Distributed Adaptive Beam Nulling to Survive Against Jamming in 3D UAV Mesh Networks ☆

Suman Bhunia[a], Paulo Alexandre Regis[b], Shamik Sengupta[b]

[a]*Dept. of Computer Science, University of California, Davis, USA 95616*
[b]*Dept. of Computer Science and Engineering, University of Nevada, Reno, USA 89557*

Abstract

In the future generation mission-centric tactical network, 3D UAV mesh network will play a crucial role for its several advantages. However, any adversarial node with the 3D movement capability poses a significant threat to these networks as the adversary can position itself to attack the crucial links. An adaptive beamnulling antenna is used to spatially filter out signal coming from a certain direction which can maintain the links active without requiring additional spectrum. However, determining the beamnull region is very challenging for jammer with mobility. This paper presents a distributed mechanism where nodes measure the jammer's direction in a discrete interval and determine the optimal beamnull for the next interval. 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 determine the 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 framework works in the physical layer and can work with any upper layer protocol. 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.

☆This research was supported by NSF CNS #1346600 and conducted when the primary author was at UNR. A preliminary version of this work was presented in IEEE ICNC 2017 [1].

Email addresses: sbhunia@ucdavis.edu (Suman Bhunia), pregis@nevada.unr.edu (Paulo Alexandre Regis), ssengupta@unr.edu (Shamik Sengupta)

Keywords: 3D mesh, MANET, VANET, directional antenna, beam nulling, Kalman filter, tracking, jamming

1. 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 networks but also providing a new dimension to next-generation wireless networking and service provisioning [2, 3, 4, 5]. The next generation public safety sector/first responders/law enforcement community, including on-site police, firefighters, emergency medical services understand the need to maintain robust and interoperable communications [6]. Thus from the public safety sectors' standpoint, enabling UAV technology with highly efficient 3D mesh networking is a great way to solve the problem of radio access without infrastructure, to augment current and emerging communications capabilities and has been the subject of many recent discussions within the first responder/law enforcement community [7, 8]. A network of UAVs or UAV and ground vehicles are commonly referred to as 3D mesh networks [9]. In future, autonomous UAVs are anticipated to play crucial roles in missions such as disaster monitoring, firefighter network, provide relay for ground networks, military or tactical field.

Even though UAV 3D mesh networks are recently emerging, such networks are also prone to various attacks by intelligent malicious agents [2]. The broadcast nature of the wireless medium makes these networks vulnerable to unwanted interference and jamming [10, 11, 12]. Interrupting a subset of links in a mesh network may result in service disruption over a wide area. The capability of movement in all three directions further strengthens the adversary, which can jam a particular area to cause the highest impact on the defending network. Hence, mitigating jamming attacks has been a crucial research issue for the wireless community [13].

In 3D mesh networks, information can be routed over a large geographic region through multiple relay nodes situated at different heights. Isotropic antennas are inherently used in this kind of network due to their simple design requirement. 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. An intelligent and moving jammer can position itself to jam a subset of nodes. Jamming a small subset of

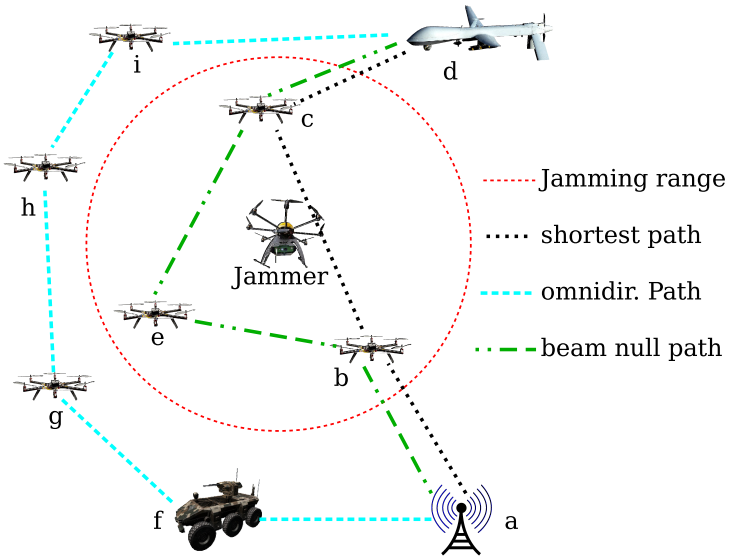


Figure 1: Advantage of beamnulling

the nodes can disrupt a large 3D mesh. In recent times, many defense mechanisms have been proposed against jamming such as spatial retreat, spread spectrum, channel surfing, etc. However, these approaches lack efficiency in aspects such as completely disconnecting nodes inside jammed region from the rest of the network or the requirement of additional spectrum. In our example, 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 [14]. 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). In this approach, nodes 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 the number of hops in a multi-hop mesh

network.

Recently there are some works in applying adaptive nulling antenna to avoid jamming [15, 16, 17, 18]. Also, with the advancement of software defined radios and digital signal processing, a transceiver can adaptively beamform and estimate the direction of arrival of a signal online [19, 20]. However, the effect of beamnulling on mobile mesh network has not been studied before. We investigate the applicability of adaptive beam nulling on the survivability of a mesh network by proposing a framework that determines optimal beamnull widths after observing the location of jammers at discrete time intervals. A beam null is defined by the cutoff 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 one angular position θ. In 3D space, the probability estimates of angular position can be denoted by θ (azimuth) and ϕ (altitude). Details about θ and ϕ are provided in section 3.2. 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. The probable path of the mobile jammer follows a joint probability distribution in terms of θ and ϕ. Creating a beam null with hard borders in θ and ϕ 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 plane.

Again, in a mesh network, centralized controlling is not preferred due to the delay imposed on the control messages and anticipation of lost messages. In our earlier work, we proposed a technique where the nodes perform beam nulling dynamically in a distributed manner [21]. Each node in a 3D mesh observes the Direction of Arrival (DoA) of the jammer's signal relative to observer's local coordinates. In that work, nodes used a higher and lower cutoff of angles both in θ and ϕ directions. To keep the jammer inside the beamnull, the defender nodes used safety buffer zone. In contrast, in this paper, we proposed a framework where the defender nodes can optimize the beamnull considering heterogeneous links and different confidence to keep the jammer inside beamnull after predicting the most likely trajectory of the jammer by learning from the history of its movements. The framework is further extended to optimize beam nulls for multiple jammers. However,

hardware limitations induce error in the measurement which can be reduced using Kalman filter [22] based tracking mechanism. With the Kalman filter, this noisy observed data can be smoothened, and an estimated position of the jammer can be obtained. 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 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 movement of neighbors, a node can choose an optimal beam null that maximizes the probability of keeping the jammer inside beamnull while reducing the number of link failures due to shadowing. This paper investigates the issue of beam null width and proposes an optimization technique which determines the optimal beamnull to be used considering multiple mobile jammers and heterogeneous links. Again, a group of nodes may be isolated from the other nodes due to multiple link failures. Thus, our algorithm also aims to reduce the number of islands (i.e. number of isolated group of nodes). The simulations confirm 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% fewer islands are created.

The remainder of this paper is organized as follows: Section 2 provides background studies on anti-jamming and beam nulling. Section 3 presents the system model and the problem statement. Section 4 describes the proposed methodology of adaptive beam nulling in 3D space. Section 5 describes the simulation setup and results. Finally, Section 6 concludes the paper.

2. Related works

A wireless jammer can launch a denial of service attack on a set of nodes by transmitting a strong noise signal. Detection of jamming is a widely studied topic in wireless communication [23, 24, 25, 26]. Our proposed algorithm is built with the assumption that jamming is detected with cross-layer jamming detection mechanism. A receiver can detect jamming when an anomaly is detected between the physical and data link layers parameters. For example, an anomaly can be when received signal strength is high but data is not decoded, or if a node receives beacons from other nodes but not actual message due to reactive jamming. Cross-layer mechanisms that observe several system parameters, such as carrier sensing time, packet delivery ratio, signal

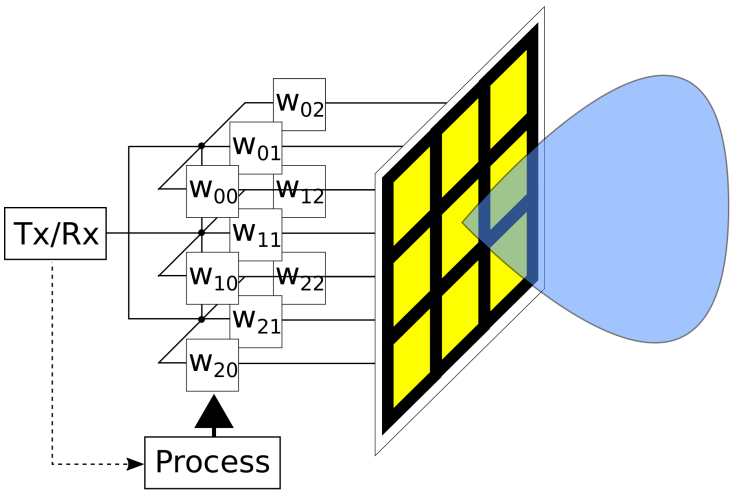


Figure 2: Schema for ANA

strength, etc. are measured, and the consistency check among them are used to detect jamming. As the current work aims at the theoretical framework for jamming prevention, the proposed model is built on the assumption that jamming is detected.

