Computer Networks 109 (2016) 50–66

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

# Adaptive beam nulling in multihop ad hoc networks against a jammer in motion☆,☆☆

Suman Bhunia*, Vahid Behzadan, Paulo Alexandre Regis, Shamik Sengupta

*Dept. of Computer Science and Engineering, University of Nevada, Reno, USA 89557, United States*

ARTICLE INFO

*Article history:*
Received 12 May 2016
Accepted 22 June 2016
Available online 1 July 2016

*Keywords:*
Beam nulling
Multihop
Ad hoc
Anti-Jamming
Mobility
VANET
FANET
3D mesh
Ns-3

ABSTRACT

In multihop ad hoc networks, a jammer can drastically disrupt the flow of information by intentionally interfering with links between a subset of nodes. The impact of such attacks can escalate when the jammer is moving. As a countermeasure for such attacks, adaptive beam-forming techniques can be employed for spatial filtering of the jamming signal. This paper investigates the performance of adaptive beam nulling as a mitigation technique against jamming attacks in multihop ad hoc networks. Considering a moving jammer, a distributed beam nulling framework is proposed. The framework uses periodic measurements of the RF environment to detect direction of arrival (DoA) of jamming signal and suppresses the signals arriving from the current and predicted locations of the jammer. Also, in the calculation of the nulled region, this framework considers and counters the effects of randomness in the mobility of the jammer, as well as errors in beam nulling and DoA measurements. Survivability of links and connectivity in such scenarios are studied by simulating various node distributions and different mobility patterns of the attacker. Also, the impact of errors in the estimation of DoA and beam-forming on the overall network performance is also examined. In comparison with an omnidirectional configuration, results indicate a 57.27% improvement in connectivity under jamming when the proposed framework is applied.



## 1. Introduction

The ecosystem of wireless communications is evolving towards distributed, self-configuring ad hoc architectures. Elimination of the need for central communications infrastructure appeals in many scenarios as it allows seamless and quick deployment of agile networks. Such agility is an essential requirement of many applications, including emergency radio networks in disaster zones, tactical communications, and inter-vehicular networks. Also, commercialization of unmanned aerial vehicles (UAVs) and the subsequent feasibility of multi-UAV missions introduce novel challenges and constraints on their network requirements, which may be adequately satisfied in the ad hoc manner. Following the same trend, the concept of 3D mesh networks is envisioned [2], in which aerial nodes collaborate with ground nodes to allow wider, more dynamic ad hoc deployments while enhancing spectrum utilization by exploiting spatial reuse.

Considering the advantages of ad hoc networking, it is envisioned that this paradigm will play a key role in future mission critical communications. Therefore, ensuring the security and robustness of such networks is essential for such applications. Even though the independence of ad hoc configurations from single points of failure is seen as a merit from the security point of view, their information flow is still susceptible to disruption by interference and jamming. Furthermore, it has been shown that jamming a subset of links in multihop networks is sufficient to incur maximal disruption of the network [3]. Hence, mitigation of jamming attacks is a necessary component of mission-critical ad hoc networks.

Some well-known categories of anti-jamming techniques proposed in the literature are those that utilize specially designed signal coding and modulation, such as Frequency Hopping Spread Spectrum (FHSS) [4] and Direct Sequence Spread Spectrum (DSSS) [5]. The downside associated with this class of techniques is their larger bandwidth requirement. Considering the state of the overcrowded electromagnetic spectrum, this overload can prove to be costly. To preserve the scarce bandwidth, an alternative is to

☆ A preliminary version of this paper appeared in the IEEE International Symposium on Cyberspace Safety and Security (CSS) 2015 [1].

☆☆ This research was supported by NSF CAREER grant CNS #1346600 and CAPES Brazil #13184/13-0.

* Corresponding author.
*E-mail addresses:* sbhunia@nevada.unr.edu, sumanbhunia@gmail.com (S. Bhunia), vbehzadan@unr.edu (V. Behzadan), pregis@nevada.unr.edu (P.A. Regis).
*URL:* http://www.cse.unr.edu/~shamik (S. Sengupta)

http://dx.doi.org/10.1016/j.comnet.2016.06.030
1389-1286/

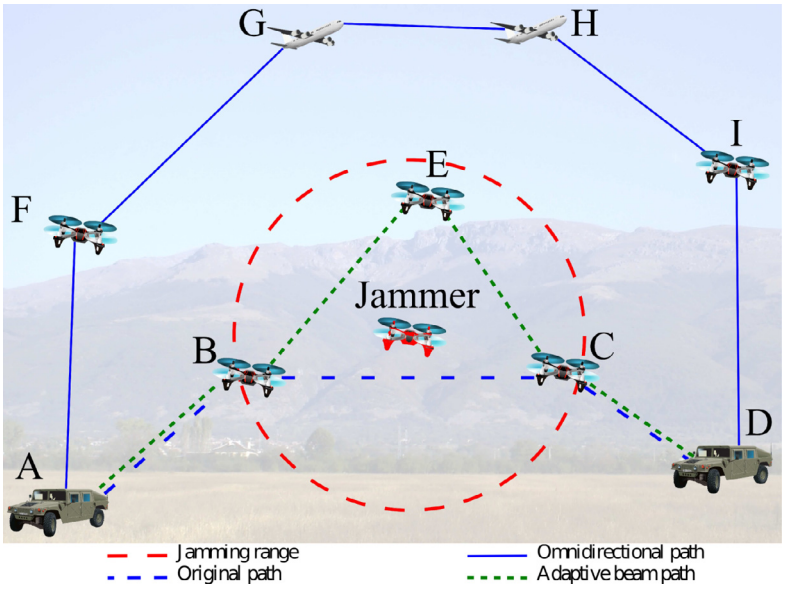



**Fig. 1.** A comparison of routing in omnidirectional vs beam nulling schemes under jamming.

apply Spatial Filtering with beam-forming antenna arrays [6]. This approach exploits the beam-formers' ability to estimate the Direction of Arrival (DoA) of signals. This direction is then used to tailor the beam-former's response, such that the signals originating from sources of interference are suppressed or eliminated. Beamforming antenna systems that implement this mechanism are referred to as Adaptive Nulling Antennas (ANA) [7].

Traditionally, ad hoc configurations assume that omnidirectional antennas are used for communications. In multihop networks, data is routed over multiple hops to reach a destination that is not within direct communication range of the source. By utilizing beam nulling techniques, a node can adapt its radiation pattern to create a null in the direction of interference. This allows maintaining the links which are not affected by the jammer. Fig. 1 provides an example of end-to-end data delivery in an ad hoc network. In the absence of jammer, packets from $A$ to $D$ follow the path $A-B-C-D$ when all nodes employ omnidirectional antennas. In this configuration, the jammer can effectively neutralize nodes $B$, $C$ and $E$. The routing protocol discovers the link failures and reroutes packets through $A-F-G-H-I-D$. This way packets are delivered at the cost of increased end-to-end delay, as well as congestion on link $G-H$.

However, when beam nulling is applied, nodes $B$, $C$ and $E$ can successfully avoid the jammer. Now packets can be delivered through $A-B-E-C-D$. Hence it can be seen that, in the presence of a jammer, adaptive beam nulling is not only capable of maintaining connectivity of the nodes inside the affected region, but also ensures less congestion on the remaining links. The majority of the literature on ANAs rely on the assumption that the jammers are stationary with respect to beamformers [8–12]. However, with the recent expansion and growth of mobile wireless technologies, this assumption does not necessarily hold true. Also, there is a lack of publicly available analysis on the network performance of ad hoc networks utilizing adaptive nulling antennas under jamming.

This paper proposes a completely distributed method for beam nulling in multihop ad hoc networks. In this method, the width and direction of null angles are calculated based on periodic sensing of the jammer's relative position to a node. As this measurement is not continuous, a prediction technique is introduced to estimate jammer's movements until the next sensing phase. Measured and predicted locations are then incorporated in the calculation of a nulled region, which suppresses signals arriving from within its angular span. As signals coming from the neighboring nodes that fall within the null angle are also subject to suppression, the proposed method for calculation of null angle aims to minimize the number of legitimate link failures, while maximizing the confidence of jamming avoidance. The proposed method also takes randomness of the jammer's movements into account by introducing safety buffer zones on both edges of the null angle. To evaluate the effectiveness of this method for both 2D and 3D configurations, physical simulations are performed. The network-level performance of the proposed method is also evaluated by ns-3 simulations, where the impact of different ad hoc routing protocols on the overall performance of this method is investigated. For this purpose, multiple simulations are performed to study the impact of jamming based on *connectivity*, *number of islands*, and *number of surviving links* for different node densities and various mobility models of the jammer. The simulation results show that the proposed framework can achieve up to 57.27% of improvement in connectivity over the omnidirectional antenna case.

