Distributed Adaptive Beam Nulling to Mitigate Jamming in 3D UAV Mesh Networks

Suman Bhunia, Shamik Sengupta
Dept. of Computer Science and Engineering, University of Nevada, Reno, USA, 89557
sbhunia@nevada.unr.edu, ssengupta@unr.edu

***Abstract*—With the advancement of unmanned aerial vehicles (UAV), 3D wireless mesh networks will play a crucial role in next generation mission critical wireless networks. Along with providing coverage over difficult terrain, it provides better spectral utilization through 3D spatial reuse. However, being a wireless network, 3D meshes are vulnerable to jamming/disruptive attacks. A jammer can disrupt the communication, as well as control of the network by intelligently causing interference to a set of nodes. This paper presents a distributed mechanism of avoiding jamming attacks by means of 3D spatial filtering where adaptive beam nulling is used to keep the jammer in null region in order to bypass jamming. Kalman filter based tracking mechanism is used to estimate the most likely trajectory of the jammer from noisy observation of the jammer's position. A beam null border is determined by calculating confidence region of jammer's current and next position estimates. An optimization goal is presented to calculate optimal beam null that minimizes the number of deactivated links while maximizing the higher value of confidence for keeping the jammer inside the null. The survivability of a 3D mesh network with a mobile jammer is studied through simulation that validates an 96.65% reduction in the number of jammed nodes.**

***Keywords*—3D mesh, MANET, VANET, directional antenna, adaptive beam nulling, Kalman filter, tracking, jamming**

I. INTRODUCTION

The advent of class-1 (micro) and class-2 (small) unmanned aerial vehicles (UAV) and unmanned ground vehicles (UGV) is not only increasing the trend of 3D wireless mesh networks, but also providing a new dimension to next generation mission critical networking and service provisioning [1]. From the public safety sectors' standpoint 3D-mesh networking is a way to solve the infrastructureless radio access problem to augment emerging communications capabilities [2].

Even though UAV 3D mesh networks are recently emerging, such networks are also prone to various attacks by smart malicious agents [1]. The broadcast nature of wireless medium makes these networks vulnerable to unwanted interference and jamming [3]. Interrupting a subset of links in a mesh network may result in service disruption over a wide area. The capability of autonomous movement further strengthens the adversary, which can jam a desired area to cause the highest impact on the mesh network.

Earlier work on the defense against jamming attacks includes spatial retreat, spread spectrum, frequency hopping, mapping jammed regions, placing decoys, etc. In spatial retreat, nodes relocate to new positions in order to avoid jamming [4]. In direct sequence spread spectrum [5], a network uses wide band spectrum and transmits with lower data rates to communicate in presence of a strong jamming signal. In frequency hopping, a network switches its operating frequency upon detection of jamming. In jamming mapped region approach, a network route packets around the jammed region

This research was supported by NSF CAREER grant CNS #1346600

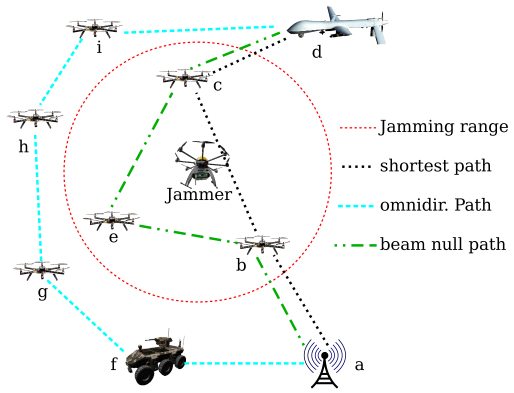


Fig. 1: Advantage of beamnulling

[3]. In decoy based technique [6], an assailant is lured to attack decoys by learning the strategy of attack. The downside associated with these techniques are that these require additional resources, for example, frequency hopping or spread spectrum techniques require the availability of wider spectrum, the honeynet requires extra resources as decoys. In other cases, the nodes within a jamming radius are deactivated, which is not desirable for mission critical networks.

Isotropic antennas are inherently used in above scenarios due to their simple design requirement. A large portion of a 3D mesh can be disrupted by jamming a smaller subset of nodes. An example can be seen in Figure 1 where nodes $a, b, c, d, e, f, g, h, i$ are located at different heights. In absence of any jammer, d can communicate with a through intermediate nodes b and c. Now as the jammer comes into the picture, nodes b and c are unable to communicate with isotropic antennas. The routing protocol now determines a new route through nodes f, g, h and i. This approach not only increases the end to end delay, but also escalates congestion on the intermediate nodes. An alternative approach that does not require additional nodes and preserves bandwidth, is spatial filtering with beamforming antenna arrays [7]. This approach exploits the beamformers' ability to detect the Direction of Arrival (DoA) of signals. This direction is then used to modify the array's response, so the interference sources are placed in the nulls of the antenna. Beamforming antenna systems that implement this mechanism are known as Adaptive Nulling Antennas (ANA). With ANA, node b and e can create beam nulls towards the jammer and still communicate with each other. So, a and d can communicate using the path $a - b - e - c - d$. The ability of adaptive beamnulling not only helps maintain the communication in the jammed region but also reduces overhead on other nodes.