Earlier works on the defense against jamming attacks include spatial retreat, spread spectrum, frequency hopping, mapping jammed regions, placing decoys, etc. In spatial retreat, nodes relocate to new positions to avoid jamming [27]. Another approach is to use direct sequence spread spectrum [28], where a network uses wideband spectrum and transmits with lower data rates to communicate in the presence of a strong jamming signal. A network switches its operating frequency upon detection of jamming in the frequency hopping [29] technique. Another solution for intensely populated networks is to map the jammed region and route packets around the jammed region [30]. In [31, 32], we proposed CR-Honeynet, a decoy based mechanism where the network learns about the strategy of an attacker through the history of attacks and bypasses attacks by luring the assailant towards active decoy nodes. The constraints associated with these techniques are that these techniques either require additional resources or the nodes within the jamming region cannot communicate.

In contrast to the works mentioned above, we use adaptive beamforming antennas for spatial filtering to create a null gain towards a jammer. With the beamnulling approach nodes inside jammed region can also communicate

with neighbors without requiring additional resources. An antenna directs the energy with a different gain in different directions in terms of θ (azimuth) and ϕ (altitude). Figure 2 illustrates the logic circuitry of a beamforming antenna array. Each element processes the desired signal mixed with interference and noise. Different weights are assigned to each element by the control process 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. Once the desired angular direction and the width of the null are determined, the beamformer calculates the weight values for creating the desired antenna pattern with null.

Determination of weights on the antenna elements to create a desired beamform is widely studied in the literature. Some of the major weight calculation methods are Dolph-Chebyshev weighting, Least Mean Squares (LMS) and Conjugate Gradient Method (CGM) [33]. In the case of mobile ad hoc networks, in which the directions of desired and interference signals are unknown and may vary, Stochastic Search algorithms are applied [34]. Examples of such methods are Gradient Search Based Adaptive algorithms [35, 36, 37], Genetic Algorithms [38, 39, 40] and Simulated Annealing [41, 42]. Thorough reviews and comparison of beamforming methods and algorithms are provided in [34] and [43]. In our study, we consider that null is a cone in the isotropic beam pattern. The gain inside the null is negligible.

The direction of arrival (DoA) of the jamming signal needed to be measured before determining the beamnull. Estimation of DoA has been widely studied in the literature. Proposed algorithms can be broadly classified into beamscan algorithm and subspace algorithm [44, 45, 46]. In beamscan methods, a region is scanned with conventional beam and the square of the received signal magnitude is recorded. Minimum Variance Distortionless Response (MVDR) and root MVDR are two examples of this class. [47]. On the other hand, the orthogonality between the signal and noise subspaces are exploited in subspace algorithms. MUSIC, Root-MUSIC and ESPIRIT are among the most efficient subspace DoA estimation algorithms in antenna arrays. A thorough review and comparison of widely used DoA estimation methods have been provided in [48]. 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 a normal distribution over θ and ϕ. In the next section, we describe the theoretical framework to determine the optimal beamnull to keep the jammer's predicted movement within

the beamnull with discrete DoA measurement.

3. System Model

The defending network considered in this study is a multi-hop UAV mesh network and arbitrarily distributed in their operating space. Each node is equipped with an antenna array capable of DoA estimation and beamforming. The jamming attack is sought to be carried out by one or more entities that transmit jamming signals to cause destructive interference on the network. If the Signal to Interference and Noise Ratio (SINR) of node communications falls below a threshold, the receiver node is considered as jammed. The threshold value of SINR depends on the MAC protocol as well as the modulations and coding scheme. Since the approach proposed in this network is developed to operate on the physical layer, it remains independent of upper layer protocols such as MAC and routing.

3.1. System assumptions

i) The jammer is assumed to be a node that moves and transmits a disrupting signal on the same frequency as the ad hoc network. The defending network is following a formation while moving and the relative movement is negligible.

ii) Each node individually detect jamming. Our proposed algorithm is built on the assumption that jamming is detected as described in Section 2. Each node also monitors the DoA of the jammer relative to its local coordinates. The jammer's DoA is distinguished as the relative position of the neighbors are known. Nodes cannot determine the distance of a jammer accurately as it would require precise distance measuring hardware.

iii) Each node is equipped with a beamforming antenna array, capable of introducing nulls in its default isotropic radiation pattern. The antenna is considered to be an ideal beam null antenna that poses a gain of 0 in the null region or null cone. The time required to change the beam of ANA is negligible compared to the change in the jammer's position as the beamnull creation takes time in the order of microseconds while UAV position takes time in the order of seconds [15]. Beamformers are assumed to have sufficient spatial resolution to form the calculated nulled regions with sufficient accuracy [49, 50, 51]. Introducing a null in the

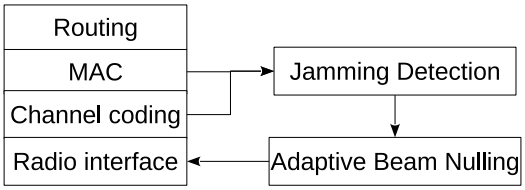


Figure 3: Representation of proposed mechanism

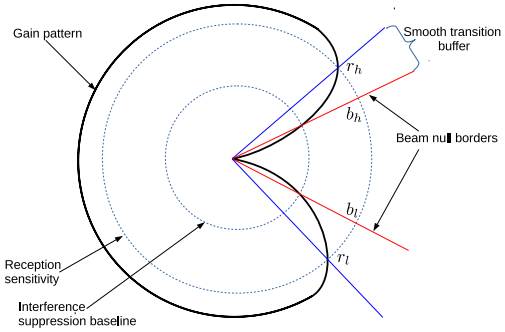


Figure 4: Depiction of null boundary

omnidirectional pattern of a beamforming node may be interpreted as changing the mode of communications to directional transmission, hence necessitating the use of Directional MAC protocols [52]. However, the higher network layers can operate under the default assumption of omnidirectional transmission, as the nulled region is already under jamming and no hidden/exposed terminal problem may arise from its direction [53]. This approach, therefore, eliminates the overheads associated with most directional communications schemes [54, 55, 56].

iv) A beam null is a region in the direction in which the antenna gain is so low that the signal arriving in this region will not have any effect on the receiver. Figure 4 illustrates an example of a gain pattern in 2D and its corresponding null borders. Here, b_h and b_l are the null borders. Within the receding lobes bounding the nulled region, the gain of received signals falls below the sensitivity threshold, while interference remains above the required cutoff. Hence, the entire transition region is blind to communications, which is accounted for by the addition of smooth transition buffers to the beam nulled angle. These regions are defined by borders r_h and r_l. As the gain pattern illustrated in this figure demonstrates, the nulled region is essentially bounded by receding lobes

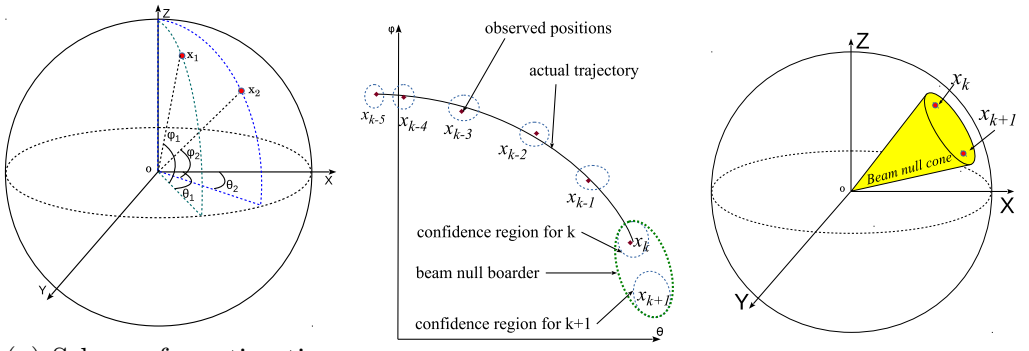
rather than sharp cutoffs. The signals arriving outside of these regions will have full reception. Communication is not possible with neighbors who lie in the buffer or the null region and hence considered as shadowed in the beam null. In the rest of this paper, we consider the beam null borders to be the boundary in which gain is below interference cutoff (i.e., b_h and b_l).

v) A node can not determine the DoA of jammer's signal while it is communicating with its neighbors. To determine the jammer's DoA, it goes through a sensing phase at every τ seconds interval. We assume that a node communicates with its neighbors in between sensing intervals. During normal communication phase, nodes cannot determine whether an interference is caused by another legitimate node in the network or by a jammer.