The remainder of this paper is organized as follows: Section 2 provides a background study on anti-jamming and beam nulling techniques. The proposed framework of adaptive beam nulling in 2D and 3D spaces are presented in Section 3. Section 4 describes the simulation setup and results. Finally Section 5 concludes the paper.

## 2. Background

This section presents an overview of methods for detection and mitigation of jamming, and reviews the terminology and concepts of adaptive beamforming. This review is not intended as a detailed discussion on beamforming and nulling techniques, but aims to provide the essential basics to equip the reader with enough background for the remainder of this paper. Interested readers are referred to [13] as a comprehensive source on beamforming and nulling antennas.

### 2.1. Jamming Detection and Mitigation

Jamming can be broadly categorized into two types [14]. The first type being physical layer jamming where the attacker jams

the channel of communication by sending strong noise or jamming signal. The second type is link layer jamming, which targets several vulnerabilities present in upper layer protocols. Jamming can also be classified based on the behavior of the jammer [15]. A jammer can be *proactive*, where it continuously emits high power signal on a target frequency. Or it can be a *reactive* jammer and save its resources by intelligently causing heavy interference to specific packets being transmitted by a legitimate node [16–19]. A reactive jammer can scan for transmission of other nodes and then start emitting a high power signal in the spectrum where a transmission is detected. It has also been observed that the success or effect of jamming depends on the transmission power of both jammer and legitimate node, distance between the jammer and target, modulation and coding scheme used by the transmitter, etc. [20].

The flow of information in wireless networks is inherently susceptible to disruption by interference. A jammer can intentionally cause interference in wireless links by transmitting a noise signal on the frequency channels of target links. To detect such jamming attacks, a node must be able to distinguish jamming signals from interference caused by legitimate nodes [17]. For this purpose, cross layer mechanisms have been proposed to estimate the possibility of intentional interference by observing the temporal consistency of certain system parameters such as carrier sensing time, packet delivery ratio and signal strength [21–23].

To defend against jamming attacks, various mitigation techniques have been presented in the literature. Prominent examples of such techniques are spread spectrum, frequency hopping, avoidance of jammed routes, spatial retreat, and deployment of decoys. In spread spectrum techniques such as direct sequence spread spectrum [24], nodes spread their narrow-band signals over a wider spectrum to allow communication under strong interference. Frequency hopping also exploits extra spectrum by switching their operating frequency to evade jamming attacks targeting single channels [4]. In densely populated networks, connectivity can be retained by mapping the jammed nodes and avoiding routes that pass through them [25]. Spatial retreat allows mitigation of attack in mobile networks by relocating the nodes to positions outside of the jammed region [14]. Another class of solutions rely on deployment of honey-pots and decoys to observe the activities of the jammer and infer its attack pattern. This information is then exploited to lure the jammer into targeting decoy nodes [26,27]. The common disadvantage in all of the aforementioned techniques is their overheads in terms of bandwidth requirement, induced delays and number of nodes, which render them unfeasible for some mission critical applications. An alternative technique is spatial filtering of the jamming signal by adaptive beam nulling. As is explained in the following sections, spatial filtering does not require additional bandwidth and redundancy, and hence will add less overhead to the network.

### 2.2. Antenna terminology

Antennas are elements that couple electromagnetic energy between free space and a guiding structure [28]. Antennas may be classified based on how they radiate and receive energy in different directions. The directionality or gain of an antenna in a direction $d = (\theta, \phi)$ is defined as:

$$G(\vec{d}) = \eta \frac{U(\vec{d})}{U_{ave}} \tag{1}$$

Where $\eta$ is the antenna efficiency, $U(\vec{d})$ is the power density in the direction of $d$, and $U_{ave}$ is the average power density in all directions. An isotropic antenna is a hypothetical radiator which has uniform gain in all directions ($U(\vec{d}) = U_{ave}$ for all directions). An omnidirectional antenna is defined as a radiator which has a relatively uniform gain in at least one 2-dimensional plane of directions. A directional antenna is one which radiates more energy in one or more directions compared to other directions. The Radiation Pattern of an antenna is the representation of its gain values in all or a subset of all directions. The pattern typically has a main lobe in which the gain is at its peak, and some side lobes. In this paper, we interchangeably refer to lobes as beams.

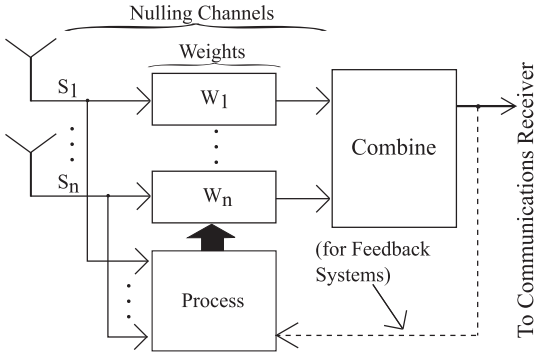



**Fig. 2.** Schema for adaptive nulling antenna.

### 2.3. Estimation of DoA and Adaptive beam-forming

A block representation of a beam-former is depicted in Fig. 2. Signals coming from antenna elements are a mixture of desired signals, interference and noise. The control process of beam-forming determines individual weights of each signal based on an array response optimization method. For non-adaptive beam-formers, weights do not depend on the received signal and can be calculated based on the array response before implementation and use. On the other hand, weights of adaptive beam-formers are functions of the received signals and desired parameters and are calculated during the operation of the antenna. In case of Adaptive Nulling Antenna (ANA) arrays, the weights are chosen so that the array response has nulls in the directions of interference sources.

Estimation of the Direction of Arrival (DoA) of signals using beam-forming antenna arrays is widely studied in the literature and various algorithms have been proposed for this purpose. The common foundation of such algorithms is exploitation of spatial diversity of the elements of antennas arrays. Due to the spatial distribution of antenna elements, a signal incident to the array arrives at each element at different times. This varying delay can consequently be used as the basis of DoA estimation algorithms [29].

In cases where multiple signal sources are present, statistical methods are applied to distinguish and separate signals of different origins. Conventional methods such as Minimum Variance Distortionless Response (MVDR) and root MVDR [30] use beam scanning over a region of interest and measure the received power in different directions. Then, the angles in which measured power peaks are designated as DoA estimations. Another class of estimation methods rely on the statistical modeling of the signal by exploiting known features and structure of the received signals, thereby allowing higher resolution and more accurate estimation compared to conventional methods. MUSIC, Root-MUSIC and ESPIRIT are prominent examples of such algorithms [31]. A thorough review and comparison of DoA estimation methods is presented in [31].

Once the angular direction of the interference signal is determined, the beam-former must calculate its weights such that signals originating from that direction are suppressed or eliminated. Some of the widely studied methods of weight calculation are Dolph–Chebyshev weighting, Least Mean Squares (LMS) and Conjugate Gradient Method (CGM) [32]. In the case of mobile ad hoc networks, where the directions of desired and interference

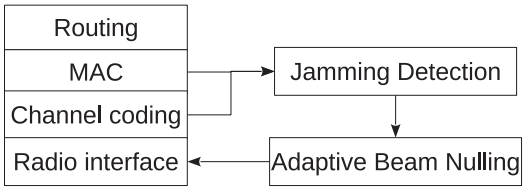



**Fig. 3.** Block representation of proposed mechanism.

signals are not known and vary, Stochastic Search algorithms are applied [33]. Examples of such methods are Gradient Search Based Adaptive algorithms [34–36], Genetic Algorithms [37–39] and Simulated Annealing [40,41]. Detailed overview of beam-forming and weight calculation methods are presented in [33] and [42].

## 3. Methodology

### 3.1. Network and jammer models

The defending network considered in this study is a multi-hop mobile network, comprised of homogeneous nodes that are static relative to each other, and arbitrarily distributed in their operating space. Each node is equipped with an antenna array capable of DoA estimation and beam-forming. There exists a one hop link between the nodes if the signal to noise ratio of the link is above a cut off threshold. This definition considers the jamming signal as a source of noise.

The jamming attack is sought to be carried out by one or more entities that continuously transmit high powered signals to cause interference in the same spectrum as the network. If the Signal to Interference and Noise Ratio (SINR) of inter-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, is remains independent of upper layer protocols such as MAC and routing.

### 3.2. Mitigation of jamming by adaptive beam nulling

The proposed framework uses adaptive beam nulling in order to avoid jamming. Fig. 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, it is assumed that jamming signals are detectable. Interested readers may refer to [21–23] 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 link failure and utilizes an alternative route.

