)$, respectively, associated with each node x . In each case, directionality is specified by a triple, denoted $(r, \theta_{mid} \pm \alpha)$, where

- r is a transmission/reception radius (either communication or sensing),
- $\theta_{mid} - \alpha$ specifies a counterclockwise (CCW) angle between two rays emanating from x ; a horizontal ray that extends to the right, and a ray that defines the start of the boundary of the transmission/reception region, and
- $\theta_{mid} + \alpha$ specifies the CCW angle between the horizontal ray, and a second ray that defines the end of the boundary of the transmission/reception region.

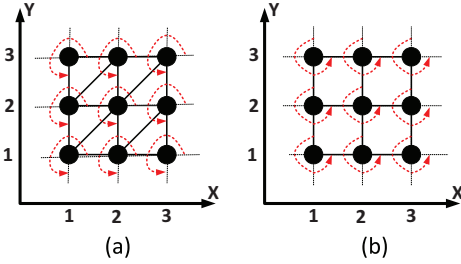


Fig. 1: Example of directional parameters

Example. Fig. 1(a) illustrates G_{com} of a subnetwork composed of 3x3 grid nodes with one unit of horizontal (or vertical) spacing. When $DIR_{com} = (1.7, 135^\circ \pm 135^\circ)$, the transmission/reception region for each node is bounded by two rays emanating from the node with CCW angles $(\theta_a = 0^\circ, \theta_b = 270^\circ)$, as shown by the dashed arrows. A link (x, y) exists in G_{com} if $dist(x, y) \leq 1.7$ unit.

Fig. 1(b) illustrates G_{sense} of a subnetwork on the same set of nodes. When $DIR_{sense} = (0.5, 225^\circ \pm 135^\circ)$, the sensing region for each node is bounded by two rays emanating from the node with CCW angles $(\theta_a = 90^\circ, \theta_b = 360^\circ)$, as shown by the dashed arrows. A link (x, y) exists in G_{sense} if each point on the line segment (x, y) is covered by x and/or y .

Here, since $r = 0.5$, both end nodes of a link are needed to cover the link. ■

The above directionality model allows for achieving spatial selectivity while controlling the consumed power through suitable setting of the transmission/reception radius r as a function of the offset angle α . For example, in Section VI, we experiment with setting $r = \frac{360^\circ}{2\alpha}$ to capture the above effect.

B. Breach Path Model

We fix an embedding of G_{sense} in the plane so that the network's perimeter corresponds to a polygon with well-defined set of perimeter nodes, denoted X , and polygon links, called *sides*. An intruder (breach) path crossing G_{sense} is associated with a pair (d_i^{in}, d_i^{out}) of entry-exit sides on the polygon. Entry-exit sides can be exchanged, so a breach path is undirected. Without loss of generality, we assume that each breach path crosses (or touches) only two perimeter sides of the network.

Our problem deals with a given set of entry-exit pairs of sides $D = \{(d_i^{in}, d_i^{out}) : i = 1, 2, \dots\}$, $d_i^{in} \neq d_i^{out}$. A (d_i^{in}, d_i^{out}) -attack is any breach path that uses the specified sides. A D -attack is any breach path that uses any pair of sides in D .

C. Reliability Model

For any non-sink node x , we adopt a model with independent operation probabilities for the communication and the sensing devices, denoted $p_{com}(x)$, and $p_{sense}(x)$, respectively. The sink is assumed to be perfectly reliable. Thus, each non-sink node x can be in one of 3 possible states with the following probabilities: $p_{cs}(x) = p_{com}(x) \times p_{sense}(x)$: the probability that x can perform both communication and sensing, $p_{cns}(x) = p_{com}(x) \times (1 - p_{sense}(x))$: the probability that x can communicate but not sense, and $p_{fail}(x) = 1 - p_{cs}(x) - p_{cns}(x)$: the probability that x can not communicate with any other node. We assume that the network utilizes adaptive routing that can utilize any existing operational path.

A random failure event leaves a network on n nodes in one of $O(3^{n-1})$ possible states (recall that the sink is reliable). Each state $T = (T_{cs}, T_{cns})$ is defined by a subset T_{cs} of nodes that can communicate and sense, and another disjoint subset T_{cns} of nodes that can communicate but not sense. Each of the remaining nodes $T_{fail} = T \setminus \{T_{cs} \cup T_{cns}\}$ is failed. The probability of being in state T , denoted $\Pr(T)$, is the product of the p_{cs} , p_{cns} , and p_{fail} probabilities of nodes in T_{cs} , T_{cns} , and T_{fail} , respectively.

A state T is *operating* if it guarantees the joint detection and reporting to the sink of any D -attack. The reliability of the network is the sum $\sum \Pr(T)$ over all operating network states T in (G_{com}, G_{sense}) . We now define our problem:

Definition (the DIR-BPDREL problem): Let (G_{com}, G_{sense}) be a WSN model where each node x has operating probabilities $p_{com}(x)$ and $p_{sense}(x)$, and directional parameters $DIR_{com}(x)$ and $DIR_{sense}(x)$. In addition, let

$D = \{(d_i^{in}, d_i^{out}) : i = 1, 2, \dots\}$, $d_i^{in} \neq d_i^{out}$, be a set of entry-exit nodes on the perimeter polygon of an embedding of G_{sense} . Find the probability $Rel(G_{com}, G_{sense}, p, D)$ (or, $Rel(G_{com}, G_{sense})$ for short) that the network is in a state that ensures joint detection and reporting of any D -attack. ■

In [11], it is mentioned that the BPDREL problem is #P-hard on grid networks by a reduction from the 2-terminal network reliability problem (see, e.g., [14]). Consequently, the more general DIR-BPDREL problem is #P-hard.