3.2. Mitigation of Jamming by Adaptive Beam Nulling

The proposed framework uses adaptive beamnulling to avoid jamming. Figure 3 provides a block representation of the relevant network layers in a node implementing this framework. The jamming detection module uses measured parameters from the medium access control (MAC) and physical layers such as carrier sensing time, packet delivery ratio, signal strength, etc. Various methods for detection of jamming signals have been proposed in the literature, but as the focus of this work is on mitigation of jamming, we assume jamming signals are detectable. Interested readers may refer to [23, 24, 25] for more details on detection techniques. The adaptive beam nulling block uses the DoA measurement of jamming signal and dynamically modifies the beamforming weights of the radio interface to create a null towards the jammer. The upper layer protocols are unaffected by the beam nulling procedure. If a link fails due to a node falling inside the beam null of its neighbor, the routing protocol treats this as a link failure and utilizes an alternative route.

The movement of a jammer can be monitored in the 3D plane by a node relative to the observer's local coordinates. The history of the movement of the jammer can be used to predict the possible movement of the jammer in the following stage. A node periodically senses the wireless environment and detects the Direction of Arrival (DoA) of the jammer's signal at a time interval of τ. Figure 5a provides an example of measuring DoA in 3D space. At each interval (k), DoA of the jammer is measured in terms of (θ_k, ϕ_k) *w.r.t.*



(a) Schema for estimation in 3D

(b) 2D representation

(c) Actual beam null

Figure 5: An example of tracking the jammer and creation of beam null

local coordinate system of the observing node. If the received jamming signal is above an interference threshold, then it is identified as a jamming attack. Let us assume $x_k(\theta, \phi)$ is the position of the jammer at interval k. If we take the projection of line connecting x_k and the origin of the local coordinate on XY plane, then the angle between the projection and X-axis is denoted as θ_k. The angle between the projection and the actual line is denoted as ϕ_k.

Figure 5b provides a simplified 2-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 measurement interval 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 the interval $k+1$. As noise is incurred in measuring the DoA, a node should use a confidence region for estimating the movement of the jammer. At interval (k), a confidence region for DoA of the jammer at interval k can be determined along with a confidence region for estimated location of the jammer at interval $k+1$. A region can be mapped that includes the possible trajectory of the jammer in between observations k and $k+1$. Figure 5b represents the cross-section of null border in terms of θ and ϕ whereas Figure 5c provides an illustration of the created null in 3D. The signal from a node whose DoA falls in the null region will have a negligible 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

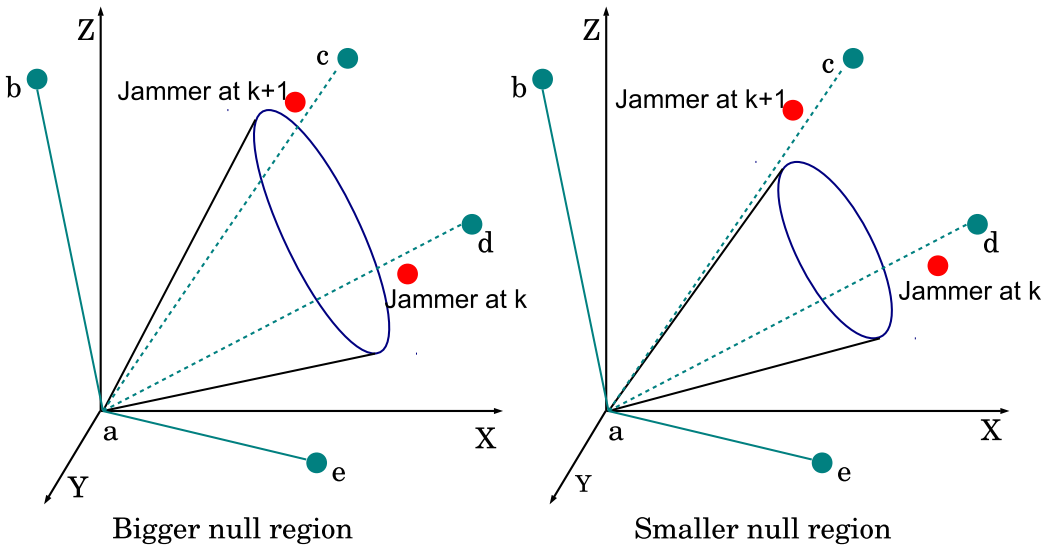


Figure 6: Trade off of having wider null region

work is to determine the optimal beamnull that minimizes the number of link failures due to the beamnull and maximizes the confidence of avoiding the attack.

3.3. Problem Statement

Figure 6 illustrates the effect of beamnulling 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 the 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 6 presents two scenarios of the beam nulling. If a uses a bigger null, the probability of the jammer movement being inside the null increases. Having a bigger null increases the number of neighbors to be shadowed in the null, which in turn causes the link with the shadowed neighbors

to be disconnected. With a bigger null, a can maintain links with b and e whereas links to c and d fail. With a smaller 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 of 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 the cost of disabling some links that are not affected by the jammer. 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 beam null region.

4. Proposed solution technique

This section describes the proposed technique of observing jammer's movement, predicting its next movement and obtaining an optimal beam null. Furthermore, we extend our study to derive optimal beam nulls in the presence of multiple jammers.

4.1. Tracking movement of a jammer with noisy observation

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 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 of

$$\mathbf{x_{k+1}} = \mathbf{F_k x_k} + \mathbf{w_k} \tag{1}$$

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

$$\mathbf{z_k} = \mathbf{H_k x_k} + \mathbf{v_k} \tag{2}$$

Where,

$\mathbf{x}_k$	=	$(n \times 1)$ state vector at time t_k
$\mathbf{F}_k$	=	$(n \times n)$ state transition matrix relating x_k to x_{k+1}
$\mathbf{w}_k$	=	$(n \times 1)$ input white noise with known covariance
$\mathbf{z}_k$	=	$(m \times 1)$ observation or measurement at time t_k
$\mathbf{H}_k$	=	$(m \times n)$ observation matrix giving the noiseless connection between the measurement and the state vector
$\mathbf{v}_k$	=	$(m \times 1)$ white sequence measurement error with known covariance

The filter assumes that $\mathbf{F_k}, \mathbf{H_k}$, and the covariance matrix describing $\mathbf{w_k}, \mathbf{v_k}$ are known. The covariance matrices for the $\mathbf{w_k}$ and $\mathbf{v_k}$ are given by

$$\mathbf{E}[\mathbf{w_k}\mathbf{w_i^T}] = \begin{cases} \mathbf{Q}_k & , \quad i = k \\ 0 & , \quad i \neq k \end{cases} \tag{3}$$

$$\mathbf{E}[\mathbf{v_k}\mathbf{v_i^T}] = \begin{cases} \mathbf{R}_k & , \quad i = k \\ 0 & , \quad i \neq k \end{cases} \tag{4}$$

$$\mathbf{E}[\mathbf{w_k}\mathbf{v_i^T}] = 0 \quad , \forall k, \forall i \tag{5}$$

Figure 7 provides a representation of the Kalman filter process [22]. It 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. We can write eq. 1 as

$$\begin{bmatrix} \theta(k+1) \\ \dot{\theta}(k+1) \\ \phi(k+1) \\ \dot{\phi}(k+1) \end{bmatrix} = \begin{bmatrix} 1 & \tau & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \tau \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \theta(k) \\ \dot{\theta}(k) \\ \phi(k) \\ \dot{\phi}(k) \end{bmatrix} + \mathbf{w_k} \tag{6}$$

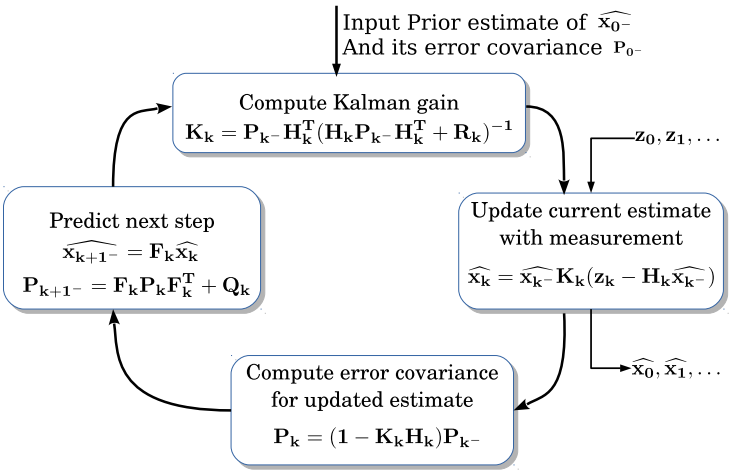


Figure 7: Kalman Filter iteration [22]

A node can observe only the position of the jammer in terms of θ and ϕ. Then eq. 2 can be written as

$$\begin{bmatrix} z_\theta(k+1) \\ z_\phi(k+1) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \theta(k) \\ \dot{\theta}(k) \\ \phi(k) \\ \dot{\phi}(k) \end{bmatrix} + \mathbf{v_k} \tag{7}$$

As the error in DoA measurement or noise in the process follows a Gaussian distribution, we can consider,

$$\mathbf{w_k} \sim \mathcal{N}(0, \mathbf{Q}) \tag{8}$$
$$\mathbf{v_k} \sim \mathcal{N}(0, \mathbf{R}) \tag{9}$$

where $\mathbf{v_k}$ is the DoA estimation error while $\mathbf{w_k}$ is the error or displacement of the jammer from it's intended position.