Each node switches to a sensing phase at every time interval of length $\tau$ to measure the DoA of jammer's signal. Since the sensing is not continuous, the history of this periodic measurement is then used in the beam nulling stage to predict the movement of the jammer in the time between the current and next sensing phases. Fig. 4 illustrates an example of DoA measurement in 3D space. In every sensing phase $m$, the jammer's DoA is measured in terms of its azimuth and elevation angles ($\theta^m$, $\phi^m$) in the local coordinate

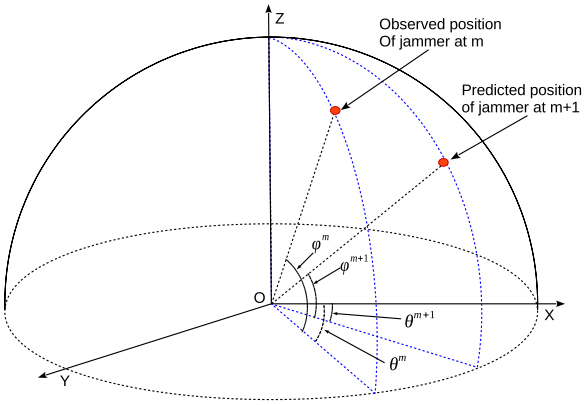



**Fig. 4.** Observation of DoA of jammer in 3D space.

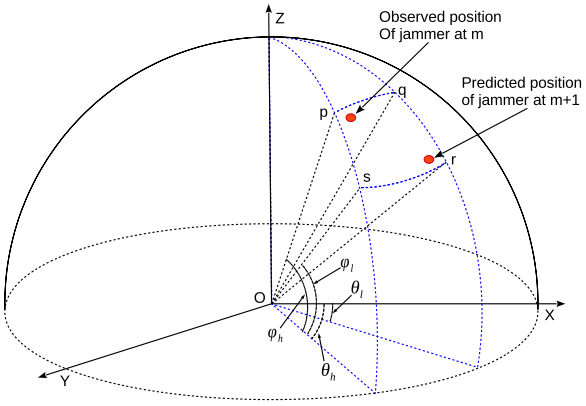



**Fig. 5.** Depiction of 3D beam null.

system of the observing node. Let $x^m$ be the observed position of the jammer in the $m$th sensing phase. The azimuth angle $\theta^m$ is then defined as the angle between the $x$-axis and the projection of the line connecting $x^m$ to the origin on XY plane, and the elevation angle $\phi^m$ is the angle between the origin–$x^m$ line and its projection on XY.

Using the history of DoA measurements, the jammer's trajectory between the $m$th and $m+1$th sensing phases can be efficiently predicted. Consequently, the nulled region is calculated such that it includes the current location of the jammer, as well as its predicted trajectory. Also, since the DoA measurements and predications are both prone to errors, the beam nulling process expands the analytically calculated nulled region by adding a safety zone with the aim of mitigating the effects of errors on nulling the jamming signal. The nulled region can be represented by two boundaries on each of $\theta$ and $\phi$ axes. As is shown in Fig. 5, the nulled region between the node $O$ and the null cross section ($pqrs$) can be defined by its borders represented by their corresponding angles $\theta_l$, $\theta_h$ and $\phi_l$, $\phi_h$.

Transmissions from neighboring nodes that fall within the nulled region of a node are also suppressed. Hence, the width of the nulled region must be determined in such a way that it maximizes the confidence in jamming avoidance while minimizing the number of link failures.

### 3.3. System assumptions

To investigate the effect and feasibility of adaptive beam nulling in practical scenarios, the proposed framework is developed based on the following set of assumptions:

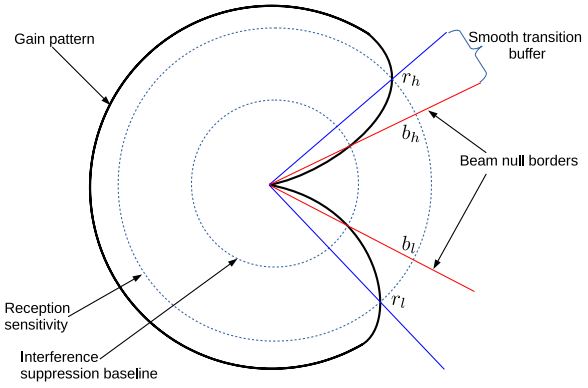



**Fig. 6.** Depiction of null boundary.

(a) Target network is a multihop ad hoc wireless network, with nodes that are static relative to each other.
(b) The jammer is capable of moving.
(c) Affected nodes are able to detect the jamming signal and distinguish it from network's signals.
(d) Communication and sensing phases are asynchronous, i.e. nodes can not measure the jammer's DoA while they are communicating. The sensing phase is triggered at intervals of $\tau$ seconds.
(e) Target nodes are equipped with antenna arrays with an omnidirectional pattern, as well as beam-forming controllers capable of modifying the gains of signals received by each antenna element. Beam-formers are assumed to have sufficient spatial resolution to form the calculated nulled regions with sufficient accuracy [43–45].
(f) After beam-forming, gain of signals arriving from the nulled region is assumed to fall below the sensitivity of the receiver, and therefore is set to be zero.
(g) Time required to form the desired beam is negligible in comparison to the jammer's velocity.
(h) A link between two nodes fails if either of the two nodes are attacked or one of the nodes falls in the beam null of the neighbor.
(i) Introducing a null in the omnidirectional pattern of a beam-forming node may be interpreted as changing the mode of communications to directional transmission, hence necessitating the use of Directional MAC protocols [46]. 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 [47]. This approach therefore eliminates the overheads associated with most directional communications schemes [48–50].
(j) A beam null is a region in the direction in which the antenna gain is below the cutoff threshold of interference, i.e. the signal arriving in the nulled direction will not cause interference on a node. Fig. 6 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 cut-off. Hence, the entire transition region is blind to communications, which is accounted for by 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. 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 $b_l$.

### 3.4. Problem statement

Let us first look at the problem in a 2-dimensional environment. Fig. 7 illustrates the effect of adaptive beam nulling in the presence of a moving jammer. In this scenario, the one hop links between node *A* and its neighbors *B, C, D* and *E* are considered. Node *A* periodically scans for the DoA of the jammer's signal $(\theta^m)$ in intervals of $(\tau)$ seconds. Due to the discontinuous observation of the jammer's DoA, while calculating the null angle, *A* must take into account the movement of the jammer between two consecutive observations. This calculation must include prediction of the jammer's angular velocity by considering its history of movements. As the mobility pattern of a jammer becomes more random, the prediction accuracy of its movements decreases. Therefore, the effect of various mobility models of the jammer on a network of beam nulling nodes can provide a practical measure for efficiency of this scheme.

Node *A* uses a modified beam pattern to communicate with its neighbors until the next sensing period. In Fig. 7a, *A* has a narrower null angle compared to Fig. 7b. With this narrow null angle, *A* can communicate with *B, D* and *E*, whereas with a wider null angle, *A* can communicate only with *B* and *D*. By the next sensing period $m+1$, the jammer moves to a new position, falling outside of the narrower null, which consequently exposes *A* to the jammer. As a result, all of *A*'s links are disrupted. On the other hand, the wider null angle maintains the jammer inside the nulled region for the whole interval. The trade-off for widening the null to cover the jammer's probable movements, is the cost of disabling unaffected links. Hence, another important factor in efficiency of beam nulling is the choice of optimum nulling angle in dynamic scenarios.

The practical limitations of adaptive beam nulling, such as inaccuracy in estimation of DoA, as well as hardware limitations in implementing a desired antenna pattern, lead to introduction of errors in a beam-former's performance. The *measurement error* is the error in DoA estimation. If $(\widehat{\theta_a^m}, \widehat{\phi_a^m})$ is the actual angular position of the jammer with respect to node *A*, but the observed DoA by *A* is $(\theta_a^m, \phi_a^m)$, we can write

$$\begin{bmatrix} \theta_a^m \\ \phi_a^m \end{bmatrix} = \begin{bmatrix} \widehat{\theta_a^m} \\ \widehat{\phi_a^m} \end{bmatrix} + \mathbf{e}_{doa} \tag{2}$$

where $\mathbf{e}_{doa}$ is the measurement noise with known covariance. Similarly, error is incurred while implementing the beam null border is called beam-forming error. Let us say a node calculates a beam null border at $\widehat{b^m}$ and the actual implemented border is at $b^m$, then we can write

$$b^m = \widehat{b^m} + e_{bn} \tag{3}$$

For a sensible study on the efficiency of practical implementation, investigating the impact of system errors in the simulation is of crucial importance. In the subsequent sections we present a framework that determines the beam bull borders dynamically by incorporating the randomness in the mobility of the jammer as well as hardware limitations.