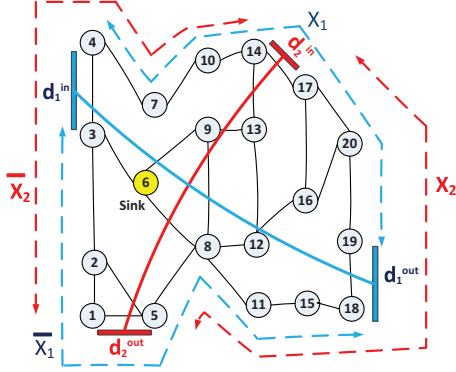


Fig. 2: An example G_{sense} of a WSN

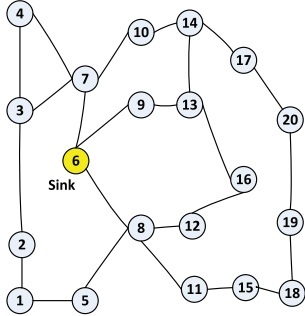


Fig. 3: An example G_{com} of a WSN

III. PROTECTION INTERVALS, PATHSETS, AND CUTSETS

This section extends the concepts of protection intervals, network configurations, pathsets and cutsets discussed, e.g., in [13] to our current problem. The concepts are needed to present our bounding approach.

A. Protection Intervals

Let $(G_{com}, G_{sense}, p, D)$ be an instance of the DIR-BPDREL problem, $X \subseteq V$ be the set of perimeter nodes in G_{sense} , and $(d^{in}, d^{out}) \in D$ be an entry-exit pair of sides. Similar to the development in [13], the *protection intervals* of (d^{in}, d^{out}) are obtained by traversing the perimeter nodes X in a clockwise direction, and write X as two node-disjoint sequences:

$$X_1 = (second(d^{in}), x_i, x_{i+1}, \dots, first(d^{out})), \text{ and} \\ X_2 = (second(d^{out}), x_j, x_{j+1}, \dots, first(d^{in})).$$

where $first(d)$ and $second(d)$ denote the two end points of a side d .

Example. In Fig. 2, we have $(d_1^{in} = (3, 4), d_1^{out} = (19, 18))$, and $(d_2^{in} = (14, 17), d_2^{out} = (5, 1))$. The corresponding protection intervals $(X_1, \bar{X}_1)$ and $(X_2, \bar{X}_2)$ are as shown in the figure. ■

We denote by $\mathbf{X}$ the set of pairs of protection intervals associated with the set D of entry-exit pairs of sides. We also note that given a protection interval X , one can immediately deduce the associated entry-exit pair of sides (d^{in}, d^{out}) on the polygon surrounding G_{sense} , and the other associated protection interval $\bar{X}$.

The importance of protection intervals arises since a path in G_{sense} where each node is in the state *cs*, and the path connects some node on X to some node on $\bar{X}$ is both sufficient and necessary to intersect every possible breach path between the corresponding entry-exit sides. Thus, any operating network state T of a given WSN contains at least one such *sensing barrier path* for each pair of protection intervals in $\mathbf{X}$.

Since $\mathbf{X}$ suffices to identify D in any problem instance, we henceforth use the notation $(G_{com}, G_{sense}, p, \mathbf{X})$ to refer to a given instance of the DIR-BPDREL problem.

B. Network Configurations

A network *configuration* C is a partial network state; that is C assigns to some nodes (but not necessarily all) one of the possible states in the set $\{cs, cns, fail\}$. The remaining nodes are *free* in C . We use $C(x)$ to denote the state s_x of node x in C .

We also use the concept of statistical disjointness: two configurations C_1 and C_2 are **s-disjoint** if at least one node that occurs in both configurations is assigned different states in the configurations. Thus, $\Pr(C_1 \cup C_2) = \Pr(C_1) + \Pr(C_2)$.

C. Pathset Structures

Given an instance $(G_{sense}, G_{com}, p, \mathbf{X})$ of the DIR-BPDREL problem, a pathset of the instance is any operating configuration C . A fundamental problem in our approach is to find pathsets that have high occurrence probability $\Pr(C)$ (subject to some additional constraints).

In this section, we identify a key structural property of pathsets that enable us to efficiently find desired pathsets. The property can be stated as follows.