4.2. Constructing beam null

At each step k, a node observes the position of a jammer in terms of θ and ϕ. Note that, we are not interested about the distance of the jammer from the observer node and thus, do not construct the beam null using (x, y, z) coordinates. 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 estimate and predicted position respectively.

$$\begin{bmatrix}\theta_A \\ \phi_A\end{bmatrix} = \begin{bmatrix}1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0\end{bmatrix} \widehat{\mathbf{x_k}} \tag{10}$$

$$\begin{bmatrix}\theta_B \\ \phi_B\end{bmatrix} = \begin{bmatrix}1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0\end{bmatrix} \widehat{\mathbf{x_{k+1^-}}} \tag{11}$$

Two confidence regions are determined that enforce 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,

$$r_A = \frac{s}{2}\sqrt{\max(cov_k(\theta,\theta), cov_k(\phi,\phi))} \tag{12}$$

$$r_B = \frac{s}{2}\sqrt{\max(cov_{k+1^-}(\theta,\theta), cov_{k+1^-}(\phi,\phi))} \tag{13}$$

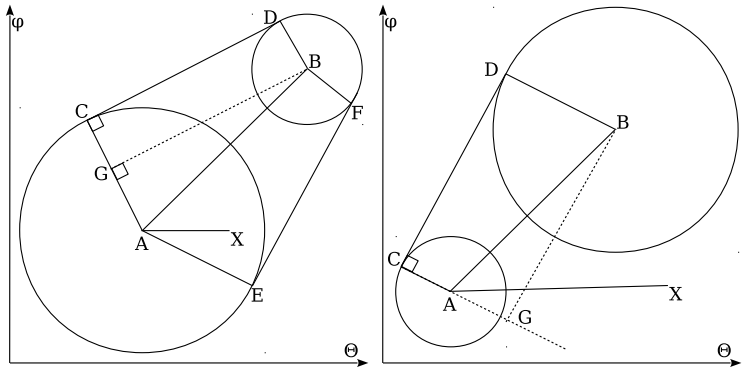
The beam null contains two circles and the region where the jammer may be in between two measurement updates. It is estimated that jammer is inside $\bigcirc_A$ at step k and predicted to be inside $\bigcirc_B$ at $k+1$. If the jammer moves straight in between the two measurement intervals, then it can only move in the area covered by the two circles and their outer tangents. If one circle covers 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:

$$\max(r_A, r_B) > \min(r_A, r_B) + \sqrt{(\theta_A - \theta_B)^2 + (\phi_A - \phi_B)^2} \tag{14}$$

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

4.2.1. 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



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

Figure 8: Schema to obtain common outer tangents

have a look at representation of θ and ϕ on a 2D plane as illustrated in Figure 8a. 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 8a 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:

$$\angle GAB = \cos^{-1} \frac{AG}{AB} = \cos^{-1} \frac{r_A - r_B}{\sqrt{(\theta_A - \theta_B)^2 + (\phi_A - \phi_B)^2}} \tag{15}$$

Let us draw a line AX that is parallel to θ axis. We can calculate:

$$\angle XAB = \tan^{-1} \frac{\phi_B - \phi_A}{\theta_B - \theta_A} \tag{16}$$

Note that tan^{-1} provides the same angle for the 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:

$$\theta_C = \theta_A + r_A \cos(\angle GAB + \angle XAB) \tag{17}$$
$$\phi_C = \phi_A + r_A \sin(\angle GAB + \angle XAB) \tag{18}$$
$$\theta_D = \theta_B + r_B \cos(\angle GAB + \angle XAB) \tag{19}$$
$$\phi_D = \phi_B + r_B \sin(\angle GAB + \angle XAB) \tag{20}$$

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:

$$\theta_E = \theta_A + r_A \cos(\angle XAB - \angle GAB) \tag{21}$$
$$\phi_E = \phi_A + r_A \sin(\angle XAB - \angle GAB) \tag{22}$$
$$\theta_F = \theta_B + r_B \cos(\angle XAB - \angle GAB) \tag{23}$$
$$\phi_F = \phi_B + r_B \sin(\angle XAB - \angle GAB) \tag{24}$$

It can be easily proven that these equations also hold true for the case of $r_A < r_B$ as demonstrated in Figure 8b. Regardless of radii of the circles, we can use the above equation.

4.2.2. 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 9 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:

$$\varrho_j = \varrho_{\bigcirc_A} \vee \varrho_{\bigcirc_B} \vee \varrho_{\triangle_{CDE}} \vee \varrho_{\triangle_{DEF}} \tag{25}$$

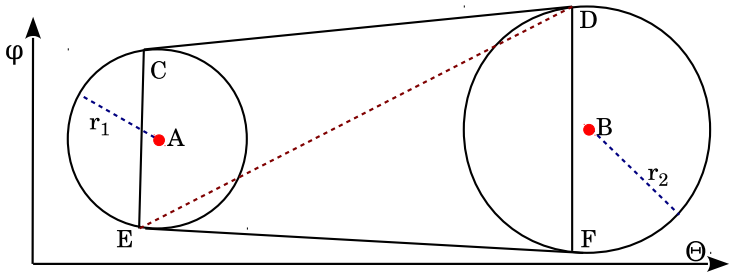


Figure 9: Cross section of beam null in (θ, ϕ)

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 eq. 14, we have to check only for the bigger circle. Conditions for being inside circle A and B are:

$$\varrho_{\bigcirc_A} = \sqrt{(\theta_j - \theta_A)^2 + (\phi_j - \phi_A)^2} < r_A \tag{26}$$

$$\varrho_{\bigcirc_B} = \sqrt{(\theta_j - \theta_B)^2 + (\phi_j - \phi_B)^2} < r_B \tag{27}$$

Let us again look at the θ, ϕ representation on a 2D plane as in Figure 9. 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:

$$\mathfrak{A}(\triangle_{CDE}) = \left| \frac{\theta_C(\phi_D - \phi_E) + \theta_D(\phi_E - \phi_C) + \theta_E(\phi_C - \phi_D)}{2} \right| \tag{28}$$

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

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

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

4.3. Optimization Goal

Let $\mathbb{N}$ be the set of nodes in a 3D mesh network. Let $j \in \mathbb{N}$ be a 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

$$\mathbb{P}(\text{ij fails}) = \begin{cases} 1 & if \quad \varrho_j = True \\ \mathbb{P}(\text{i is jammed}) & otherwise \end{cases} \tag{29}$$

$$= \varrho_j + (1 - \varrho_j)\mathbb{P}(\text{node i is jammed}) \tag{30}$$

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 eq. 12 and eq. 13 for calculating circle diameters,

$$\mathbb{P}(\text{jammer in the estimated region}) \approx 1 - \frac{1}{s_i^2} \tag{31}$$

For the optimization purpose, we can consider,

$$\mathbb{P}(\text{node i being jammed}) = \frac{1}{s_i^2} \tag{32}$$

Thus, we can write,

$$\mathbb{P}(\text{ij fails}) = \varrho_j + \frac{(1 - \varrho_j)}{s_i^2}) \tag{33}$$

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 in the best interest of the network that these links are safeguarded against failure. Let $w_{ij};\ i, j \in \mathbb{N}$ denote the weight for a link between i and j. If all links are equally important for a network then $w_{ij} = 1; \forall i, j$. Thus, the optimization problem becomes:

$$maximize \quad \sum_{j \in \mathbb{N}} w_{i,j} \left(\varrho_j + \frac{(1 - \varrho_j)}{s_i^2}) \right) \tag{34}$$

The lower value of s_i reduces the beam null region that in effect reduces the number of deactivated links. However, it comes at 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.

4.4. Algorithm

Each node $i \in \mathbb{N}$ follows Algorithm 1 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 $z_\theta(k), z_\phi(k)$ 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 value of s_i is determined. The optimal s_i is then used to determine the beam null. Node i uses this beam null until the next observation at step $k+1$. At each step, the Kalman filter algorithm is run which takes a negligible amount of processing time as it deals with only matrix multiplications. The maximization of s_i also runs in the order of $\mathbb{N}$ as it checks with the neighbor of j 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 microseconds [19, 20].