Majority of the work on ANA assumes a jammer to be stationary [8]. Also, the effect of ANA has not been investigated for 3D mesh networks. In this paper we investigate

the applicability of adaptive beam nulling on the survivality of a mesh network. A beam null is defined by the cut off angle inside which the gain of an antenna is zero. In a 2D plane, a node can decide two beam null borders by estimating the most likely region where the jammer may move, measured in terms of horizontal. In 3D space, the probability estimates of angular position can be denoted by azimuth and altitude angles, ϕ, θ. As the jammer is mobile, constructing beam null for every position of it in continuous time manner degrades the legitimate communication with other neighbors. Hence it is more feasible for a beam null to be constructed at discrete intervals while estimating future movements of the jammer in the next interval. Creating a beam null with hard borders would create a wide beam null. Thus, the joint probability distribution of an estimated path should be estimated with noisy measurement of jammer's DoA. This paper aims to find the optimal way to determine the beam null border in the 3D space.

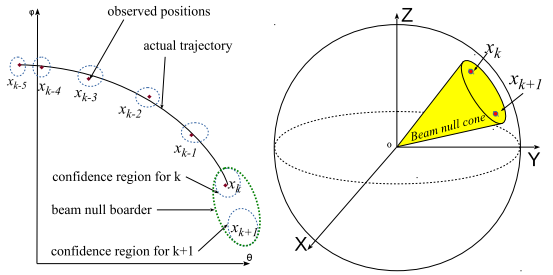
Again, in a mesh network, centralized controlling is not preferred due to delay imposed on the control messages and anticipation of adrift control messages. Thus, in this paper we present a technique where the nodes perform beam nulling dynamically in a completely distributed manner. Each node in a 3D mesh observes the Direction of Arrival (DoA) of the jammer's signal relative to observer's local coordinates. A node can predict the most likely trajectory of the jammer by learning from the history of its movements. However, hardware limitations induce error in the measurement. To resolve this issue, the Kalman filter [9] is used. With the Kalman filter, this noisy observed data can be smoothened and an estimation can be obtained for the position of the jammer. At each step, a confidence region can be mapped for both the current position and predicted movement. Each node uses this area as its beam null in order to avoid jamming until the next DoA measurement of the position of the jammer. The null becomes very wide if we require higher confidence in estimation which, in effect, increases the number of legitimate neighbors to be shadowed in the null. With the known position of neighbors, a node should address this trade off by choosing optimal confidence to create beam null. This paper investigates the issue of beam null width and proposes an optimization technique which determines the optimal beam null to be used. The simulation results confirms the effectiveness of the proposed mechanism as average node jamming is reduced by 96.65%, connectivity is increased up to 42.47% and 91.21% less islands are created.

II. Theory

In this work, we use beam forming antennas for creating a null towards a jammer. If an antenna poses the same gain for all direction it is called an isotropic antenna. A directional antenna poses different gain in different direction. The representation of gain in all direction is called antenna radiation pattern. A beam forming antenna uses array of antenna elements where different weights are assigned on each element in order to create the desired gain pattern. In case of ANA, the weights are assigned in such a way that the radiation pattern creates a null in the desired direction.

The movement of a jammer can be monitored in the 3D plane by a node *w.r.t.* to the observer's local coordinates [1].

[1]The exact process of jamming detection is beyond the scope of this paper. However, our proposed mechanism is valid as long as jamming can be detected through any existing mechanisms [3], [4], [10].



(a) 2D representation (b) Actual beam null

Fig. 2: Tracking the jammer and creation of beam null

At a time interval of τ, each node measures the Direction of Arrival (DoA) of the jammer's signal in terms of (θ_k, ϕ_k). The history of the movement of the jammer can be used to predict possible movement of the jammer in the following stage.

Figure 2a provides a simplified two dimensional representation of the movement of the jammer. Values of θ_k are plotted on the X-axis whereas the Y-axis represents ϕ_k of measured DoA. The solid black line represents the actual movement of the jammer. A node may err in measuring DoA due to many reasons such as hardware degradation, multipath propagation of the jammer's signal, etc. The red points indicate the measured DoA at each step k. After observing the history of jammer's movement, a node can efficiently predict the possible trajectory of the jammer within the next observation at $k+1$. As noise is incurred in measuring the DoA, a node should use a confidence region for estimating the movement of the jammer. At step k, a confidence region for DoA of the jammer at k can be determined along with a confidence region for estimated location of the jammer at step $k+1$. A region can be mapped that includes the possible trajectory of jammer in between observations k and $k+1$. Figure 2a represents the cross section of null border in terms of θ and ϕ whereas Figure 2b provides an illustration of the created null in 3D. The signal from a node in the null region will have 0 gain. Thus, signals from the jammer as well as the neighbors that are shadowed in the null will not reach a node's receiver. The goal of the current work is to determine the optimal beam null that minimizes the number of link failures due to the beam null while minimizing probability of attack.