Lemma 1: A configuration C is a pathset of $(G_{sense}, G_{com}, p, \mathbf{X})$ if for each pair of intervals $(X_i, \bar{X}_i) \in \mathbf{X}$ we have

- G_{sense} has a path, denoted $P_{sense}(X_i, \bar{X}_i)$, connecting some node $x \in X_i$ to some node $\bar{x} \in \bar{X}_i$, where each node in the path is assigned the state *cs* in configuration C , and
- each node in $P_{sense}(X_i, \bar{X}_i)$ can reach the sink in G_{com} by a path composed of nodes assigned either state *cs* or *cns* in configuration C . ■

The rationale behind the above lemma is that condition (a) guarantees that each path $P_{sense}(X_i, \bar{X}_i)$ can detect any (d_i^{in}, d_i^{out}) -attack. Moreover, condition (b) guarantees that the detected event can be reported to the sink. Thus, C is a pathset, as required.

Example. In G_{sense} of Fig. 2, the following paths satisfy condition (a) above:

- $P_{sense}(X_1, \bar{X}_1)$: the edge $((3, cs), (4, cs))$
- $P_{sense}(X_2, \bar{X}_2)$: the edge $((14, cs), (17, cs))$

Hence, each path forms a sensing barrier path. In G_{com} of Fig. 3, the following additional node-state pairs guarantee that each node in each of the above two sensing barrier paths can reach the sink at node 6: $\{(7, cns), (10, cns)\}$. Thus, configuration $C = \{(3, cs), (4, cs), (7, cns), (14, cs), (17, cs), (10, cns)\}$ is a pathset. ■

D. Cutset Structures

Given an instance $(G_{sense}, G_{com}, p, \mathbf{X})$ of the DIR-BPDREL problem, a cutset of the instance is a configuration C that can **not** be extended to a pathset even if all free nodes are assigned the cs state. For a given pair $(X, \bar{X}) \in \mathbf{X}$, we also define an $(X, \bar{X})$ -cutset to be a configuration that can not be extended so as to guard against any $(X, \bar{X})$ -attack. Thus, a necessary and sufficient condition for C to be a cutset is that it includes an $(X, \bar{X})$ -cutset for at least one pair of protection intervals $(X, \bar{X}) \in \mathbf{X}$.

Similar to the previous section, a fundamental problem in our approach for computing UBs is to find cutsets that have high occurrence probability. The following observation specifies a structural property that is used to find good quality cutsets.

Lemma 2: A configuration C is an $(X, \bar{X})$ -cutset of $(G_{com}, G_{sense}, p, \mathbf{X})$ if there exist two subsets of nodes, denoted $U_{cns, fail}$ and U_{fail} , such that each node $x \in U_{cns, fail}$ is assigned a state $C(x) \in \{cns, fail\}$, and each node $x \in U_{fail}$ is assigned a state $C(x) = fail$, and any possible path $P_{sense}(X, \bar{X})$ in G_{sense} that connects some node in X to some node in $\bar{X}$ has either

- at least one node in $U_{cns, fail}$, or
- the failed nodes in U_{fail} disconnect at least one node on $P_{sense}(X, \bar{X})$ from the sink node. ■

The rationale behind the lemma is that condition (a) implies that the path $P_{sense}(X, \bar{X})$ is not an operational sensing barrier path since it has at least one non-sensing node. Likewise, condition (b) implies that at least one node on such a path can not report a detected intrusion event.

Example. In G_{sense} of Fig. 2, the two sets $(U_{cns, fail} = \{3, 9, 12, 18\}, U_{fail} = \emptyset)$ form an $(X_1, \bar{X}_1)$ -cutset that forbids the construction of a sensing barrier path $P_{sense}(X_1, \bar{X}_1)$.

In G_{com} of Fig. 3, the two sets $(U_{cns, fail} = \emptyset, U_{fail} = \{7, 8, 9\})$ disconnects the sink, and thus they form an $(X_1, \bar{X}_1)$ -cutset that disconnects the sink from some node on any possible sensing barrier path $P_{sense}(X_1, \bar{X}_1)$, as required by condition (b). ■

IV. OVERVIEW OF THE BOUNDING APPROACH

Our approach obtains lower bounds (LBs) on $Rel(G_{com}, G_{sense})$ by computing a set of pairwise s-disjoint pathsets $\{P_1, P_2, \dots, P_r\}$, and observing that $Rel(G) \geq \sum_{i=1}^r \Pr(P_i)$ where the summation on the RHS constitutes the intended lower bound.

In a parallel vein, upper bounds (UBs) are obtained by computing a set of pairwise s-disjoint cutsets $\{U_1, U_2, \dots, U_r\}$, and observing that $Rel(G) \leq 1 - \sum_{i=1}^r \Pr(U_i)$ where the RHS is the desired upper bound.

The above approach has been shown to be effective in dealing with various network reliability problems. For example, in [15], the authors use the approach in the context of deriving bounds on several connectivity based network reliability problems with 2-state links. The approach is also used in [16] for evaluating the exposure in a probabilistic setting with 3-state nodes, and in [13] for the basic BPDREL problem with 2-state nodes.

We implement the above approach with an iterative algorithm (omitted for space) that performs a user specified number of iterations while maintaining a set of s-disjoint configurations. In each iteration, the algorithm selects an unprocessed configuration C and seeks to extend it either to a pathset (in LB computations), or a cutset (in UB computations). The new extension, denoted C_{new} , is desired to have a high occurrence probability $\Pr(C)$. The extension process is critical to the quality of the obtained bounds. Configurations that can not be extended are ignored. An exact solution is obtained if the algorithm processes all of its generated configurations.