### 3.5. Calculation of null borders in 2D

This section presents a framework for determining the beam null borders in 2D environment. Each node in a multihop ad hoc network uses this method to create a beam null in a distributed

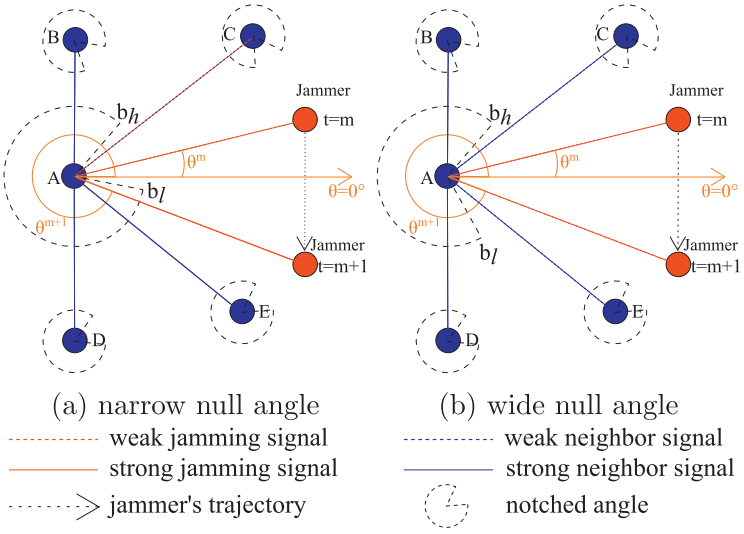



**Fig. 7.** Depiction of the beam nulling principle.

manner according to its own frame of reference. After sensing the presence of a jammer, a node $i$ observes the angular position of the jammer or the angle of attack ($\theta_a^m$) with its frame of reference at every sensing phase $m \in \{1, \ldots, M\}$. Node $i$ then adjusts its beam-form to attenuate the jamming signal and communicate with its neighbors until the next sensing phase $(m+1)$. In Fig. 7, at the $m$th sensing phase, the jammer is sensed at angle $\theta_a^m$. In the next sensing phase $(m+1)$, $i$ senses the jammer at $\theta_a^{m+1}$. Since the jammer is moving, it may cross the null of the beam-form and node $i$ would be affected by the jamming signal. The aim of adaptive beam nulling is to make sure the jammer stays within the nulled region for the entire time between two consecutive sensing phases. Node $i$ calculates the angular velocity of the jammer ($v_a^m$) as:

$$v_a^m = \frac{\theta_a^m - \theta_a^{m-1}}{\tau}$$

Consider $\overline{v_a}$ and $\sigma(v_a)$ as the mean and standard deviation of the velocity ($v_a$) of the jammer, respectively. Node $i$ constructs a beam null using an algorithm that considers the history of jammer's movement. A beam null is defined by two borders: $b_l^m$ and $b_h^m$ which are lower and higher angles respectively. Clearly, $\theta_a^m + \tau\overline{v_a}$ gives the estimated location of the jammer at the $(m+1)^{th}$ slot. Since the actual velocity and direction of the jammer are unknown, the null should be wider in case of sudden change in direction or velocity of the jammer. Change of velocity of the jammer can be estimated with $\sigma(v_a)$. If a jammer changes its direction or velocity, $\sigma(v_a)$ would be high compared to the case when the jammer moves at the same direction with constant velocity. An estimation for the beam null angle can be calculated as:

$$b_h^m = \max(\theta_a^m, \theta_a^m + \tau(\overline{v_a} + \alpha\sigma(v_a))) \quad (4)$$

$$b_l^m = \min(\theta_a^m, \theta_a^m + \tau(\overline{v_a} - \alpha\sigma(v_a))) \quad (5)$$

$$\psi^m = b_l^m - b_l^m \quad (6)$$

Where $\psi^m$ is the null angle constructed at the $m$th sensing phase, and $\alpha$ is a multiplying factor. Note that the higher the value of $\alpha$, higher the null angle is. Now, if the null is wider, chances are more legitimate neighbors fall in nulled region. Node $i$ cannot communicate with its neighbor $j$ if $j$ is in the nulled region of $i$ and vice versa. A higher value of $\alpha$ guarantees a higher probability that the jammer stays in the nulled region until the next sensing period. A very high value of $\alpha$ results in more deactivated links.

In Section 4.3.1 we observe that the system performance is a convex function w.r.t. $\alpha$. Since the jammer's mobility pattern is not completely observable by a node, it should dynamically adjust the value of $\alpha$. To mitigate this effect, we propose a heuristic that dynamically calculates the value of $\alpha$ based on the observed history of jammer's movements.

### 3.6. Heuristic for dynamic $\alpha$

Algorithm 1 presents a heuristic for adapting the value of $\alpha$ at each sensing period $m$. Fig. 8 presents the schema for this procedure. The beam null has been created in the previous sensing period $m-1$. At the $m$th sensing phase, if the jammer stays inside the nulled region ($\psi^{m-1}$), then the node successfully avoids the attack. If the jammer is too close to the null border, $\alpha$ is increased. The algorithm considers a *safety zone* defined by two fences: $f_h$ and $f_l$. We consider a factor $k > 2$ which defines how defensive the network is. The safety fence is a $\psi^{m-1}/k$ deviation from the null border towards the center of the null. Larger values of $k$ increase the probability of the jammer being in the safety zone, which con-

**Algorithm 1:** Heuristics for dynamic $\alpha$.

$\psi^{m-1} \leftarrow b_h^{m-1} - b_l^{m-1}$
$f_l \leftarrow b_l^{m-1} + \frac{\psi^{m-1}}{k}$; $f_h \leftarrow b_h^{m-1} - \frac{\psi^{m-1}}{k}$
**if** $f_l < \theta_a^m < f_h$ **then**
  $\alpha \leftarrow \epsilon\alpha$
**else if** $\theta_a^m > b_l^{m-1}$ **then**
  $\delta \leftarrow \theta_a^m - f_h$; $\alpha \leftarrow \alpha(1 + (\frac{k\delta}{\psi^{m-1}})^2)$
**end**
**else if** $\theta_a^m < b_l^{m-1}$ **then**
  $\delta \leftarrow f_l - \theta_a^m$; $\alpha \leftarrow \alpha(1 + (\frac{k\delta}{\psi^{m-1}})^2)$
**end**
**else**
  **if** $\theta_a^m > f_h$ **then**
    $\delta \leftarrow \theta_a^m - f_h$; $\alpha \leftarrow \alpha(1 + \frac{k\delta}{\psi^{m-1}})$
  **else**
    $\delta \leftarrow f_l - \theta_a^m$; $\alpha \leftarrow \alpha(1 + \frac{k\delta}{\psi^{m-1}})$
  **end**
**end**

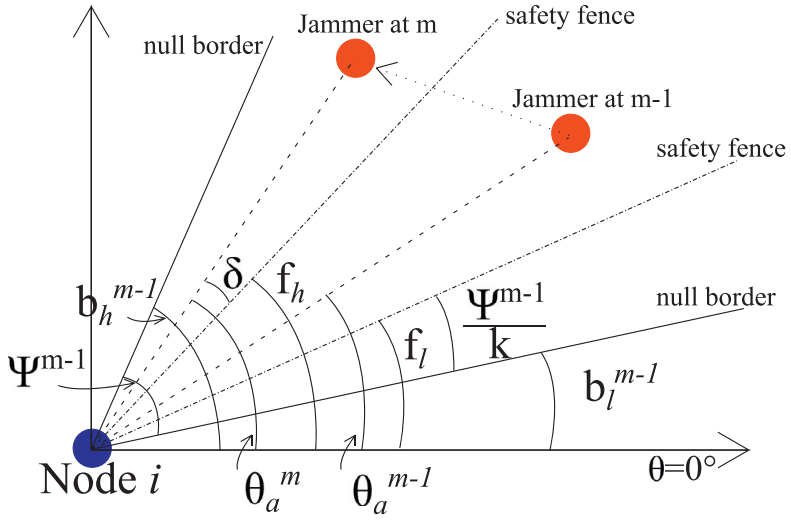



**Fig. 8.** Schema for adaptive $\alpha$ heuristics.

sequently decreases $\alpha$, resulting in a narrower null for the next interval. If the jammer stays inside the safety zone, $\alpha$ is reduced by a factor of $\epsilon \in (0, 1)$. $\delta$ is defined as the deviation of the jammer from the safety fence. At the $m$th sensing phase, if the jammer is observed between the null border and the safety fence, $\alpha$ is increased by a factor of $(1 + \frac{k\delta}{\psi^{m-1}})$. This entails $\alpha$ is doubled if the jammer is at the null border. If the jammer crosses the a border, $\alpha$ is aggressively increased by a multiplying factor of $(1 + (\frac{k\delta}{\psi^{m-1}})^2)$.

### 3.7. Calculation of null borders in 3D

From a practical point of view, The 2D framework can be applied to ground and sensor networks under attack by a ground-based jammer. To extend the compatibility of this framework to beam nulling in flying ad hoc networks and 3D mesh scenarios, the framework is generalized by considering 3D distributions of nodes and jammer. Therefore, the method of calculating null borders in the 2D framework is extended as follows.