A. Problem Statement

Figure 3 provides an illustration of the effect of beam-nulling against a moving jammer. The picture emphasizes the effect of null region size of node a in between sensing intervals k and $k+1$. The one hop links for node a with its neighbors b, c, d and e are depicted here. Node a observes the DoA of the jammer at every sensing period $k \in [0, 1, \dots]$. As the observation is not continuous, a takes into account the movement of jammer in between the sensing periods. By learning from the history of the position of the jammer, a node can predict the probable trajectory of the attacker at step $k+1$. If movement pattern of the jammer is random, the prediction accuracy decreases. Therefore a buffer region should be considered which will guarantee to keep the movement of the jammer within the buffer zone. This buffer zone can be used to create the beam null.

Again, Figure 3 presents two scenarios of the beam nulling. If a uses a wider null, the probability of the jammer movement being inside the null increases with drawback of more neighbors to be shadowed in the null. With a wider null, a

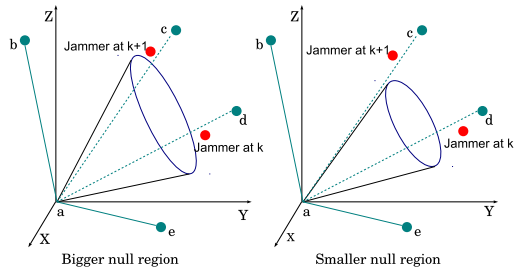


Fig. 3: Trade-off of having wider null region

can maintain links with b and e whereas links to c and d fail. With a narrow null as depicted in the figure, a can preserve link c. However, as the jammer moves to the position in step $k+1$, the jammer falls outside of the beam null used by a. As soon as a is exposed to the jammer, a would experience jamming that results in failure in all links of a. The trade-off for widening the beam null to cover the probable movement of the jammer with higher confidence comes at a cost of disabling some unaffected links. Hence, the goal of this paper is to derive an optimization technique that considers the cost and benefits of beam null and finds out the optimal null region.

B. System assumptions

i) A jammer is considered to be a node that emits jamming signal on the operating spectrum of a mesh network. It has the capability of moving in 3D space and follow a trajectory model for higher impact [11].

ii) A node senses DoA of jammer at τ seconds interval and communicates with its neighbors in between sensing intervals. Estimation of DoA has been widely studied in literature which can be broadly classified into beamscan algorithm and subspace algorithm. A thorough review and comparison of widely used DoA estimation methods has been provided in [12]. The current work does not deal with measuring DoA with actual antenna arrays. Instead it simply assumes that DoA can be measured with an error that follows joint normal distribution over θ, ϕ.

iii) The antenna is considered to be an ideal beam null antenna that poses a gain of 0 in the null region or null cone. In literature, these kind of antennas are used for theoretical modeling of MAC layer, and are commonly referred to as an ideal directional antenna. Time required to change the null border is negligible compared to the change in the jammer's position. Once the desired angular direction and the width of the null are determined, the node calculates the weight values for antenna elements. Some of the major weight calculation methods are Dolph-Chebyshev weighting, Least Mean Squares (LMS) and Conjugate Gradient Method (CGM) [7].

iv) A link between two nodes fails if either of the two nodes are jammed or one of the nodes fall in the beam null of that neighbor. The MAC layer is unaware of the beamform and the complexity of directional MAC protocol [13] is avoided considering the shadowed neighbors as out of range.

C. Tracking movement of a jammer

The Kalman filter provides an estimation of the position of the jammer when there is an error in obtaining the jammer's position. After obtaining the history of possible jammer's position, the node obtains the next possible position of the jammer using multivariate time series analysis. The Kalman

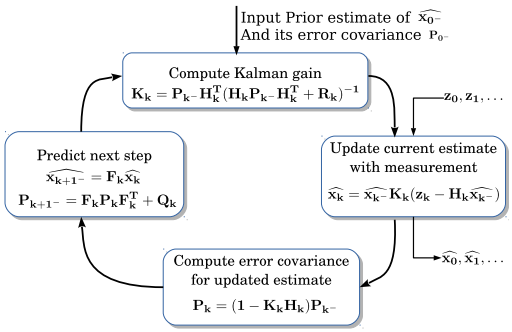


Fig. 4: Kalman Filter iteration [9]

filter is a recursive equation that aims at minimizing the mean-square estimation error of a random variable $\mathbf{x}$. It assumes that a random process to be estimated can be modeled in the form $\mathbf{x_{k+1}} = \mathbf{F_k x_k} + \mathbf{w_k}$. Measurements of this process occur at discrete time intervals. The filter assumes a linear relationship between the observation and the actual state of the process as: $\mathbf{z_k} = \mathbf{H_k x_k} + \mathbf{v_k}$. Here, $\mathbf{x}_k$ and $\mathbf{z}_k$ are the actual state and measurement at time t_k. $\mathbf{F}_k$ and $\mathbf{H}_k$ are state transition matrix and observation matrix giving the noiseless connection between the measurement and state respectively. $\mathbf{w}_k$ and $\mathbf{v}_k$ are input white noise and measurement error respectively. Since the error in DoA measurement or noise in the process follow Gaussian distribution, we can consider, $\mathbf{w_k} \sim \mathcal{N}(0, \mathbf{Q})$ and $\mathbf{v_k} \sim \mathcal{N}(0, \mathbf{R})$.