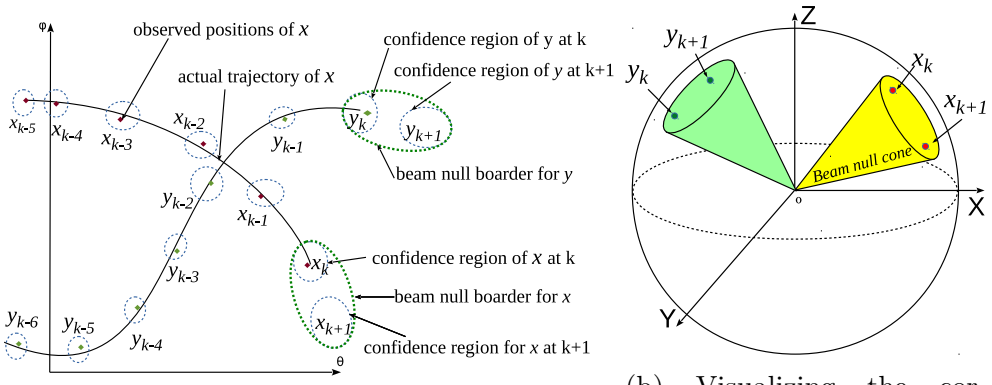
Algorithm 1: Algorithm for beam null at step k

```
measure angular position of jammer z_θ(k), z_φ(k)
if not enough observation then
    create beam null centering z_θ(k), z_φ(k) with a cone radius of r_th
else
    Calculate x_k, P_k, x_{k+1^-}, P_{k+1^-}
    Determine optimal s_i using eq. 41
    Create the beam null using optimal value of s_i
```

4.5. Defense against Multiple Jammers

So far we have discussed the calculation of a beam null for a single moving jammer. A node can defend itself against multiple jammers by adapting its gain pattern to include multiple nullified regions [57, 58]. However, determining the optimized beam nulls to defend multiple jammers is not very simple. It is not just creating multiple independent nulls for each jammer. The nodes need to consider joint probability of link failure for link shadowing and probability of being attacked.

In this framework, a node maintains separate Kalman state vectors ($\mathbf{x}, \mathbf{F}, \mathbf{w}, \mathbf{z}, \mathbf{H}$) for each jammer. The node monitors the DoA jammer $z^v_\theta(k), z^v_\phi(k)$ at an interval of τ. Now, the node uses different standard deviation s^v to create



(a) Movement of the jammer and corresponding DoA measurements in (θ, ϕ) plane

(b) Visualizing the corresponding beamnull in (x, y, z) plane

Figure 10: Creating beam null for multiple jammer.

the beam null border described in the earlier section. Figure 10 illustrates the scenario. Figure 10a presents the observation of jammer's DoA in (θ, ϕ) plane. The solid line is the actual trajectory of the jammer in reference to the observer. Dotted circles are associated with the DoA observations to indicate the confidence region determined by the observer where the jammer is at that moment. The radii of these circles are determined by the coefficient matrices and the s^v used. In this figure, we also illustrate the beamnull border which incorporates the circle for current position and the circle for the next predicted position. The actual beamnull representation in 3D plane (x, y, z) is represented in Figure 10b. Here we do not show the previous observations of the jammers. Instead, we just show the two beam nulls corresponding to Figure 10a. The lower value of s^v reduces the beam null region that in effect reduces the number of deactivated links. However, it comes at a cost that there is a higher probability of a node being jammed that, in effect, deactivates all links. Again, a very high value of s^v increases the number of deactivated links.

For multiple beamnull, with known beam nulls, a node 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 can be obtained by modifying eq. 36 as:

$$\mathbb{P}_{(\text{ij fails})} = \begin{cases} 1 & if \quad \exists_{v \in V} \quad \varrho_j^v = True \\ \mathbb{P}(\text{i is jammed}) & otherwise \end{cases} \tag{35}$$

$$= (1 - \mathbb{P}(\text{i is jammed})) \cup \varrho_j^v + \mathbb{P}(\text{i jammed}) \tag{36}$$

where ϱ_j^v checks if the link ij falls in the beamnull created for the jammer v.

The probability of a node successfully avoiding jamming is same as the probability that the jammers stay 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^v standard deviation in eq. 12 and eq. 13 for calculating circle diameters for jammer v,

$$\mathbb{P}(\text{jammer } v \text{ stays in beamnull}) \approx 1 - \frac{1}{(s_i^v)^2} \tag{37}$$

$$\mathbb{P}(\text{all the jammers stay in beamnulls}) \approx \prod_{v \in V} (1 - \frac{1}{(s_i^v)^2}) \tag{38}$$

For the optimization purpose, we can consider,

$$\mathbb{P}(\text{node i being jammed}) = 1 - \prod_{v \in V} (1 - \frac{1}{(s_i^v)^2}) \tag{39}$$

Thus, we can write,

$$\mathbb{P}(\text{ij fails}) = \cup \varrho_j^v + (1 - \cup \varrho_j^v)(1 - \prod_{v \in V} (1 - \frac{1}{(s_i^v)^2}))) \tag{40}$$

Considering $w_{ij};\ i, j \in \mathbb{N}$ denotes the weight for a link between i and j, If all links are equally important for a network then $w_{ij} = 1; \forall i, j$. Thus, the optimization problem becomes:

$$maximize \quad \sum_{j \in \mathbb{N}} w_{i,j} \left(1 - \cup \varrho_j^v - (1 - \cup \varrho_j^v)(1 - \prod_{v \in V} (1 - \frac{1}{(s_i^v)^2}))) \right) \tag{41}$$

The maximization problem stated above is a convex optimization problem that computes optimal s_i^v for each of the jammers at every step. The basic idea is it is sometimes better to choose lower confidence beam null for a jammer if that intended beamnull result in too many link failure.

5. Simulation results and discussion

The proposed mechanism is evaluated in two methods. First, a custom built simulator is used to analyze the performance regarding network connectivity, and then to observe the performance of upper layer protocol in more detail, we simulated the network using ns3.

5.1. Simulation setup

We built a custom simulator using python scripting language. It keeps track of the performance at every tick interval of value τ seconds. The parameters of the simulation are listed in Table 1. All nodes use Algorithm 1 individually to create the desired beam null. The nodes run the algorithm at every tick interval after measuring the local DoA of the jammer. The goal of this work is to evaluate the efficiency of the beam nulling mechanism. Hence, we assume that the nodes are capable of detecting and measuring the DoA of the jammer through the mechanisms proposed in [23, 24, 25, 44, 45, 46]. Next, after running the algorithm, each node enters the communication phase, in which nodes outside the beam nulled region can communicate. In the meantime, if the attacker moves outside the nulled region of a node, then the node is considered jammed, preventing it to communicate with any previous neighbor. In the simulation, the nodes are positioned following a uniform random distribution. Same positions are used to compare different mobility models. The received power is calculated using the free space path loss model. A link between nodes is active only if both nodes are inside each other's communication range, and only if none of them are jammed. If nodes fall inside their neighbors' null region, then the link is also considered broken. We use three different 3D mobility models for the jammer [59]: Random Walk, Random Direction, and Gauss-Markov; and compare the performance in each case. A sample trajectory of each model is shown in Figure 11. The performance of our proposed model is compared among these mobility models, and the results are presented in Section 5.5

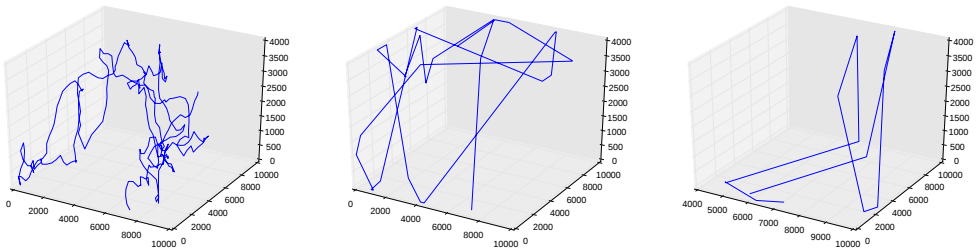
5.2. Performance Metrics

We use four performance parameters:

i) *Average number of nodes jammed* defines the average number of nodes that are jammed during a simulation.