In the next section, we discuss in more detail the extension to a pathset problem. For space consideration, we omit the presentation of an algorithm that handles the extension to a cutset problem.

V. OPTIMAL EXTENSION TO A PATHSET

A core problem in computing LBs in our approach is to extend a configuration C (that is not a cutset) to a pathset (i.e., an operating configuration). Formally the optimal *extension to a pathset* (E2P) problem calls for computing a configuration C_{new} such that $C \cup C_{new}$ is a pathset, and $\Pr(C_{new})$ is as high as possible. The E2P problem is discussed in [13] for the BPDREL problem where we deal with the special case $G_{sense} = G_{com}$.

For the DIR-BPDREL problem, the E2P problem is expected to be NP-complete, and hence we consider in this section developing an efficient heuristic algorithm. Our algorithm computes a solution to any given instance of the E2P problem, if one exists. The computed solution, however, may not be an optimal solution. Function E2P in Fig. 4 outlines the general steps of the algorithm. The function performs its work by calling functions `find_SB` and `connect_SB` described below.

A. Use of Most Probable Paths

To obtain good solutions, function E2P makes frequent use of an algorithm to solve the following problem. Given a weighted undirected graph where each node x has an

Function E2P($G_{sense}, G_{com}, p, \mathbf{X}, C$)
Input: An instance $(G_{sense}, G_{com}, p, \mathbf{X}, C)$ of the E2P problem
Output: If the E2P has a solution, return a configuration C_{new} containing additional nodes used to form a pathset. Else, return an error value.
1. call function find_SB($G_{sense}, G_{com}, p, \mathbf{X}, C$) to identify a set of sensing barrier paths that satisfies condition (a) in Lemma 1. C_{new_sense} is the configuration of the new added nodes in the set of paths.
2. call function connect_SB($G_{com}, p, \mathbf{X}, C, C_{new_sense}$) to identify a set of nodes that satisfies condition (b) in Lemma 1; that is, a set of nodes that connect every node in the set of sensing barrier paths found in step 1 to the sink. C_{new_com} is the configuration of the new added nodes in this set.
3. **return** $C_{new} = C_{new_sense} \cup C_{new_com}$

Fig. 4: Pseudo-code for function E2P

associated weight $\tilde{p}(x)$, and two disjoint subsets of nodes X and Y of G , find the *most probable* path that links some node in X to some node in Y . The weight $\tilde{p}(x)$ may either be zero, indicating that x is to be deleted from G before the computations, or a positive probability value.

An efficient algorithm for solving the problem is readily available by using a standard shortest path algorithm through the following transformation: (1) delete nodes with $\tilde{p}(x) = 0$ from G , (2) associate with each node x a non-negative weight $w(x) = -\log \tilde{p}(x)$ (so, a small $\tilde{p}(x)$ probability maps to a large positive weight $w(x)$), and (3) introduce two new nodes x and y ; connect x (respectively y) to each node in X (respectively, Y). A shortest (x, y) -path in the constructed graph is a most probable path connecting some node in X to some node in Y in the original problem, as required.

The asymptotic running time of the algorithm is the same as the running time of the used shortest path problem, denoted T_{SP} . For the purpose of analyzing our algorithm that solves the E2P problem, we use T_{SP} as a component in our timing analysis.

B. Finding a Sensing Barrier

Function find_SB (Fig. 5) aims at computing a set of sensing barrier paths $\{P_{sense}(X_i, \bar{X}_i) : i = 1, 2, \dots\}$ in G_{sense} that satisfies condition (a) in Lemma 1, whenever possible.

The function returns an error value in C_{new_sense} if no such set exists (no solution is found). On the other hand, if a solution exists then nodes in the solution that are not in C are returned in the new configuration C_{new_sense} . The function aims at maximizing the occurrence probability $\Pr(C_{new})$. To this end, the function performs the following steps.

Step 1 initializes $C_{new_sense} = \emptyset$. Step 2 initializes a probability array $\tilde{p}(x \in G_{com})$ that guides the function so that (a) it never selects a node x that can not reach the sink node in (G_{com}, C) , and (b) it favours the selection of nodes that can jointly be assigned the cs state and reach the sink with high probability. The setting of $\tilde{p}(x)$ is such that higher numerical values indicate higher desirability of selecting node

Function find_SB($G_{sense}, G_{com}, p, \mathbf{X}, C$)
Input: An instance $(G_{sense}, G_{com}, p, \mathbf{X}, C)$ of the E2P problem
Output: If the E2P problem has a set of sensing barrier paths satisfying condition (a) in Lemma 1 then return a configuration C_{new_sense} containing the additional nodes used to form the paths. Else, return an error value.
1. set $C_{new_sense} = \emptyset$
2. use (G_{com}, p, C) to assign to each node x a weight $\tilde{p}(x)$ (high values indicate node importance)
3. **for each** pair $(X_i, \bar{X}_i) \in \mathbf{X}$ {
4. find a most probable $(X_i, \bar{X}_i)$ -path in $(G_{sense}, \tilde{p})$. If no such path exists return an error value in C_{new_sense} . Else, denote the path by $P_{sense}(X_i, \bar{X}_i)$.
5. **for each** (free node x in $P_{sense}(X_i, \bar{X}_i)$) {
6. add (x, cs) to configuration C_{new_sense}
7. set $\tilde{p}(x) = 1$
} }
8. **return** C_{new_sense}