At every sensing phase $m$, each node $i$ observes the DoA of the jammer or the angle of attack $(\theta_a^m, \phi_a^m)$ with its frame of reference. Let us consider that at the $m$th sensing phase, node $i$ measures the DoA of jammer as $(\theta_a^m, \phi_a^m)$. Node $i$ has to create a beam null that incorporates the movement of jammer in both $\theta$ and $\phi$ direction during the time interval between sensing phases at $m$ and $m+1$. In 3D space, a beam null is defined by four null borders: two borders in each of $\theta$ and $\phi$ directions. Let us define $\theta_l^m$ and $\theta_h^m$ as lower and higher null borders respectively in $\theta$ direction and $\phi_l^m$ and $\phi_h^m$ as lower and higher borders respectively in $\phi$ direction. Similar to the 2D approach, angular velocity components in $\theta$ and $\phi$ directions are used to predict the movement of the jammer. At each step $m$, $i$ calculates the angular velocity of the jammer in $\theta$ and $\phi$ directions as $v_a^m$ and $u_a^m$ respectively.

$$v_a^m = \frac{(\theta_a^m - \theta_a^{m-1})}{\tau} \tag{7}$$

$$u_a^m = \frac{(\phi_a^m - \phi_a^{m-1})}{\tau} \tag{8}$$

Consider $\overline{v_a}$ and $\sigma(v_a)$ as the mean and standard deviation of $v_a^m$, $(m \in \{1, 2, \ldots\})$ respectively. Similarly, $\overline{u_a}$ and $\sigma(u_a)$ are the mean and standard deviation of $u_a^m$. Node $i$ constructs a beam null using an algorithm that considers the history of jammer's movement. Thus, $(\theta_a^m + \tau\overline{v_a}, \phi_a^m + \tau\overline{u_a})$ gives the estimated DoA of the jammer at the $(m+1)$th phase. Since the actual velocity and direction of the jammer are unknown, the beam null should be wider in case of sudden changes in jammer's direction or velocity change. Change of velocity can be estimated with $\sigma(v_a)$ in $\theta$ direction and $\sigma(u_a)$ in $\phi$ direction. An estimation for the beam null borders in 3D can be calculated as:

$$\theta_h^m = \max(\theta_a^m, \theta_a^m + \tau(\overline{v_a} + \alpha\sigma(v_a))) \tag{9}$$

$$\theta_l^m = \min(\theta_a^m, \theta_a^m + \tau(\overline{v_a} - \alpha\sigma(v_a))) \tag{10}$$

$$\psi_\theta^m = \theta_l{}^m - \theta_l{}^m \tag{11}$$

$$\phi_h^m = \max(\phi_a^m, \phi_a^m + \tau(\overline{u_a} + \alpha\sigma(u_a))) \tag{12}$$

$$\phi_l^m = \min(\phi_a^m, \phi_a^m + \tau(\overline{u_a} - \alpha\sigma(u_a))) \tag{13}$$

$$\psi_\phi^m = \phi_l^m - \phi_l^m \tag{14}$$

Where $\psi_\theta^m$ and $\psi_\phi^m$ are the null widths constructed at the $m$th sensing phase in $\theta$ and $\phi$ directions respectively. $\alpha$ is a multiplying factor controlling the influence of randomness in the mobility. As discussed earlier, having a wider null provides higher probability of keeping the jammer inside the beam null at the cost of deactivating more links with legitimate nodes. To keep the beam null optimal based on the history of jammer's DoA observations, a heuristic for dynamic adjustment of $\alpha$ is presented in the next section.

### 3.8. Heuristics for dynamic $\alpha$ in 3D

This section demonstrates the concept of dynamically adapting the value of $\alpha$ in 3D space. Unlike the 2D approach, this heuristic in 3D has to consider the DoA of jammer in both directions, as their safety zones are dependent on each other. Fig. 9 provides a 2D representation of the $\theta$, $\phi$ space. Algorithm 2 is used at every step $m$ to adjust $\alpha$ based on the observed DoA of jammer at phase $m$ compared to the null created in the phase $m-1$. At phase $m-1$ node $i$ calculates the null borders as $\theta_l^{m-1}, \theta_h^{m-1}, \phi_l^{m-1}$ and $\phi_h^{m-1}$. These borders are implemented for the interval between $m-1$ and $m$. At phase $m$, DoA of jammer is observed at $(\theta_a^m, \phi_a^m)$. For dynamically changing the value of $\alpha$, we use a safety zone. The safety zone is bordered by two safety fences in $\theta$ direction $(f_l, f_h)$ and two fences in $\phi$ direction $(g_l, g_h)$. If the current DoA of jammer is within the safety zone, $\alpha$ is decreased by multiplying by a factor $\epsilon \in (0.5, 1)$. If the current location is outside the safety zone, then the deviation of the current position is calculated as $\gamma$, which is

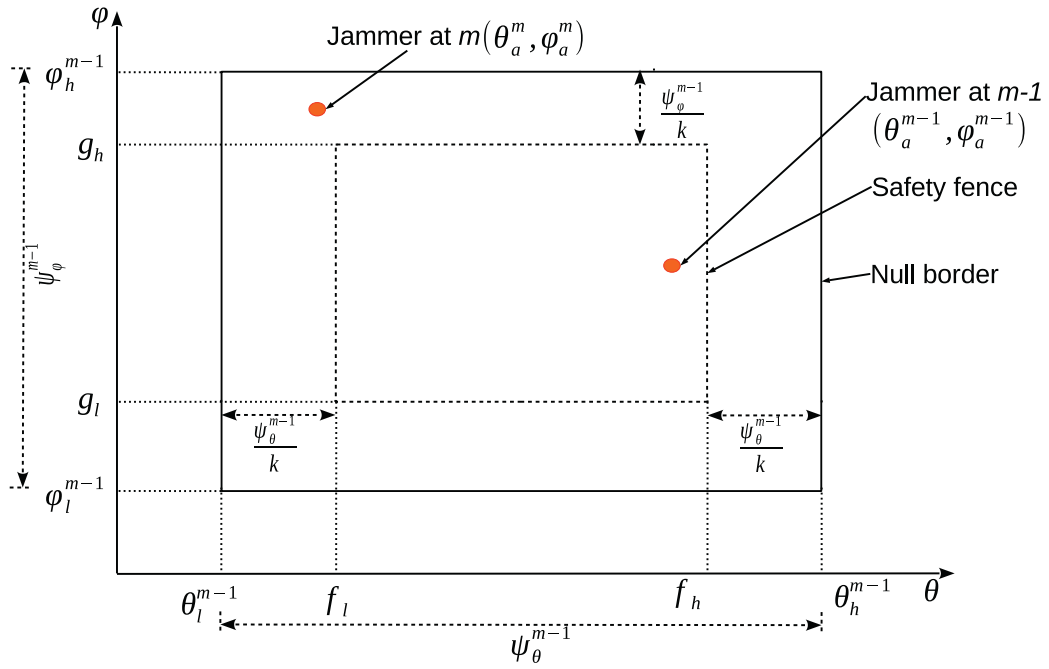



**Fig. 9.** Schema for adaptive $\alpha$ heuristics for 3D.

**Algorithm 2:** Heuristics for dynamic $\alpha$ in 3D.

$\psi_\theta^{m-1} \leftarrow \theta_h^{m-1} - \theta_l^{m-1}$
$f_l \leftarrow \theta_l^{m-1} + \frac{\psi_\theta^{m-1}}{k}$
$f_h \leftarrow \theta_h^{m-1} - \frac{\psi_\theta^{m-1}}{k}$
$\psi_\phi^{m-1} \leftarrow \phi_h^{m-1} - \phi_l^{m-1}$
$g_l \leftarrow \phi_l^{m-1} + \frac{\psi_\phi^{m-1}}{k}$
$g_h \leftarrow \phi_h^{m-1} - \frac{\psi_\phi^{m-1}}{k}$
**if** $(f_l < \theta_a^m < f_h) \wedge (g_l < \phi_a^m < g_h)$ **then**
  $\alpha \leftarrow \epsilon\alpha$
**else**
  **if** $\theta_a^m > f_h$ **then**
    $\delta_\theta \leftarrow \theta_a^m - f_h$
  **else if** $\theta_a^m < f_l$ **then**
    $\delta_\theta \leftarrow f_l - \theta_a^m$
  **end**
  **else**
    $\delta_\theta \leftarrow 0$
  **end**
  **if** $\phi_a^m > g_h$ **then**
    $\delta_\phi \leftarrow \phi_a^m - g_h$
  **else if** $\phi_a^m < g_l$ **then**
    $\delta_\phi \leftarrow g_l - \phi_a^m$
  **end**
  **else**
    $\delta_\phi \leftarrow 0$
  **end**
  $\gamma \leftarrow \max\left(\frac{k\delta_\theta}{\psi_\theta^{m-1}}, \frac{k\delta_\phi}{\psi_\phi^{m-1}}\right)$
  **if** $\gamma < 1$ **then**
    $\alpha \leftarrow \alpha(1+\gamma)$
  **else**
    $\alpha \leftarrow \alpha(1+\gamma^2)$
  **end**
**end**

the maximum value of deviation in both $\theta$ and $\phi$ directions. If $\gamma < 1$ (i.e. the current position is within the safety fence and the null border), $\alpha$ is increased by a small factor. On the other hand if the current DoA of the jammer is outside the null border ($\gamma > 1$), $\alpha$ is increased aggressively by multiplying it with a factor $(1+\gamma^2)$.