Figure 4 provides a representation of the Kalman filter process [9] which starts with an initial or apriori estimate about the first observation and its covariance. At every step k, it takes measurement $\mathbf{z_k}$ and updates the estimated state ($\widehat{\mathbf{x_k}}$) of the actual process. The covariance of the estimated state ($\mathbf{P_k}$) is also updated. Then it predicts the state of the actual process on the next step ($\widehat{\mathbf{x_{k+1^-}}}$) and the covariance of the predicted next step ($\mathbf{P_{k+1^-}}$). Then it updates the gain of the filter $\mathbf{K_k}$ and waits for the next measurement.

Now, for our system, the actual state of the jammer ($\mathbf{x_k}$) consists of four variables: $\theta, \dot{\theta}, \phi, \dot{\phi}$. Here $\dot{\theta}$ and $\dot{\phi}$ are velocity in θ and ϕ directions respectively. The state transition equation can be written as:

$$\begin{bmatrix} \theta(k+1) \\ \dot{\theta}(k+1) \\ \phi(k+1) \\ \dot{\phi}(k+1) \end{bmatrix} = \begin{bmatrix} 1 & \tau & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \tau \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \theta(k) \\ \dot{\theta}(k) \\ \phi(k) \\ \dot{\phi}(k) \end{bmatrix} + \mathbf{w_k} \quad (1)$$

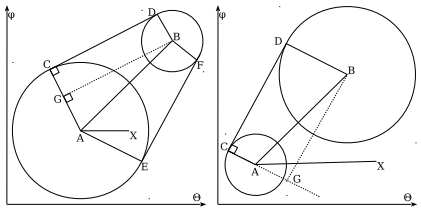
A node can observe only the position of the jammer in terms of θ and ϕ. Then observation relation can be written as:

$$\begin{bmatrix} z_\theta(k+1) \\ z_\phi(k+1) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \theta(k) \\ \dot{\theta}(k) \\ \phi(k) \\ \dot{\phi}(k) \end{bmatrix} + \mathbf{v_k} \quad (2)$$

D. Constructing beam null

At each step k, a node observes the position of jammer in terms of θ and ϕ. This observation is fed to the Kalman estimator which determines the estimated current position of the jammer ($\widehat{\mathbf{x_k}}$) and predicts position of the jammer at next step ($\widehat{\mathbf{x_{k+1^-}}}$). We construct two circles $\bigcirc_A$ and $\bigcirc_B$ whose centers are at the current and predicted position respectively.

$$\begin{bmatrix} \theta_A \\ \phi_A \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \widehat{\mathbf{x_k}}; \quad \begin{bmatrix} \theta_B \\ \phi_B \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \widehat{\mathbf{x_{k+1^-}}}$$



(a) $r_A > r_B$ (b) $r_A < r_B$

Fig. 5: Schema for obtaining common outer tangents

Two confidence regions are determined that enforces certain confidence level for the estimation process. Having a bigger diameter for the confidence circles will result in a greater probability that the jammer is inside the circle. We consider the diameter of the circles to be s times the standard deviation of the estimated position and the predicted position. The filter also provides two covariance matrices: covariance for the current position estimation ($\mathbf{P_k}$) and covariance for predicted position in next step ($\mathbf{P_{k+1^-}}$). $\mathbf{P_k}$ contains $cov_k(\theta,\theta)$ and $cov_k(\phi,\phi)$. $\mathbf{P_{k+1^-}}$ contains $cov_{k+1^-}(\theta,\theta)$ and $cov_{k+1^-}(\phi,\phi)$. The radii for $\bigcirc_A$ and $\bigcirc_B$ are respectively,

$$r_A = \frac{s}{2}\sqrt{\max(cov_k(\theta,\theta), cov_k(\phi,\phi))} \quad (3)$$

$$r_B = \frac{s}{2}\sqrt{\max(cov_{k+1^-}(\theta,\theta), cov_{k+1^-}(\phi,\phi))} \quad (4)$$

The beam null contains two circles and the region where the jammer may be in between two measurement updates. It is estimated that jammer is inside $\bigcirc_A$ at step k and predicted to be inside $\bigcirc_B$ at $k+1$. If the jammer moves straight in between the two measurement interval then it can only move in the area covered by the two circle and their outer tangents. If one circle cover the entire region (i.e. one circle stays inside the other) then the beam null will only be the bigger circle. The condition for this is as follows:

$$\max(r_A, r_B) > \min(r_A, r_B) + \sqrt{(\theta_A-\theta_B)^2 + (\phi_A-\phi_B)^2} \quad (5)$$

If the above condition is not valid then the null area is determined by calculating the common outer tangents.