Fig. 5: Pseudo-code for function find_SB

x . More specifically, for each node x , Step 2 initializes $\tilde{p}(x)$ as follows:

- 1) If $C(x)$ (i.e., x 's state in C) is in $\{fail, cns\}$, or if x can not reach the sink in (G_{com}, C) then set $\tilde{p}(x) = 0$. Nodes in this class can not appear in any sensing barrier path (hence, the setting $\tilde{p}(x) = 0$).
- 2) Else, if $C(x) = cs$ then set $\tilde{p}(x) = 1$. Nodes in this class are already part of the input configuration; they can contribute to building the required sensing barrier paths (at no extra cost).
- 3) Else, if $C(x) = free$ then set $\tilde{p}(x) =$ the probability of a most probable path in (G_{com}, C) connecting x to the sink. Nodes in this class are candidates for use in building the required sensing barrier paths. The setting of $\tilde{p}(x)$ gives more suitable nodes higher values.

Steps 3 to 7 iterate over all pairs of protection intervals $(X_i, \bar{X}_i) \in \mathbf{X}$. Each iteration finds a suitable most probable path in $(G_{sense}, \tilde{p})$. Step 7 sets $\tilde{p}(x) = 1$ if x is a free node selected for inclusion in any path.

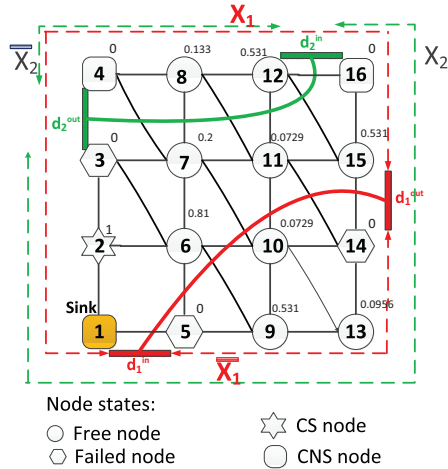
Example. Fig. 6 illustrates a G_{sense} of a WSN. The shape of each node specifies its state in the input configuration $C = \{(2, cs), (3, fail), (4, cns), (5, fail), (14, fail), (16, cns)\}$. Node 1 is the sink node. Each non-sink node x is labelled with the weight $\tilde{p}(x)$, as defined above. Function find_SB finds the following most probable paths (and assigns to the new nodes the following states):

- $P_{sense}(X_1, \bar{X}_1)$: $((2, cs), (6, cs), (9, cs))$
- $P_{sense}(X_2, \bar{X}_2)$: $((12, cs), (15, cs))$

The function returns $C_{new_sense} = \{(6, cs), (9, cs), (12, cs), (15, cs)\}$ ■

C. Connecting a Sensing Barrier

Function connect_SB (Fig. 7) aims at adding new nodes to configuration $C \cup C_{new_sense}$ to satisfy condition (b) in Lemma 1. That is, we want to ensure that each node on each



sensing barrier path can reach the sink. The function does not fail since care has been given in selecting each node on any such path to ensure that the node can reach the sink in (G_{com}, C) .

Step 3 sets V_{SBP} to nodes in the computed sensing barrier paths; each node in the set needs to be connected to the sink in the final solution of the E2P problem. Steps 4 to 9 iterate until all nodes in V_{SBP} are connected to the sink. Step 5 finds a most probable path P in $(G_{com}, \tilde{p})$ that links some node, denoted x_{new} , to the sink. Steps 6 to 8 assign a state to each free node x in P , and adds the node-state pair (x, s_x) to C_{new_com} , where state $s_x \in \{cs, cns\}$ is the state of x with higher p_{s_x} probability. Step 8 updates $\tilde{p}(x)$ to 1.

Step 1 initializes $C_{new_com} = \emptyset$. Step 2 initializes the array $\tilde{p}(x \in G_{com})$ that guides the function to favour selection of nodes that are either already assigned a state in $\{cs, cns\}$, or a free node that has a high $\max(p_{cs}(x), p_{cns}(x))$ probability. In more detail, Step 2 initializes $\tilde{p}(x)$ as follows:

- 1) If $C(x)$ (x 's state in C) is *fail* then set $\tilde{p}(x) = 0$.
- 2) Else, if x 's state in $C \cup C_{new_sense}$ is in $\{cs, cns\}$ then set $\tilde{p}(x) = 1$.
- 3) Else, if x is a free node in C then set $\tilde{p}(x) = \max(p_{cs}(x), p_{cns}(x))$.

Example. Fig. 8 illustrates G_{com} for the network in Fig. 6. The shape of each node in the diagram specifies its state in the input configuration $C \cup C_{new_sense}$. Each non-sink node x is labelled with the weight $\tilde{p}(x)$ as described above where we assume that state *cms* has higher probability than state *cs*. Function `connect_SB` returns $C_{new_com} = \{(10, cms), (11, cms)\}$.