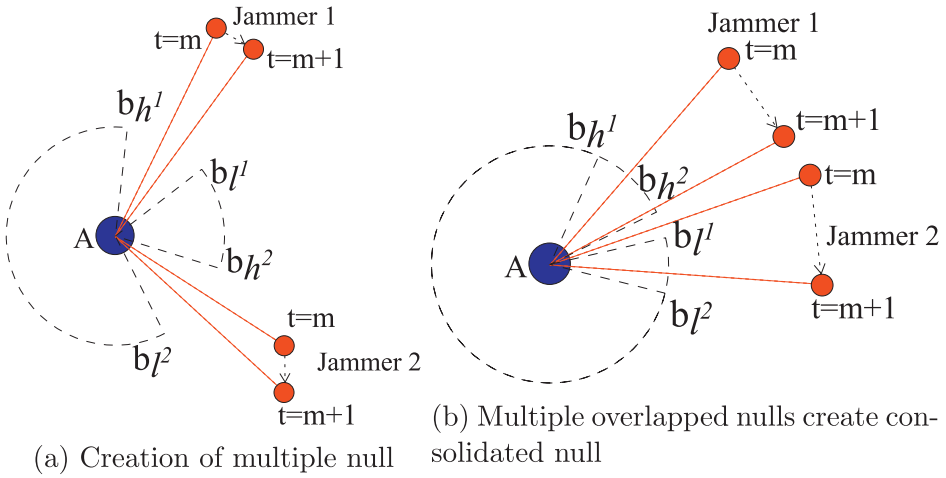
### 3.9. Defense against multiple jammers

So far we have discussed the calculation of a beam null for a single moving jammer. Each node in a network observes the position of the jammer at discrete sensing intervals ($\tau$). We assume that a node can detect a jammer precisely. In lieu of this assumption, a node can build a model to monitor the trajectory of each jammer within the jamming radius. With an antenna array, a node can adapt its gain pattern to include multiple nulls [51,52]. A node can create multiple nulls in its modified antenna gain pattern to keep the jammers in the vicinity in null region and communicate with other legitimate nodes that are not in the beam null.

For Each jammer j ($j \in 1, \ldots, J$) in the vicinity, a node monitors the DoA $(\theta_j^m, \phi_j^m)$ at each sensing period $m$. The node then use Eq. 8 to calculate the angular speed of the jammer $j$ w.r.t. the observing node. The beam null borders for the jammer $j$ is calculated using Eq. 14 at each step $m$. Fig. 10a provides an example of defense against multiple jammer. In this case, node A is within jamming radius of 2 jammers. Node a determines beam null borders $(b_{l1}, b_{h1})$ for jammer 1 and $(b_{l2}, b_{h2})$ for jammer 2. Note that, each node maintains separate value of $\alpha$ for each jammer. After observing the position of the jammer at the sensing period $m+1$, the value of $\alpha_j$ is updated using Algorithm 2. It is noteworthy that some beam nulls can overlap with each other creating a combined beam null as shown in Fig. 10b.

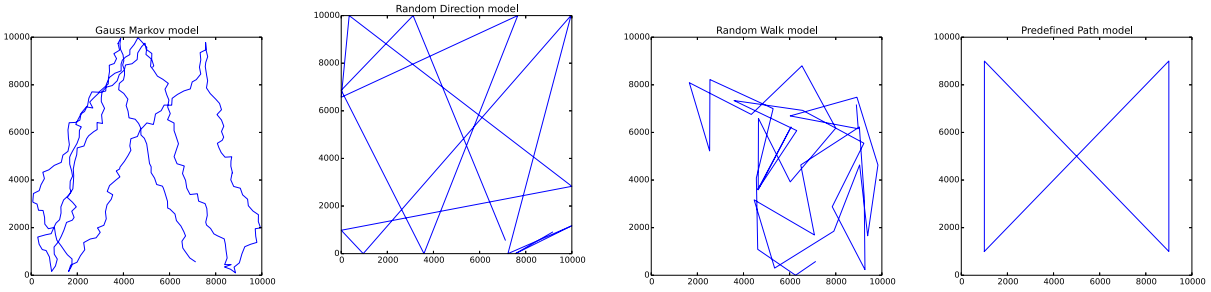
## 4. Simulation and results

To evaluate the performance of the proposed beam nulling framework, several simulations are performed. The initial simulations investigate the physical layer behavior of networks employing the proposed framework against jamming attacks. The first of these simulations considers 2D ad hoc networks where the jammer also moves in the same plane that represents the node mobil-




(a) Creation of multiple null (b) Multiple overlapped nulls create consolidated null

**Fig. 10.** Defense against multipler jammers.




(a) Gauss Markov (b) Random direction (c) Random Walk (d) Predefined Path

**Fig. 11.** Time domain sketch of different mobility models in 2D.

ity of ground vehicles. This simulation is further upgraded to emulate similar scenarios for networks and jammer in 3D space. In these simulations, survivability of networks is measured with respect to various physical layer parameters, as well as different mobility models of the jammer. The scope of measurements in then extended to include the behavior of upper layer network protocols. For this purpose, discrete event simulations in ns-3 [53] are performed to monitor the interoperability of the proposed framework with upper network layers. This section defines the parameters and configurations for each simulation, and presents the obtained results through illustrations and discussions.

### 4.1. Jammer and mobility model

In this work a moving jammer is considered. Different mobility models of the jammer impact differently on a network. A mobility model defines how a node moves or changes its direction with time. The details of the selected models (Random Walk, Random Direction, Gauss-Markov, and a predefined path) can be seen in [54] and [55]. Random-based models are vastly used in the research community but they might not reproduce a realistic movement. Gauss–Markov is a temporal dependency model that can be considered more realistic, where the velocity and direction are correlated to the previous values, avoiding abrupt changes that occur in the other models. A predefined path is also experimented assuming that a node follows a previously assigned path. Each model has its own influence in the performance of the network. Figs. 11 and 12 illustrate time domain traces of different mobility models in 2D and 3D respectively.

### 4.2. Performance metrics

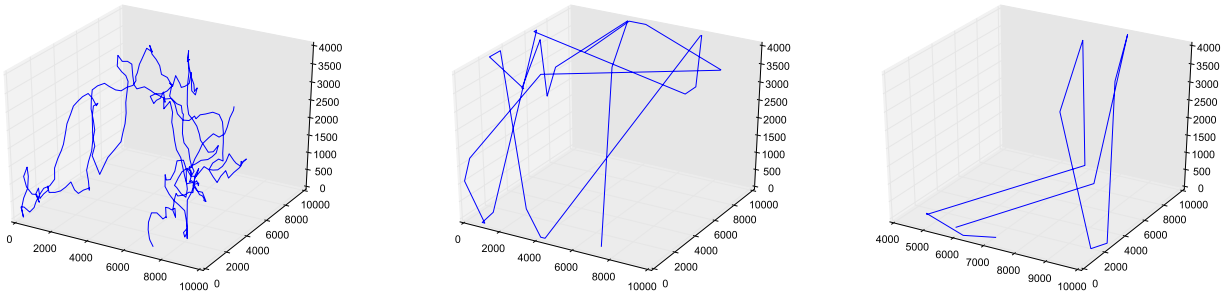
Three performance parameters are defined as follows:

- *Connectivity* is defined as the total number of connected pairs of nodes, which reflects how well connected a network is. 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 $j$ and 0 otherwise.
- The second parameter is *average number of active links*. We consider a link as the one hop communication between two neighbors. A link may fail if either of the nodes is jammed or falls in the nulled region of the other one.
- The next performance parameter considered is the *average number of islands*. Islands are the subgroups of nodes in a disconnected network where the nodes inside an island are connected. If a network is completely connected, the number of islands is 1. A higher number of islands reflects more disruption in the network.

The simulator monitors the above mentioned metrics at each iteration. It calculates the average of these metrics after the full simulation and records them as the result.