1) Determining the outer tangents: Let the center points of two circles $\bigcirc_A$ and $\bigcirc_B$ be $A(\theta_A, \phi_A)$ and $B(\theta_B, \phi_B)$ respectively. The radii for these two circles are r_A and r_B respectively. Let us have a look at representation of θ and ϕ on a 2D plane as illustrated in Figure 5a. From the previous section we have calculated $\theta_A, \phi_A, r_A, \theta_B, \phi_B$ and r_B. Now we are interested in determining the outer tangents that connect these two circles.

Let us consider the case of Figure 5a where $r_A > r_B$. Let us consider the tangents are CD and EF whose coordinates are unknown at this point. A tangent of a circle is always perpendicular to the line that connects the touching point and the center. Thus, $CD \perp AC$ and $CD \perp BD$. Which entails that $AC \parallel BD$.

Now, lets consider a point G on line AC such that length of $CG = r_B$. As $CD \perp AC$, $CD \perp BD$, and $CG = BD$, quadruple $GBDC$ is a rectangle. Thus, $GB \perp CG$. This entails, $GB \perp AG$. Now we can calculate:

$$\angle GAB = \cos^{-1}\frac{AG}{AB} = \cos^{-1}\frac{r_A - r_B}{\sqrt{(\theta_A-\theta_B)^2 + (\phi_A-\phi_B)^2}} \quad (6)$$

Let us draw a line AX that is parallel to θ axis. We can calculate: $\angle XAB = \tan^{-1}\frac{\phi_B-\phi_A}{\theta_B-\theta_A}$. Note that tan^{-1} provides same angle for first and third quadrant. So, we checked the sign of numerator and the denominator and then corrected the formula during simulation. If the target point is in third or fourth quadrant relative to the observer, then π should be added to the tan^{-1} angle. As $AC \parallel BD$, both of these lines are making the same angle with θ axis. The angle is $\angle GAB + \angle XAB$. Thus we can determine the position of C and D as:

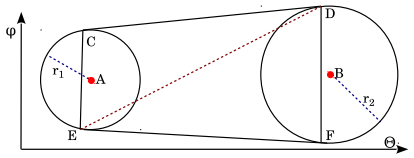


Fig. 6: 2D representation of the cross section of beam null

$$\theta_C = \theta_A + r_A\cos(\angle GAB + \angle XAB) \quad (7)$$

$$\phi_C = \phi_A + r_A\sin(\angle GAB + \angle XAB) \quad (8)$$

$$\theta_D = \theta_B + r_B\cos(\angle GAB + \angle XAB) \quad (9)$$

$$\phi_D = \phi_B + r_B\sin(\angle GAB + \angle XAB) \quad (10)$$

We know that the lines connecting the center and two outer common tangents make same angle with the line connecting the centers of the circles. Thus, $\angle EAB = \angle GAB$. So, $\angle XAE = \angle XAB - \angle GAB$. We can determine the positions for E and F as:

$$\theta_E = \theta_A + r_A\cos(\angle XAB - \angle GAB) \quad (11)$$

$$\phi_E = \phi_A + r_A\sin(\angle XAB - \angle GAB) \quad (12)$$

$$\theta_F = \theta_B + r_B\cos(\angle XAB - \angle GAB) \quad (13)$$

$$\phi_F = \phi_B + r_B\sin(\angle XAB - \angle GAB) \quad (14)$$

It can be easily proven that these equations also hold true for the case of $r_A < r_B$ as demonstrated in Figure 5b. Regardless of radii of the circles, we can use the above equation.

2) Determining if a node is inside beam null: The node needs to determine how many links would be broken due to the newly created null region. If a node j has DoA of (θ_j, ϕ_j) relative to node i, then j would be inside beam null if the DoA is inside either circle A, or circle B or inside the quadruple $CDEF$. Figure 6 depicts a 2D representation of θ, ϕ. Total area covered by the null is $Area(\bigcirc_A) \cup Area(\bigcirc_B) \cup Area(\square_{CDFE})$.

Now, let us consider the case of the quadruple $CDFE$. As two sides of this quadruple are outer tangent of two circles, we can divide the area into two non overlapping triangles: $\triangle_{CDE}$ and $\triangle_{DEF}$. So, the condition whether DoA of j is inside the null is: $\varrho_j = \varrho_{\bigcirc_A} \vee \varrho_{\bigcirc_B} \vee \varrho_{\triangle_{CDE}} \vee \varrho_{\triangle_{DEF}}$. Where $\varrho_{\bigcirc_A}, \varrho_{\bigcirc_B}, \varrho_{\triangle_{CDE}}, \varrho_{\triangle_{DEF}}$ are conditions for being inside circle A, circle B, triangle CDE and triangle DEF respectively. Note that if one circle is inside another circle as determined in (5), we have to check only for the bigger circle. Conditions for being inside circle A and B are:

$$\varrho_{\bigcirc_A} = \sqrt{(\theta_j-\theta_A)^2 + (\phi_j-\phi_A)^2} < r_A \quad (15)$$

$$\varrho_{\bigcirc_B} = \sqrt{(\theta_j-\theta_B)^2 + (\phi_j-\phi_B)^2} < r_B \quad (16)$$