D. Running Time (of E2P)

Denote by n the number of nodes in the given WSN. The running time of function `find_SB` is determined by the loop in Steps 3 to 7. The loop iterates $|\mathbf{X}|$ times. Each iteration executes a most probable path algorithm (cf., Sec. V-A). Since $|\mathbf{X}| \in O(n^2)$, function `find_SB` requires $O(n^2 \cdot T_{SP}(n))$.

Similarly, the running time of function `connect_SB` is determined by the loop in Steps 4 to 9. The loop iterates $|V_{SBP}|$ times. Each iteration executes a most probable path algorithm. Since $|V_{SBP}| \in O(n)$, function `connect_SB` requires $O(n \cdot T_{SP}(n))$.

We conclude that function E2P requires $O(n^2 \cdot T_{SP}(n))$ time.

VI. NUMERICAL RESULTS

In this section, we investigate the performance of our approach for computing LBs and UBs for the DIR-BPDREL problem with 3-state nodes. The algorithms are implemented in C++ with the use of STL (Standard Template Library) container classes on a 2.4 GHz laptop computer with 8 GBytes of memory. The length of the documented LB and UB code is about 700 lines each.

We report on experiments done on WSNs formed by $W \times L$ grid nodes with W rows at (x, y) -coordinates $y =$

$0, 1, \dots, W - 1$, and L columns at coordinates $x = 0, 1, \dots, L - 1$. The sink node is located at the bottom left corner of the grid at coordinates $(0, 0)$. The communication (or sensing) direction of a node is specified by the parameters $(r, \theta_{mid} \pm \alpha)$, as described in Section II-A.

Note that the methodologies devised in the paper work for any pair of arbitrary G_{com} and G_{sense} graphs. In the graphs used in this section, nodes are arranged in a grid layout, however, some experiments vary the directionality parameters, and consequently the transmission and/or sensing radii are changed creating graphs that are less regular and much denser than regular grids.

Each WSN network has one pair of entry-exit sides with a corresponding protection interval $(X, \bar{X})$, where X is the set of top row perimeter grid nodes, and $\bar{X}$ is the set of remaining perimeter nodes. By default, the algorithm executes 10,000 iterations to obtain a bound.

1. Exact reliability results. Here, we explore the range of grid sizes for which the algorithms compute $Rel(G_{com}, G_{sense})$ exactly in time less than 0.5 hour. In the experiments, $DIR_{com} = (1.7, 135^\circ \pm 135^\circ)$, and $DIR_{sense} = (0.5, 225^\circ \pm 135^\circ)$ (cf., the examples in Section II-A). The LB (or UB) algorithm produces an exact result if it processes all of the generated configurations.

Table I presents the number of configurations generated by the LB and the UB algorithms, respectively. As can be seen, compared to the total number of states, the algorithms compute exact results using a significantly smaller number of configurations. Such reductions are achieved by avoiding the generation of configurations that can not be extended to either a pathset (for the LB algorithm), or a cutset (for the UB algorithm).

TABLE I: Exact computations

$W \times L$	Network states	Generated configurations (LB)	Generated configurations (UB)
2×2	3^3	17	17
2×3	3^5	43	21
3×3	3^8	395	167
3×4	3^{11}	2503	849
4×4	3^{15}	58199	21531
4×5	3^{19}	703411	384997

2. Gaps between LBs and UBs.

Fig. 9 illustrates the gap between the obtained LBs and UBs for networks composed of $W \times W$ grid nodes, where W (on the x-axis) varies in the range $[2, 9]$. We set $p_{com} = p_{sense} = 0.8$ for each node. G_{com} and G_{sense} correspond to using $DIR_{com} = (1.7, 135^\circ \pm 135^\circ)$, and $DIR_{sense} = (0.5, 225^\circ \pm 135^\circ)$, respectively. The upper and lower curves in Fig. 9 correspond to bounds obtained using 10,000 iterations. The middle LB curve is obtained using 500,000 iterations (with a corresponding increase in the number of the stored configurations.)

The gap between the bounds given by the upper and lower curves increases as the network size increases. This follows since increasing the number of nodes causes a rapid increase

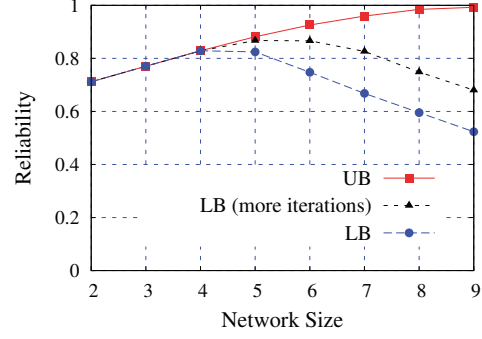


Fig. 9: Effect of varying network size on LBs and UBs

in the number of pathsets and cutsets. So, 10,000 iterations fail short in covering a reasonable number of configurations. In addition, the size of each pathset or cutset increases, and hence the contribution of each such a configuration to the final answer decreases as well.

Our past experience, however, indicates that the LB curve tends to be more accurate than the UB curve. For example, by increasing the number of iterations from 10,000 to 500,000, one obtains the middle curve in Fig. 9; this middle curve has a shape similar to the lower curve (with improved LB values). For networks with $W \geq 5$, each point of the middle curve requires about 7 hours of running time.