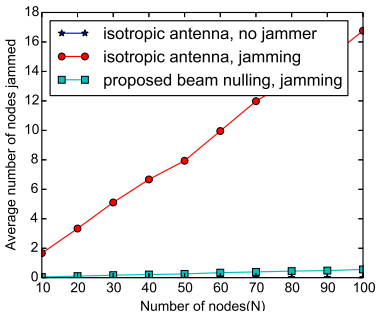
Table 1: Simulation Parameters

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
Jammer' 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

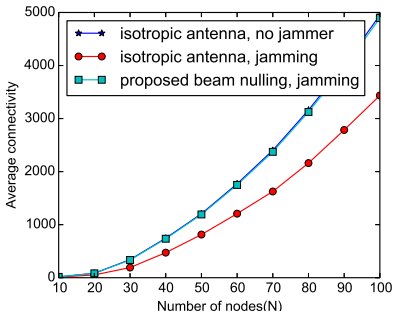


(a) Gauss Markov (b) Random Direction (c) Random Walk

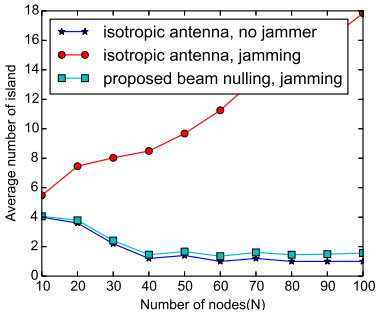
Figure 11: Trace of different mobility models in 3D



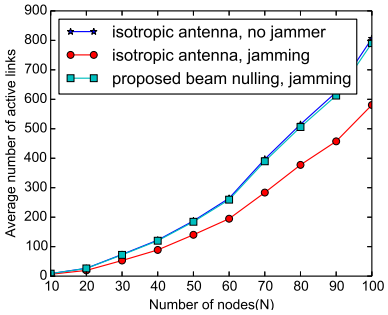
(a) Average number of nodes attacked



(b) Average number of connectivity



(c) Average number of islands created



(d) Average number of active links

Figure 12: Simulation results

ii) 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.

iii) The third 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.

iv) 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. The simulator counts the number of islands present in the network at each tick. If a network is completely connected, the number of islands is 1. The more islands, the more disrupted a network is.

5.3. Simulation varying number of nodes

To correctly evaluate the performance of the proposed scheme, two benchmark scenarios are considered. The first being *isotropic antenna without jammer*, where all nodes use isotropic antennas for communication in the absence of any jammer. The second scenario is the *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 to avoid the jammer.

Figure 12a depicts the comparison of the average number of attacked nodes. We obtain the result for various node densities. As the simulation area is fixed, the number of nodes represented in x-axis reveals the node density. Nodes with an isotropic antenna are vulnerable to jamming. As the density of nodes increases, more nodes are attacked as can be seen in the figure. The proposed mechanism uses adaptive nulling and avoids jamming attacks. 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 attacked nodes close to the ideal case of the no jammer

scenario. The results show that with the proposed mechanism, a network can decrease the average number of jammed nodes up to 96.65%.

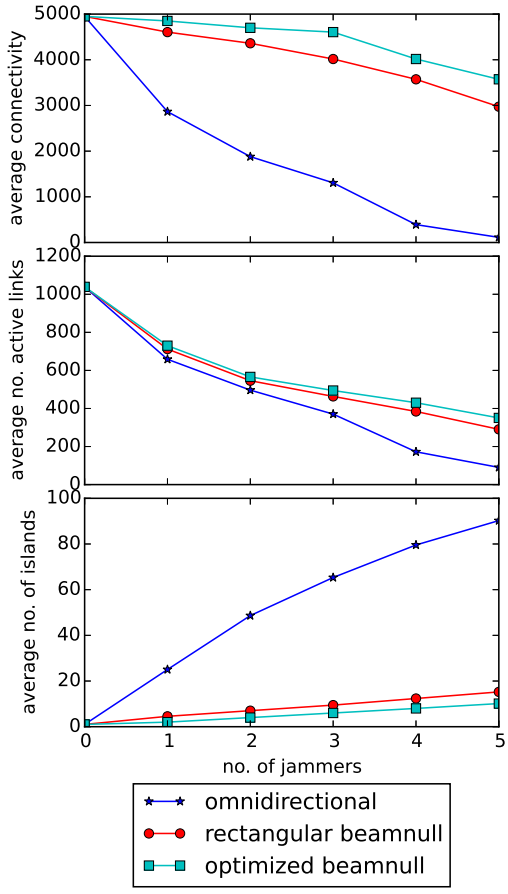


Figure 13: Results with multiple jammers

Figure 12b provides the performance of three scenarios regarding average connectivity. At each tick, the simulator calculates the connectivity of the network. For a fully connected network with n nodes, the connectivity should indicate $\frac{n(n-1)}{2}$. For, a fully connected network with 100 nodes, the value should be 4950, which can be seen in the plot. It can be observed that with our proposed scheme, the network remains almost unaffected in terms of connectivity, as the connectivity is close to the benchmark case of no jammer. The plot reveals that with the proposed scheme, a network can increase its connectivity to 42.47% in the presence of a jammer.

In Figure 12c, the average number of islands are represented. For a very sparse network, where the number of nodes in the network is very small, the network is not well connected. In these cases, the network is not fully connected, and the network is divided into more than one islands. Even for the benchmark case of no jammer, there can exist more than one island. Multiple simulations with the same number of nodes (N) are run with different random node positions are run, 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 1 even for a high value of N. It can be observed that with our proposed algorithm, the network can keep the number of islands very close to the scenario of no jammer. The simulation reveals that with proposed beam nulling method, the

number of islands can be decreased by 91.21%.

Figure 12d 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. With the proposed mechanism, a network can retain 36.14% of its links that are jammed. 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 as discussed in earlier. This proves that the proposed scheme successfully maintains the communication in the jammed region.

5.4. Simulation with multiple jammers

We simulated the network with multiple jammers to illustrate the behavior of the adaptive beam nulling method as described in Section 4.5. The simulated network consists of 100 nodes, the rest of the parameters are kept same as before as listed in Table 1. In Figure 13 we plot the simulation results. Simulations are performed for a different number of jammers in between 0 and 5, where 0 represents the case of no jammer as a benchmark of best case scenario.

In our earlier work, we have proposed a framework that creates beam nulls whose borders are defined by lower and higher cutoffs in θ and ϕ direction. We call this framework as rectangular beamnull creation [60, 21].

Results show that using the optimized beam nulling improves the network connectivity when compared with the same case with the omnidirectional antenna. In all the observed results, we see improvement from our earlier framework that creates rectangular beamnull.

5.5. Simulation with upper layer protocols

To observe the effect the proposed mechanism may cause on the network as a whole, we simulate the new approach in a full stack simulator, namely ns-3 [61]. Ns-3 is a discrete event simulator that provides reliable results when testing complex network configurations with all network layers protocols integrated. We focus the simulations on the interoperability of the proposed framework with two mesh network routing protocols: Ad hoc On-demand Distance Vector (AODV), and Destination-Sequenced Distance Vector (DSDV). In both scenarios we consider the MAC layer to be the IEEE 802.11b standard. AODV can be considered a hybrid protocol as it has both proactive and reactive characteristics. It maintains a route for as long as possible, but it only discovers a new route if there is data to be sent. This

Table 2: Simulation Parameters for ns-3

Parameter	Value
Number of nodes	100
Tick interval	50 ms
Simulation time	500 s
Transport layer protocol	TCP
Dimension	$10 \times 10 \times 10\ km^3$
Number of sources	10
Number of destinations	10
MAC protocol	IEEE 802.11b
Receiver Sensitivity	-78 dBm
Propagation loss model	Friis free space propagation
Data rate	1 Mbps

scheme may cause some delay if there is data ready to be sent but no route has been discovered yet [62]. DSDV on the other hand proactively maintains the routing table updated, regardless if there is data to be sent or not. It may be unfit to use DSDV is highly mobile ad hoc networks since it requires new sequence numbers before the entire topology converges [63].

We use a proof-of-concept antenna model to simulate our proposed mechanism [64]. The model has only a few parameters: beamwidth, gain inside the beamwidth, gain outside the beamwidth and orientation. For our purpose, the beamwidth represents the nulled region, and the gain inside it is set to $-60\ dB$, while the gain outside is $0\ dB$. The orientation is defined by the DoA of the jamming signal. The ns-3 simulation parameters are listed in Table 2, while Table 3 and Table 5 provide default values for AODV and DSDV, respectively. The data traffic is generated by 10 random nodes and received by another 10 random nodes (i.e. one source has only one destination). We define three different types of application, which generate data in different data-rates. The probability of a node to select a certain type of traffic, and the respective data-rate it generates, are defined in Table 4.

Once again we compare different mobility models for the jammer. We measure four performance metrics: *throughput, mean delay, mean hop counts*, and *packet delivery ratio (PDR)*. Their definition is as follows:

- **Throughput**: the number of bytes received by a node in the application layer.