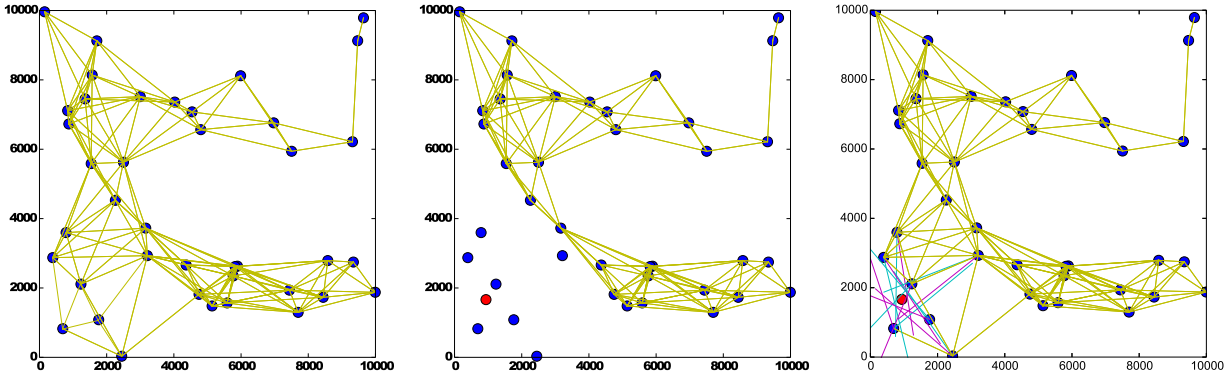
### 4.3. Simulation for 2D environment

A customized tick based simulator is developed to measure the performance of the proposed algorithm. Each tick represents the time interval ($\tau$) between two consecutive sensing periods. The default parameters used are listed in Table 1. During the sensing phase, at each tick ($m$), every node checks for the jammer's angular position ($\theta_a^m$). Each node then determines its new beamform



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

**Fig. 12.** Trace of different mobility models in 3D.



(a) no jammer, omnidirectional antenna (b) jammer, omnidirectional antenna (c) jammer, adaptive nulling antenna

**Fig. 13.** Snapshots of simulations.

**Table 1**
Default parameters for 2D simulation.

| Parameters | Symbol | Values |
|---|---|---|
| Simulation area | | $10,000 \times 10,000\,\text{m}^2$ |
| Transmission power | $P_t$ | 30 dBm |
| Received power cutoff | $P_r$ | –78 dBm |
| Communication frequency | | 2.4 GHz |
| Communication radius | | 3146 m |
| Initial $\alpha$ | $\alpha$ | 2.5 |
| DOA error standard deviation | $\sigma_{doa}$ | 0.05 |
| Beam nulling error standard deviation | $\sigma_{bn}$ | 0.05 |
| Number of nodes simulated | $N$ | 100 |
| Sensing interval | $\tau$ | 50 ms |
| Simulation time | | 500 s |
| Jammer's mobility model | | Random walk |

according to Eq. 14 and updates $\alpha$ using Algorithm 1. After the sensing and beamforming phases, communication with neighbors takes place until the time interval ($\tau$) ends, when the same cycle is repeated.

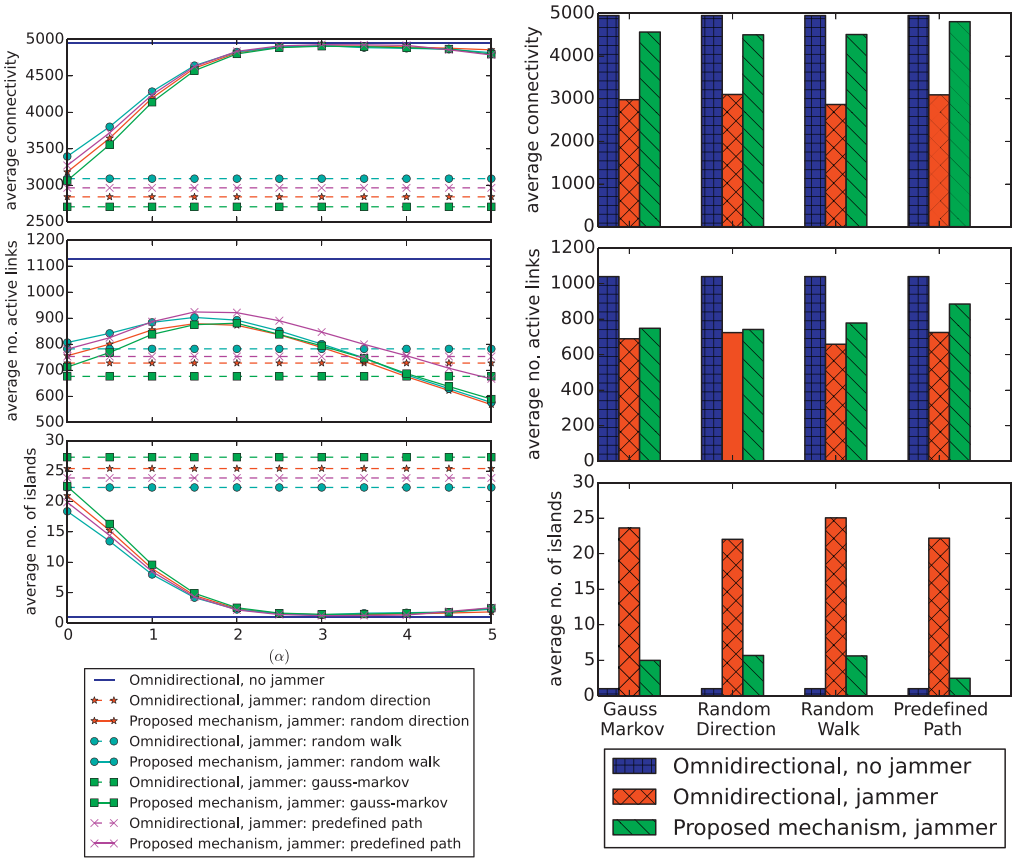
Each simulation generates the position of nodes randomly. The same set of positions is used to measure the performance of the network while varying other parameters. For simplicity, the simulator considers a free space path loss model to calculate the received power. The simulator defines the links between two nodes on each iteration based on the received power from the corresponding neighbor and interference from the jammer at that instance. If the power received is above the cutoff and neither of the nodes are jammed, the simulator considers the link to be active. The simulator considers a scenario of $N$ nodes scattered randomly in an area of $10,000 \times 10,000\,\text{m}^2$. Each node transmits with power of 30 dBm and the average communication radius is calculated as 3146m.

Fig. 13 illustrates the advantage of using the proposed framework in the presence of a jammer in 2D environment. Here, 100 nodes are scattered over the geographical area. This snapshot is taken in the middle of a simulation. One hop communication links are represented with yellow lines. Network connectivity of two benchmark scenarios are considered. Fig. 13a depicts the case of no jamming which leaves the network connected. Fig. 13b presents network connectivity in the presence of a jammer when nodes use omnidirectional antennas. In this scenario, we can clearly see that many links are deactivated as the nodes are exposed to destructive interference from the jammer. Fig. 13c demonstrates the effect of employing the proposed framework where the null borders $b_l$ and $b_h$ are represented by cyan and magenta lines respectively. The nodes in the vicinity of the jammer use adaptive beam nulling in order to avoid disruption, and are able to maintain the connectivity with neighboring nodes active. It can also be observed that the nodes which are further from the jammer do not use beam nulling.

The simulator considers the possibility of errors in DoA estimation and beam nulling. As discussed in Section 3.4, we consider the measurement error ($\mathbf{e}_{doa}$) and beam nulling error $e_{bn}$ to be zero mean Gaussian noise. Where $\mathbf{e}_{doa} \sim \mathcal{N}(0, \sigma_{doa})$, and $e_{bn} \sim \mathcal{N}(0, \sigma_{bn})$. Here $\sigma_{doa}$ and $\sigma_{bn}$ are standard deviation of error for DoA measurement and beam nulling respectively.

#### 4.3.1. Discrete fixed $\alpha$

In the initial phase of the simulation, the effect of $\alpha$ on the network's performance is investigated. In this case, the network is




(a) Simulation results with fixed $\alpha$

(b) Simulation with different trajectory models

**Fig. 14.** Results for simulation in 2D environment.

simulated without adaptive $\alpha$, i.e. nodes do not use Algorithm 1. Fig. 14a presents the simulation results when $\alpha$ is fixed. The $x$-axis of these plots represent discrete values of $\alpha$ that form the beam null in Eq. 14. Nine different scenarios are considered: one benchmark scenario with no jamming, and for each mobility model we simulated the network once with omnidirectional antenna, and once with the proposed beam nulling algorithm. The worst case scenario occurs when there is a jammer in the vicinity and the nodes use omnidirectional antenna, consequently the performance is heavily affected by the presence of the jammer. The top benchmark result is obtained similarly to the worst case but with no jammer present, therefore the communications are not affected by any adversary. It can be seen from the results that when there is no jammer, the network is completely connected as the number of islands is 1. For a completely connected network with $n$ nodes, the connectivity value is $\frac{n(n-1)}{2}$. Therefore, in a network of 100 nodes with no jammer, the connectivity is 4950, confirming the simulated result.