Let us again look at the θ, ϕ representation on a 2D plane as in Figure 6. A point $j(\theta_j, \phi_j)$ is inside triangle CDE if the area of triangle CDE is same as the sum of area of triangles jDE, CjE, and CDj. The area of a triangle CDE can be calculated as:

$$\mathfrak{A}(\triangle_{CDE}) = \left|\frac{\theta_C(\phi_D-\phi_E) + \theta_D(\phi_E-\phi_C) + \theta_E(\phi_C-\phi_D)}{2}\right|$$

The condition to check for the DoA of j inside triangles are:

$$\varrho_{\triangle_{CDE}} = \begin{cases} 1 & if\ \mathfrak{A}(\triangle_{CDE}) = \mathfrak{A}(\triangle_{jDE}) + \mathfrak{A}(\triangle_{CjE}) + \mathfrak{A}(\triangle_{CDj}) \\ 0 & otherwisw \end{cases}$$

$$\varrho_{\triangle_{DEF}} = \begin{cases} 1 & if\ \mathfrak{A}(\triangle_{DEF}) = \mathfrak{A}(\triangle_{jEF}) + \mathfrak{A}(\triangle_{DjF}) + \mathfrak{A}(\triangle_{DEj}) \\ 0 & otherwisw \end{cases}$$

E. Optimization Goal

Let $\mathbb{N}$ be the set of nodes in a 3D mesh network. Let $j \in \mathbb{N}$ be an one hop neighbor of $i \in \mathbb{N}$. With a known beam null, i can assess the probability of the failure of link with j. Considering that j is not jammed, we can determine the probability that link ij fails as

$$\begin{aligned} \mathbb{P}(\text{ij fails}) &= \begin{cases} 1 & if\ \ \varrho_j = True \\ \mathbb{P}(\text{i is jammed}) & otherwise \end{cases} \\ &= \varrho_j + (1 - \varrho_j)\mathbb{P}(\text{node i is jammed}) \end{aligned} \quad (17)$$

The probability of a node successfully avoiding jamming is same as the probability that the jammer stays within the null during next transmission interval. As the error in the DoA estimation model is a normal distribution, we can say that probability of successful estimation would closely follow Chebyshev's inequality. In that case, if node i uses s_i standard deviation in (3) and (4) for calculating circle diameters, then probability of jammer in the estimated region $\approx 1 - 1/s_i^2$. For the optimization purpose, we can consider, probability of i being jammed is $1/s_i^2$. Thus probability of link between i, j fails is $\varrho_j + (1 - \varrho_j)/s_i^2)$

In 3D mesh networks every link has a different importance level in the network. For example if a link is relaying data from many nodes or a link is transmitting crucial data, it can be assigned higher weight. It is for the best interest of the network that these links are safe guarded from failure. Let w_{ij} denote the weight for a link between i and j. Thus, the optimization problem becomes:

$$maximize \qquad \sum_{j \in \mathbb{N}} w_{i,j} \left(\varrho_j + \frac{(1 - \varrho_j)}{s_i^2}) \right) \quad (18)$$

The lower value of s_i reduces the beam null region that in effect reduces number of deactivated links. But it comes with a cost that there is a higher probability of i being jammed that, in effect, deactivates all links of i. Again, a very high value of s_i increases the number of deactivated links. The maximization problem stated above is a convex optimization problem that computes optimal s_i at every step.

F. Algorithm

A node $i \in \mathbb{N}$ follows this algorithm at each step k to create the beam null. At first i observes the position of the jammer $z_\theta(k), z_\phi(k)$. If it is being jammed and there is not enough data to predict the possible trajectory of the jammer, i creates a beam null cone centering on $z_\theta(k), z_\phi(k)$, using a threshold value r_{th} as the radius of the beam null cone. Note that this fixed radius cone would be used only at the first few steps of the observations. After that, at each step k, the position estimates of jammer $(\mathbf{x_k}, \mathbf{x_{k+1^-}})$ and the covariances $\mathbf{P_k}, \mathbf{P_{k+1^-}}$, are calculated. The optimal s_i, calculated from (18), is used to determine the effective beam null. Node i uses this beam null until the next observation at step $k + 1$. At each step, the kalman filter algorithm is iterated, which takes negligible amount of processing time since it deals with only matrix multiplications. The maximization of s_i also runs in the order of $\mathbb{N}$ as it checks with the neighbors of i whether j falls in the null region or not. The creation of the null depends on the hardware efficiency which takes time in the order of μs.