Table 3: Parameters for Simulating AODV

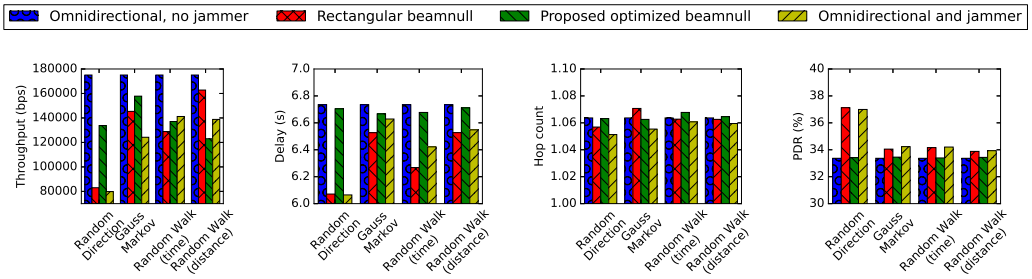
Parameters	Values
Hello interval	1 s
RREQ retries	2
RREQ rate limit	10 per second
RERR rate limit	10 per second
Node traversal time	40 ms
Next hop wait	50 ms
Active route timeout	3 s
Net diameter	35
Max queue length	64 packets
Max queue time	30 s
Allowed hello loss	2
Enable hello	TRUE
Enable broadcast	TRUE

Table 4: Application layer parameters

Application	Bytes generated	Probability
Text	10000	0.6
Image	500000	0.3
Video	5000000	0.1

Table 5: Parameters for Smulation of DSDV

Parameters	Value
Periodic update interval	15 s
Max queue length	500 packets
Max queue time	30 s
Max queue per destination	5packets



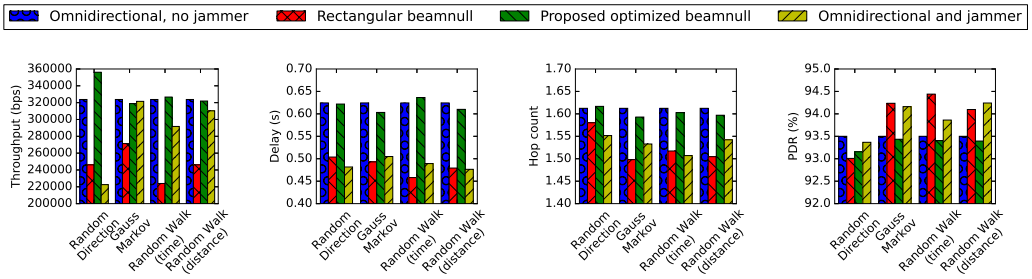
(a) Throughput (b) Delay (c) Hop counts (d) PDR

Figure 14: Simulation results with AODV as routing protocol

- **Mean delay**: average time all packets (data and control) take to arrive at the destination.
- **Mean hop count**: average number of hops all packets take to reach the destination.
- **PDR**: the ratio between the number of packets received by the destination nodes to the number of packets sent by the source nodes.

As mentioned before, we compare this beamnulling framework with our previous work on the same subject area [60, 21], where the proposed approach estimates the speed and direction of the jammer based on the previous measurements. In that work, the beam null has a rectangular border in the (θ, ϕ) plane. We compare both beam nulling models with two benchmark scenarios. One is considered the best case scenario, where nodes use an omnidirectional antenna, and there is no jammer to disrupt the system. The second is the worst case, where nodes use omnidirectional antennas, but there is a jammer, meaning nodes cannot adapt to avoid the jamming signal.

The results for AODV and DSDV routing protocols are shown in Figure 14 and Figure 15 respectively. The network connectivity for both beamnulliung mechanisms is slightly affected by the mobility model of the jammer. The average throughput plots show the improvement of the new framework compared to our earlier work with rectangular beamnull. Overall the plots show some fluctuation when varying the mobility models. However, both beamnulling approaches improve the performance most of the times or provide close to the best scenario performances (note that the plots were zoomed



(a) Throughput (b) Delay (c) Hop counts (d) PDR

Figure 15: Simulation results with DSDV as routing protocol

in for easier visualization). Comparing only the two mechanisms, the new approach offers better indicators. The higher throughput provided by our approach in Figure 15 Random Direction is due to the application probability: in average there are more nodes selecting a video application than text or image. The higher PDR for the rectangular beam null is due to the decrease in throughput.

In both AODV and DSDV, the proposed mechanism provides delay close to the best case scenario. The worst-case scenario of the omnidirectional antenna in the presence of a jammer results in several link failures, and consequently some sources or destination nodes are prevented from communicating. Due to the less amount of traffic (from the blocked nodes), the load and congestion on the rest of the nodes are reduced. Thus, the computed delay is also reduced. However, the proposed mechanism allows nodes close to the jamming radius to stay active, but with the loss of some links, some rerouting is necessary to maintain the connectivity. As a result, the average hop count and delay increase. However, the results clearly indicate the benefits of having optimal beamnull over the rectangular beamnull.

We have compared the performance of the proposed mechanism for four different mobility model of the jammer. Random walk and random direction models are more straightforward models and are therefore easy to predict. Random direction model, the direction is not changed for a long time. Gauss-Markov model provides more realistic movement of a UAV and the direction of movement changes more frequently. Thus, larger beamnull is required to keep a jammer inside beamnull. For Gauss-Markov model there is more link failure. As a result, the number of hop count for delivering packets

is increased which also increase the end-to-end delay. Since data is routed using more hops, and the throughput is measured at the MAC layer, higher throughput is observed high for Gauss-Markov model. The performance of the proposed mechanism can be ordered as random direction, random walk (distance), random walk (time) and then Gauss-Markov.

6. Conclusion

A 3D mesh network is vulnerable to mobile jammers. 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 a jammer in the near future. The mechanism uses Kalman filtering to estimate the current and future position of a jammer from a history of noisy DoA measurements of the jammer. 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 increased prediction efficiency. The research is further extended to consider the case for multiple jammers. Here nodes track the movement of jammers and create optimal beam nulls with different confidence for each of the jammer. An optimization function is proposed that determine the optimal confidence to be used to create beamnull for each jammer. Simulation results support the effectiveness of the mechanism where the number of jammed nodes decreases up to 96.65% compared to the legacy isotropic communication under jamming.

7. References

[1] S. Bhunia and S. Sengupta, "Distributed Adaptive Beam Nulling to Mitigate Jamming in 3D UAV Mesh Networks," in *2017 International Conference on Computing, Networking and Communications (ICNC)*, IEEE, 2017.

[2] lker Bekmezci, O. K. Sahingoz, and amil Temel, "Flying ad-hoc networks (fanets): A survey," *Ad Hoc Networks*, vol. 11, no. 3, pp. 1254 – 1270, 2013.

[3] C. Yu, K. G. Shin, and L. Song, "Maximizing communication concurrency via link-layer packet salvaging in mobile ad hoc networks," *IEEE transactions on mobile computing*, vol. 6, no. 4, 2007.

[4] L. Gupta, R. Jain, and G. Vaszkun, "Survey of important issues in uav communication networks," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1123–1152, 2016.

[5] N. H. Motlagh, M. Bagaa, and T. Taleb, "Uav-based iot platform: A crowd surveillance use case," *IEEE Communications Magazine*, vol. 55, no. 2, pp. 128–134, 2017.

[6] `Http://www.firstresponder.gov/SitePages/HomePage/FirstResponder.aspx.`

[7] "Faa makes progress with uas integration." `http://www.faa.gov/news/updates/?newsId=68004`.

[8] "Washoe county test drones for emergency use." `http://www.kolotv.com/home/headlines/Washoe-County-Will-Test-Drones-for-Emergency-Use-305475261.html`.

[9] L. De Filippis, G. Guglieri, and F. Quagliotti, "Path planning strategies for uavs in 3d environments," *Journal of Intelligent & Robotic Systems*, vol. 65, no. 1-4, pp. 247–264, 2012.

[10] R. H. Jhaveri, S. J. Patel, and D. C. Jinwala, "Dos attacks in mobile ad hoc networks: A survey," in *Advanced Computing & Communication Technologies (ACCT), 2012 Second International Conference on*, pp. 535–541, IEEE, 2012.

[11] J. von Mulert, I. Welch, and W. K. Seah, "Security threats and solutions in manets: A case study using {AODV} and {SAODV}," *Journal of Network and Computer Applications*, vol. 35, no. 4, pp. 1249 – 1259, 2012.

[12] M. Li, I. Koutsopoulos, and R. Poovendran, "Optimal jamming attack strategies and network defense policies in wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 9, no. 8, pp. 1119–1133, 2010.

[13] R. Saranyadevi, M. Shobana, and D. Prabakar, "Article: A survey on preventing jamming attacks in wireless communication," *International*

Journal of Computer Applications, vol. 57, pp. 1–3, November 2012. Full text available.

[14] B. D. Van Veen and K. M. Buckley, "Beamforming: A versatile approach to spatial filtering," *IEEE assp magazine*, vol. 5, no. 2, pp. 4–24, 1988.

[15] N. Seifi, J. Zhang, R. W. Heath, T. Svensson, and M. Coldrey, "Coordinated 3d beamforming for interference management in cellular networks," *IEEE Transactions on Wireless Communications*, vol. 13, no. 10, pp. 5396–5410, 2014.

[16] W. C. Cummings, "Anadaptive nulling antenna for military," *Lincoln Laboratory Journal*, vol. 5, no. 2, 1992.

[17] 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, IEEE, 2001.

[18] G. K. Rao and R. S. H. Rao, "Status study on sustainability of satellite communication systems under hostile jamming environment," in *India Conference (INDICON), 2011 Annual IEEE*, pp. 1–7, IEEE, 2011.