3. Effect of node communication and sensing probabilities.

In this part, we explore the effect of varying node's p_{com} and p_{sense} on $Rel(G_{com}, G_{sense})$. We present numerical results of a WSN formed by 6×6 grid nodes, as a representative of other obtained results.

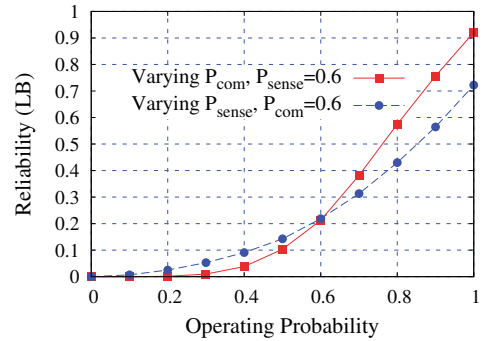


Fig. 10: Effect of varying p_{com} and p_{sense} on $Rel(G_{com}, G_{sense})$

Fig. 10 illustrates the results. The dashed curve corresponds to setting $p_{com} = 0.6$, and varying $p_{sense} \in [0, 1]$ for all nodes. Similarly, the solid curve corresponds to setting $p_{sense} = 0.6$, and varying $p_{com} \in [0, 1]$ for all nodes.

The results show that the dashed curve has better performance than the solid curve when the x-axis values vary in the

range $[0, 0.6]$. In this region, both curves result in nodes with the same p_{cs} ($= p_{com} \cdot p_{sense}$) probabilities. However, they differ widely in the produced p_{cns} ($= p_{com} \cdot (1 - p_{sense})$) probabilities.

Specifically, the dashed curve results in p_{cns} varying in the range $[0.6 \times (1.0 - 0.6), 0.6 \times (1.0 - 0.0)]$ whereas the solid curve results in p_{cns} values varying in the range $[0.0 \times (1.0 - 0.6), 0.6 \times (1.0 - 0.6)]$. Thus, the dashed curve results in better p_{cns} values.

The p_{cns} probabilities are used in selecting relay nodes needed to construct pathsets. This explains the better performance of the dashed curve when the x-axis values vary in the range $[0, 0.6]$. Following a similar reasoning, one can explain the behaviour of the curves when the x-axis values vary in the range $[0.6, 1.0]$.

4. Effect of directional parameters. Directional communication and sensing devices can achieve spatial selectivity without significant changes in the consumed power. In this part, we explore the effect of changing DIR_{com} and DIR_{sense} on $Rel(G_{com}, G_{sense})$. We present numerical results of a WSN formed by 6x6 grid nodes. We set $p_{com} = p_{sense} = 0.8$ for all nodes. DIR_{com} and DIR_{sense} are specified by $(r = \frac{360}{2 \cdot \alpha}, 180^\circ \pm \alpha)$, where the offset angle $\alpha \in \{30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ, 180^\circ\}$.

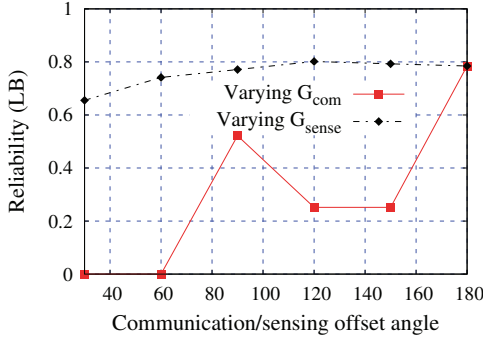


Fig. 11: Effect of varying the offset angle $\alpha \in \{30^\circ, 60^\circ, \dots, 180^\circ\}$ of DIR_{com} and DIR_{sense} on $Rel(G_{com}, G_{sense})$

Fig. 11 illustrates the obtained results. The solid curve corresponds to setting $\alpha = 180^\circ$ to obtain G_{sense} , and varying $\alpha \in \{30^\circ, 60^\circ, \dots, 180^\circ\}$ to experiment with different G_{com} graphs. In the histogram of Fig. 12, columns with dark colour give the number of links in the constructed G_{com} graphs (in the range 0 to 60 links). We note that the performance given by the solid curve of Fig. 11 has a strong correlation to the number of links in G_{com} ; that is, communication graphs with fewer edges give lower reliability values.