TABLE I: Simulation Parameters

Parameters	Symbol	Values
Simulation area		$10,000 \times 10,000 \times 4,000\ m^3$
Transmission power		30 dBm
Received Power cutoff		-78 dBm
Communication Frequency		2.4 GHz
Communication Radious		3146 m
Sensing interval	τ	$50\ ms$
Simulation Time		$500\ s$
Path loss model		Free-space
Jammer's mobility model		Gauss-Markov
Transition covariance	$\mathbf{Q}$	4×4 identity matrix
Observation covariance	$\mathbf{R}$	2×2 identity matrix
Estimated initial state	$\widehat{\mathbf{x}_{0^-}}$	4×1 zero matrix
Initial state covariance	$\widehat{\mathbf{P}_{0^-}}$	4×4 identity matrix

III. SIMULATION

To evaluate the performance of the proposed algorithm, a tick based simulator is used. The ticks correspond to the sensing time with an interval of τ seconds. Table I lists all parameters used for the simulation. At each tick, all nodes measure the DoA for the jammer individually and create the desired beam null as described in Section II-F. After creating the beam null each node moves to the communication phase and carries the communication with neighbors outside the beam null for τ sec. The simulator generates positions for the nodes randomly in accordance with uniform distribution. While comparing the models, the same set of node positions are used.

A. Performance Metrics

To evaluate the performance of our proposed technique, four network parameters are used. *Average nodes jammed* defines the average number of nodes that are effectively jammed during a simulation. The second parameter is the *average number of active links*. A link between two nodes is considered to be deactivated if either of the corresponding nodes is attacked or one of the nodes fall in the beam null of the other. The total deactivated links are then divided by simulation time to obtain the average. We define *Connectivity* as the total number of connected pairs. This is a measure of how well connected the network is. It is defined as the summation of connected nodes. More precisely, connectivity of a network is $\frac{1}{2} \times (\sum_{i \in \mathbb{N}} \sum_{j \in \mathbb{N}} connected(i, j))$, where $connected(i, j) = 1$ if there exists at least one path from i to j, 0 otherwise. The next performance parameter considered is the *average number of islands*. Sometimes, a node or a group of nodes may be isolated from the rest of the network. If a network is completely connected, the number of island is 1. With more islands, network disruption increases.

B. Results

Two bench mark scenarios are considered to correctly evaluate the performance of the proposed scheme. The first being *isotropic antenna without jammer*, where all nodes use isotropic antennas for communication at the absence of any jammer. The second scenario is *isotropic antenna with a jammer*, where isotropic antennas are used for communication in the presence of a jammer. The third scenario uses the proposed adaptive beam nulling for avoiding the jammer. Figure 7 depict the simulation results, carried out for various node densities. Since the simulation area is fixed, the number of nodes, represented in x-axis reveals the node density.

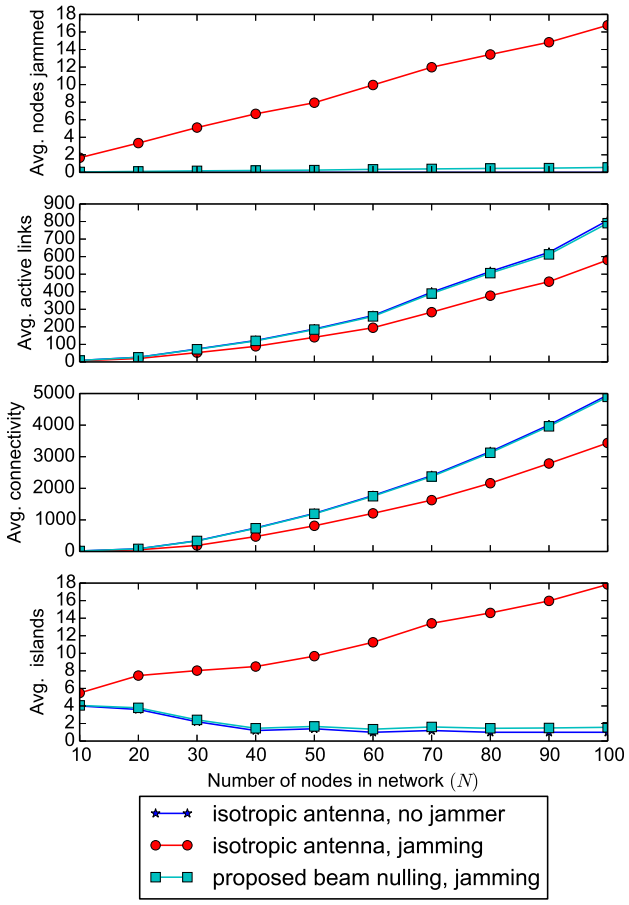


Fig. 7: Simulation Results

The upper subplot compares of average number of jammed nodes. As the density of nodes increases, more nodes are attacked as can be seen in the subplot. Some nodes will observe jamming due to inaccurate prediction. Note that a node would also experience jamming if the jammer was not in the vicinity in the previous step and, as a result, the node did not use beam nulling in that step. However, with the proposed scheme, nodes manage to keep the number of jammed nodes close to the ideal case of the no jammer scenario. The results shows that with the proposed mechanism, a network can decrease the average jammed nodes up to 96.65%.