[19] M. Willerton and D. Yates, "Imperial College London Enhances Direction Finding and Beamforming Using LabVIEW and NI USRP." `http://sine.ni.com/cs/app/doc/p/id/cs-15016#`.

[20] "Angle of Arrival Detection with NI USRP RIO." `http://forums.ni.com/t5/Software-Defined-Radio/Angle-of-Arrival-Detection-with-NI-USRP-and-LabVIEW/ta-p/3534214`.

[21] S. Bhunia, V. Behzadan, P. A. Regis, and S. Sengupta, "Adaptive beam nulling in multihop ad hoc networks against a jammer in motion," *Computer Networks*, 2016.

[22] R. G. Brown and P. Y. Hwang, *Introduction to Random Signals and Applied Kalman Filtering with Matlab Exercises, 4th Edition*. Wiley, 2012.

[23] J. T. Chiang and Y.-C. Hu, "Cross-layer jamming detection and mitigation in wireless broadcast networks," *IEEE/ACM Transactions on Networking (TON)*, vol. 19, no. 1, pp. 286–298, 2011.

[24] C. Sorrells, L. Qian, and H. Li, "Quickest detection of denial-of-service attacks in cognitive wireless networks," in *Homeland Security (HST), 2012 IEEE Conference on Technologies for*, pp. 580–584, IEEE, 2012.

[25] 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.

[26] Z. Lu, W. Wang, and C. Wang, "Modeling, evaluation and detection of jamming attacks in time-critical wireless applications," *IEEE Transactions on Mobile Computing*, vol. 13, no. 8, pp. 1746–1759, 2014.

[27] X. He, H. Dai, and P. Ning, "Dynamic adaptive anti-jamming via controlled mobility," *IEEE Transactions on Wireless Communications*, vol. 13, no. 8, pp. 4374–4388, 2014.

[28] M. K. Hanawal, M. J. Abdel-Rahman, and M. Krunz, "Joint adaptation of frequency hopping and transmission rate for anti-jamming wireless systems," *IEEE Transactions on Mobile Computing*, vol. 15, no. 9, pp. 2247–2259, 2016.

[29] J.-F. Huang, G.-Y. Chang, and J.-X. Huang, "Anti-jamming rendezvous scheme for cognitive radio networks," *IEEE Transactions on Mobile Computing*, vol. 16, no. 3, pp. 648–661, 2017.

[30] C. Sorrells, P. Potier, L. Qian, and X. Li, "Anomalous spectrum usage attack detection in cognitive radio wireless networks," in *IEEE International Conference on Technologies for Homeland Security (HST), 2011*, pp. 384–389, IEEE, 2011.

[31] S. Bhunia, S. Sengupta, and F. Vazquez-Abad, "Cr-honeynet: A learning & decoy based sustenance mechanism against jamming attack in crn," in *Military Communications Conference (MILCOM), 2014 IEEE*, pp. 1173–1180, IEEE, 2014.

[32] 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.

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

[34] J. M. Becker, *Dynamic beamforming optimization for anti-jamming and hardware fault recovery*. PhD thesis, Carnegie Mellon University, 2014.

[35] L. Lei, X. Rongqing, and L. Gaopeng, "Robust adaptive beamforming based on generalized sidelobe cancellation," in *Radar, 2006. CIE'06. International Conference on*, pp. 1–4, IEEE, 2006.

[36] Y. Xu and Z. Liu, "Noncircularity-rate maximization: a new approach to adaptive blind beamforming," in *WiCom*, pp. 1–4, IEEE, 2009.

[37] S. Chen, L. Hanzo, N. N. Ahmad, and A. Wolfgang, "Adaptive minimum bit error rate beamforming assisted receiver for qpsk wireless communication," *Digital Signal Processing*, vol. 15, no. 6, pp. 545–567, 2005.

[38] R. Haupt and H. Southall, "Experimental adaptive nulling with a genetic algorithm," *MICROWAVE JOURNAL-EUROGLOBAL EDITION-*, vol. 42, pp. 78–89, 1999.

[39] A. Massa, M. Donelli, F. G. De Natale, S. Caorsi, and A. Lommi, "Planar antenna array control with genetic algorithms and adaptive array theory," *Antennas and Propagation, IEEE Transactions on*, vol. 52, no. 11, pp. 2919–2924, 2004.

[40] Y.-j. Lee, J.-W. Seo, J.-K. Ha, and D.-c. Park, "Null steering of linear phased array antenna using genetic algorithm," in *Microwave Conference, 2009. APMC 2009. Asia Pacific*, pp. 2726–2729, IEEE, 2009.

[41] D. J. Sadler, "Planar array design for low ambiguity," in *Antennas & Propagation Conference, 2009. LAPC 2009. Loughborough*, pp. 713–716, IEEE, 2009.

[42] H. Evans, P. Gale, B. Aljibouri, E. Lim, E. Korolkwiwicz, and A. Sambell, "Application of simulated annealing to design of serial feed sequentially rotated 2× 2 antenna array," *Electronics Letters*, vol. 36, no. 24, pp. 1987–1988, 2000.

[43] S. Ram, *A Study of Adaptive Beamforming Techniques Using Smart Antenna For Mobile Communication*. PhD thesis, National Institute of Technology Rourkela, 2007.

[44] H. L. Van Trees, *Detection, estimation, and modulation theory*. John Wiley & Sons, 2004.

[45] M. Jin, G. Liao, and J. Li, "Joint dod and doa estimation for bistatic mimo radar," *Signal Processing*, vol. 89, no. 2, pp. 244–251, 2009.

[46] X. Zhang, L. Xu, L. Xu, and D. Xu, "Direction of departure (dod) and direction of arrival (doa) estimation in mimo radar with reduced-dimension music," *Communications Letters, IEEE*, vol. 14, no. 12, pp. 1161–1163, 2010.

[47] C. Vaidyanathan and K. M. Buckley, "Performance analysis of the mvdr spatial spectrum estimator," *Signal Processing, IEEE Transactions on*, vol. 43, no. 6, pp. 1427–1437, 1995.

[48] 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.

[49] Z. Szalay and L. Nagy, "Target modeling, antenna array design and conventional beamforming algorithms for radar target DOA estimation," in *Transparent Optical Networks (ICTON), 2015 17th International Conference on*, pp. 1–4, July 2015.

[50] N. Tayem, "2D DOA estimation of multiple coherent sources using a new antenna array configuration," in *Signals, Systems and Computers (ASILOMAR), 2012 Conference Record of the Forty Sixth Asilomar Conference on*, pp. 212–216, Nov 2012.

[51] F. Akbari, S. Moghaddam, and V. Vakili, "MUSIC and MVDR DOA estimation algorithms with higher resolution and accuracy," in *Telecommunications (IST), 2010 5th International Symposium on*, pp. 76–81, Dec 2010.

[52] 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.

[53] A. A. Ansari, Z. Hasan, M. J. Alam, K. Mohammad, and A. Siddique, "Performance comparison for omnidirectional and directional mac protocols for ad hoc network,"

[54] J. Niu, R. Zhang, L. Cai, and J. Yuan, "A fully-distributed directional-to-directional MAC protocol for mobile ad hoc networks," in *IEEE ICNC*, pp. 766–770, 2015.

[55] M. S. Ullah, F. N. Nur, and N. N. Moon, "Optimization of wireless ad-hoc networks using an adjacent collaborative directional MAC (ACDM) protocol," *International Journal of Computer Applications*, vol. 114, no. 3, 2015.

[56] A. Nasipuri, S. Ye, J. You, and R. E. Hiromoto, "A MAC protocol for mobile ad hoc networks using directional antennas," in *Wireless Communications and Networking Confernce, 2000. WCNC. 2000 IEEE*, vol. 3, pp. 1214–1219, IEEE, 2000.

[57] W.-D. Wirth, *Radar techniques using array antennas (FEE radar, sonar, navigation & avionics series)*, vol. 10. IET, 2001.

[58] S. Chandran, *Adaptive antenna arrays: trends and applications*. Springer Science & Business Media, 2013.

[59] P. A. Regis, S. Bhunia, and S. Sengupta, "Implementation of 3d obstacle compliant mobility models for uav networks in ns-3," in *Proceedings of the Workshop on Ns-3*, WNS3 '16, (New York, NY, USA), pp. 124–131, ACM, 2016.

[60] S. Bhunia, V. Behzadan, P. A. Regis, and S. Sengupta, "Performance of adaptive beam nulling in multihop ad-hoc networks under jamming," in *IEEE CSS*, pp. 1236–1241, IEEE, 2015.

[61] ns-3 Consortium, "Network simulator, ns-3." https://www.nsnam.org/.

[62] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc on-demand distance vector (aodv) routing," tech. rep., 2003.

[63] C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," in *SIGCOMM*, (New York, NY, USA), pp. 234–244, ACM, 1994.

[64] "Directional antenna patch for ns-3." https://codereview.appspot.com/6620057/#ps1.