The dashed curve corresponds to setting $\alpha = 180^\circ$ to obtain G_{com} , and varying $\alpha \in \{30^\circ, 60^\circ, \dots, 180^\circ\}$ to experiment with different G_{sense} graphs. In the histogram of Fig. 12, columns with light colour give the number of links in the

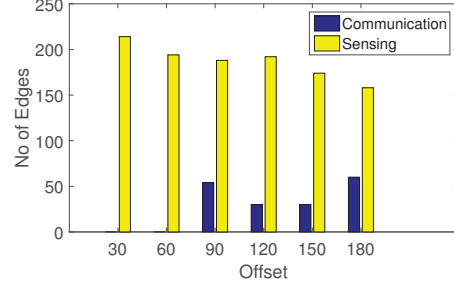


Fig. 12: Number of links in different G_{com} and G_{sense} graphs obtained by varying the offset angle α

constructed G_{sense} graphs (in the range 158 to 214 links). In general, the obtained G_{sense} graphs have considerable number of links (≥ 158) that support the construction of many sensing barrier paths $P_{sense}(X, \bar{X})$. Here, the obtained LBs exceed 0.60. We note, however, when the offset angle α is small (e.g., $\alpha = 30^\circ$) the sensing links are confined to a narrow region around each node compared to networks where α is large (e.g., the omnidirectional case where $\alpha = 180^\circ$). Thus, the distribution of links in G_{sense} graphs obtained using large offset angles allows the construction of more sensing barrier paths. This explains the observed trend where large offset angles in G_{sense} result in better reliability values.

VII. CONCLUDING REMARKS

Current technological advances have enabled the production of low cost wireless sensor nodes with directional communication and sensing abilities. Accordingly, mathematical modelling to analyze network wide performance should strive to support such advanced features while taking into consideration situations where components fail during operation.

To this end, we develop an approach for analyzing networks with directional sensor nodes deployed for an application where a WSN guards against unauthorized area traversal. Our model separates the network's sensing graph from its communication graph as a flexible way to model separate directional sensing and communication functions. In addition, the model separates events of communication failure from sensing failure within any single node. Under the above general model, we devise algorithms to assess the dependability of the network in such a critical application. The obtained results give lower and upper bounds on an underlying reliability measure. For future work, we consider extending our methodologies to tackle other WSN protection and security applications.

REFERENCES

- [1] D. Tao and T. Wu, "A survey on barrier coverage problem in directional sensor networks," *IEEE Sensors Journal*, vol. 15, no. 2, pp. 876 – 885, February 2015.
- [2] M. Guvensan and A. Yavuz, "On coverage issues in directional sensor networks: A survey," *Ad Hoc Networks*, vol. 9, no. 7, pp. 1238 – 1255, September 2011.
- [3] L. Liu, X. Zhang, and H. Ma, "Minimal exposure path algorithms for directional sensor networks," in *IEEE Global Telecommunications Conference (GLOBECOM)*, Honolulu, HI, November 2009, pp. 1 – 6.

- [4] Y. Song, L. Liu, H. Ma, and A. Vasilakos, "A biology-based algorithm to minimal exposure problem of wireless sensor networks," *IEEE Transactions on Network and Service Management*, vol. 11, no. 3, pp. 417 – 430, August 2014.
- [5] J. Wang and X. Zhang, "3d percolation theory-based exposure-path prevention for optimal power-coverage tradeoff in clustered wireless camera sensor networks," in *IEEE Global Telecommunications Conference (GLOBECOM)*, Austin, TX, December 2014, pp. 305 – 310.
- [6] L. Liu, X. Zhang, and H. Ma, "Exposure-path prevention in directional sensor networks using sector model based percolation," in *IEEE International Conference on Communications (ICC)*, Dresden, June 2009, pp. 1 – 5.
- [7] L. Zhang, J. Tang, and W. Zhang, "Strong barrier coverage with directional sensors," in *IEEE Global Telecommunications Conference (GLOBECOM)*, Honolulu, HI, November 2009, pp. 1 – 6.
- [8] C. Cheng and K. Tsai, "Distributed barrier coverage in wireless visual sensor networks with β -QoM," *IEEE Sensors Journal*, vol. 12, no. 6, pp. 1726 – 1735, June 2012.
- [9] Z. Yu, F. Yang, J. Teng, A. Champion, and D. Xuan, "Local face-view barrier coverage in camera sensor networks," in *IEEE INFOCOM*, Kowloon, April 2015, pp. 684 – 692.
- [10] Y. Hu, X. Wang, and X. Gan, "Critical sensing range for mobile heterogeneous camera sensor networks," in *IEEE INFOCOM*, Toronto, ON, April 2014, pp. 970 – 978.
- [11] M. H. Shazly, E. Elmallah, and J. Harms, "On breach path detection reliability of wireless sensor grids," in *Computer Communications and Networks (ICCCN)*, 2012.
- [12] M. H. Shazly, E. S. Elmallah, and J. Harms, "An approach for bounding breach path detection reliability in wireless sensor networks," in *the IEEE 37th Conference on Local Computer Networks (LCN)*, 2012.
- [13] M. Elmorsy and E. S. Elmallah, "On pathsets and cutsets of a wireless sensor network surveillance problem," in *Proceedings of the IEEE ICC*, 2014.
- [14] H. AboElFotouh and C. Colbourn, "Computing 2-terminal reliability for radio-broadcast networks," *IEEE Transactions on Reliability*, vol. 38, no. 5, pp. 538–555, December 1989.
- [15] D. D. Harms, M. Kraetzl, C. J. Colbourn, and J. S. Devitt, *Network Reliability Experiments with a Symbolic Algebra Environment*. CRC Press, 1995.
- [16] M. Elmorsy, E. S. Elmallah, and H. M. AboElFotouh, "On path exposure in probabilistic wireless sensor networks," in *the 38th IEEE Conference on Local Computer Networks (LCN)*, 2013.