The second subplot depicts the average number of active links in the network during simulation. The plot reveals that with proposed adaptive beamnulling, the network can retain more active links in the presence of a jammer, where a network can retain 36.14% of its links that are jammed.

The third subplot provides average connectivity. For a fully connected network with 100 nodes, the connectivity is 4950, which can be seen in the plot. With our proposed scheme, the network remains almost unaffected in terms of connectivity, since the connectivity is close to the benchmark case of no jammer and can increase its connectivity to 42.47%.

In the lower subplot, the average number of islands are represented. For a very sparse network, the network is not well connected, hence, dividing the network into multiple islands. Even for the benchmark case of no jammer, multiple islands can exist. Multiple simulations with the same number of nodes (N) are repeated with different random node positions, and the average is taken to obtain reliable results. For some generated graphs, the random position will make the network partitioned into islands. Thus, the average number of islands is not one even for the higher value of N. It can be clearly observed that with our proposed algorithm, the network can keep the number of islands very close to no jammer case, decreasing the number of islands by 91.21%. It is noteworthy to mention that although there are many links deactivated, mostly due to neighbors being shadowed by beam null, the network remains connected. This proves that the proposed scheme successfully maintains the communication in jammed regions.

IV. CONCLUSION

A 3D mesh network is vulnerable to mobile jammer. This paper investigates the effectiveness of adaptive beam nulling in 3D mesh networks under attack from a moving jammer. A schema is modeled that determines the optimal beam null region from the estimated trajectory of jammer in near future. The mechanism uses Kalman filtering to estimate current and future position of a jammer from a history of noisy DoA measurements. The covariance of the two angles denoting the predicted DoA is used to create an optimal confidence region that minimizes link failure as well as increases prediction efficiency. If the jammer moves randomly, the defending nodes create wider nulls to keep the jammer in the null. Simulation results support the effectiveness of the mechanism where the number of jammed nodes decreases up to 96.65% compared to legacy isotropic communication.

REFERENCES

[1] I. Bekmezci, O. K. Sahingoz, and Ş. Temel, "Flying ad-hoc networks (FANETs): A survey," *Ad Hoc Networks*, vol. 11, no. 3, pp. 1254–1270.

[2] "FAA makes progress with UAS integration." http://www.faa.gov/news/updates/?newsId=68004.

[3] C. Sorrells, P. Potier, L. Qian, and X. Li, "Anomalous spectrum usage attack detection in cognitive radio wireless networks," in *IEEE HST, 2011*, pp. 384–389, IEEE, 2011.

[4] W. Xu, K. Ma, W. Trappe, and Y. Zhang, "Jamming sensor networks: attack and defense strategies," *Network, IEEE*, vol. 20, no. 3, pp. 41–47.

[5] C. Popper, M. Strasser, and S. Capkun, "Anti-jamming broadcast communication using uncoordinated spread spectrum techniques," *IEEE JSAC*, vol. 28, no. 5, pp. 703–715, 2010.

[6] S. Bhunia, S. Sengupta, and F. Vzquez-Abad, "Performance analysis of cr-honeynet to prevent jamming attack through stochastic modeling," *Pervasive and Mobile Computing*, vol. 21, pp. 133 – 149, 2015.

[7] J. Volakis, *Antenna Engineering Handbook, Fourth Edition*. McGraw-Hill Companies,Incorporated, 2007.

[8] D. Mingjie, P. Xinjian, Y. Fang, and L. Jianghong, "Research on the technology of adaptive nulling antenna used in anti-jam gps," in *Radar, 2001 CIE International Conference on, Proceedings*, pp. 1178–1181.

[9] R. G. Brown and P. Y. Hwang, *Introduction to Random Signals and Applied Kalman Filtering with Matlab Exercises*. Wiley, 2012.

[10] M. Spuhler, D. Giustiniano, V. Lenders, M. Wilhelm, and J. B. Schmitt, "Detection of reactive jamming in dsss-based wireless communications," *Wireless Communications, IEEE Transactions on*, vol. 13, no. 3, pp. 1593–1603, 2014.

[11] A. B. Gershman, U. Nickel, and J. F. Bohme, "Adaptive beamforming algorithms with robustness against jammer motion," *IEEE Transactions on Signal Processing*, vol. 45, no. 7, pp. 1878–1885, 1997.

[12] V. Krishnaveni, T. Kesavamurthy, *et al.*, "Beamforming for direction-of-arrival (doa) estimation: A survey," *International Journal of Computer Applications*, vol. 61, no. 11, pp. 4–11, 2013.

[13] O. Bazan and M. Jaseemuddin, "A survey on mac protocols for wireless adhoc networks with beamforming antennas," *Communications Surveys & Tutorials, IEEE*, vol. 14, no. 2, pp. 216–239, 2012.