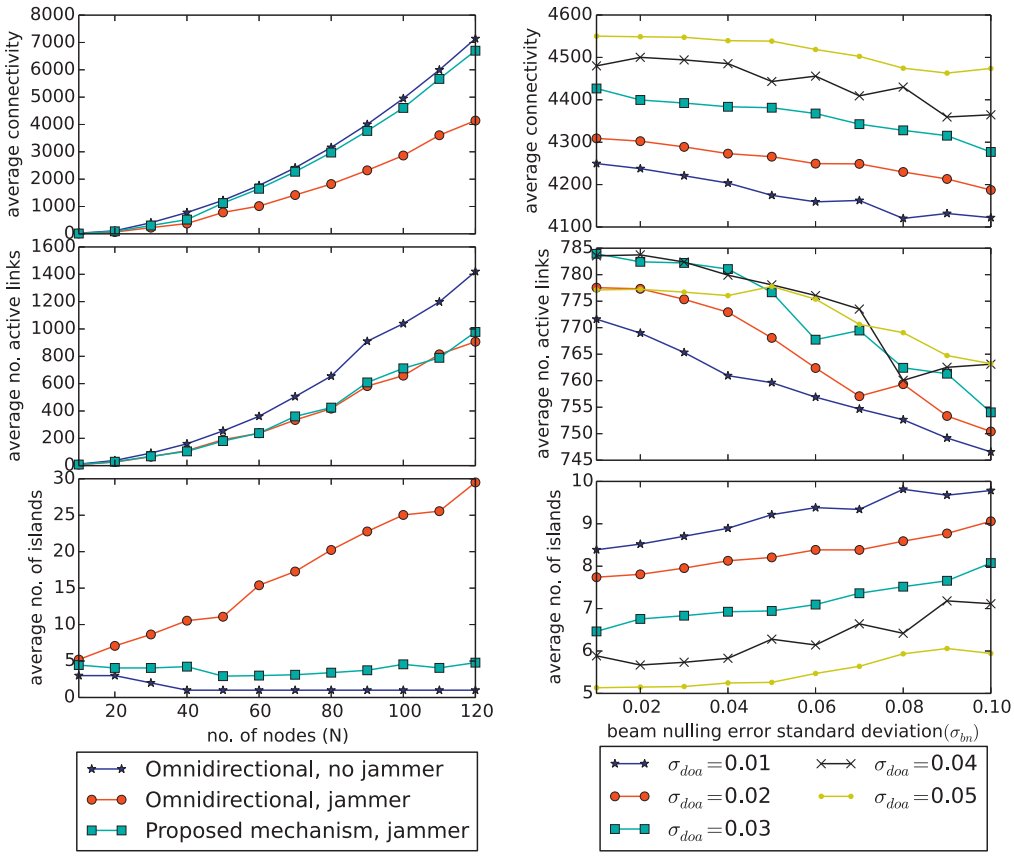
When nodes do not use beam nulling, islands are created, resulting in a poor connectivity value. Also it is observed that in the presence of a jammer, adaptive beam nulling significantly improves the overall performance in terms of all the metrics considered. In addition, when a jammer is present and the nodes do not apply beam nulling, the network is heavily affected, and a larger number of islands is created. However, when nodes apply adaptive beam nulling, different trajectory models perform differently with respect to the values of $\alpha$.

It is noteworthy to mention that for higher values of $\alpha$, the number of average links may fall below the benchmark case of omnidirectional nodes in the presence of a jammer. This is because a higher value of $\alpha$ creates a wider null that results in deactivation of more links. A node may reduce this shortcoming by sensing the jammer more frequently but this also reduces the data communication window. In addition, it can be observed that as $\alpha$ increases, the average number of islands decreases, while the number of active links begin to deteriorate after a peak. This phenomenon can be interpreted as a rise in congestion.

Another conclusion that can be derived from these results is that a fixed value of $\alpha$ does not guarantee the optimal performance, since the mobility pattern of the jammer is not known to the nodes. A node estimates the jammer's mobility through periodic sensing. Therefore, the value of $\alpha$ must be dynamically updated based on the history of the jammer's movements.

#### 4.3.2. Effect of jammer's mobility model

Four different mobility models for the jammer are considered. Fig. 14b illustrates the impact of these models on the defending network. It can be seen that the Random Direction and Random Walk models adversely affect the performances of the network, since the direction of the jammer undergoes abrupt changes in random intervals. For the predefined path and Gauss-Markov models, the direction and velocity are constant for the majority of the time, which allows the proposed framework to accurately estimate the jammer's movement. It is observed that for 100 nodes, the proposed mechanism achieves an improvement in connectivity of up to 57.27% relative to the omnidirectional case under jamming.




(a) varying number of Nodes (b) varying error rate

**Fig. 15.** Simulation results varying different simulation parameters in 2D environment.

*4.3.3. Effect of node density*

Fig. 15a illustrates the effect of varying number of nodes in the network which constitutes a change in node density. It is observed that when a network is not connected, the number of islands increases. As the number of nodes increases, connectivity is well preserved in the no jamming scenario. The jammer succeeds in disabling more links when the node density is higher. Even though the number of link failures is on a similar level as the worst benchmark of omnidirectional with jamming, connectivity and number of islands demonstrate a better performance. In the benchmark scenario with omnidirectional antennas, the number of islands increases greatly with an increase in the number of nodes, since the density is higher and the attacker has more links in its jamming range. The proposed adaptive beam nulling approach succeeds in keeping the connectivity and number of islands close to the benchmark scenario of no jamming.

*4.3.4. Effect of errors in beam nulling*

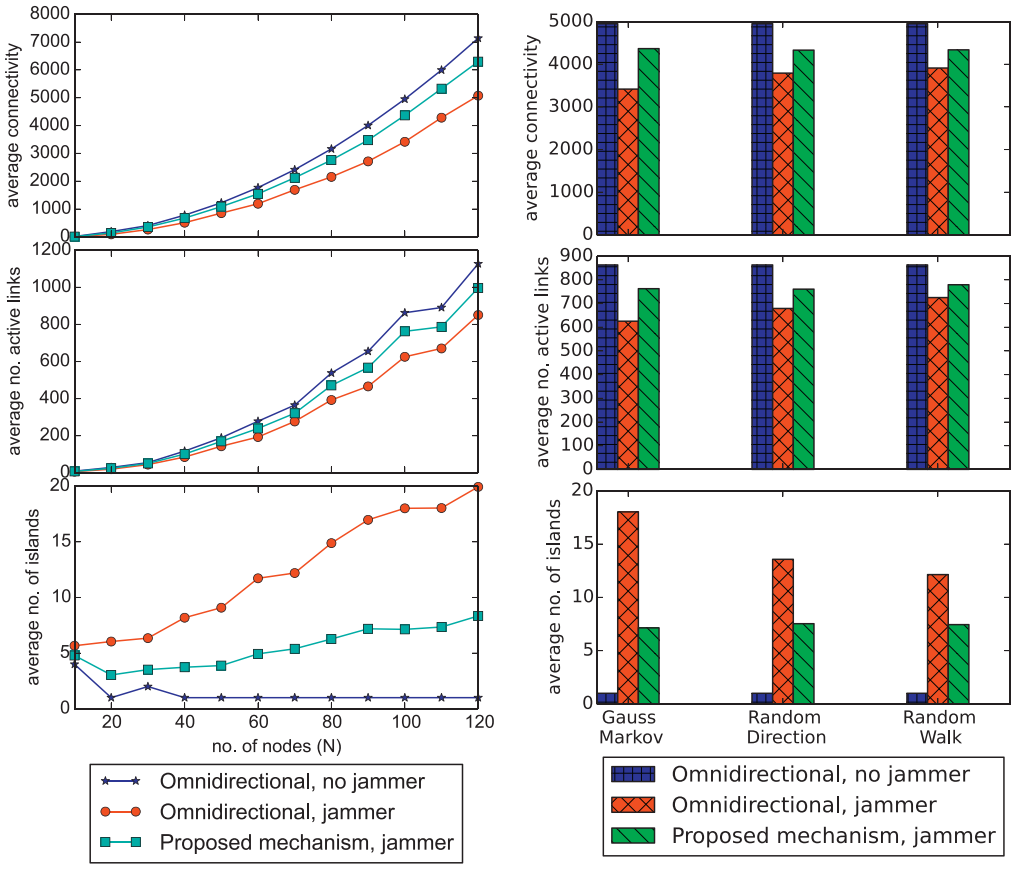
As discussed earlier, errors are introduced in the simulator to account for the practical inaccuracies in beam nulling and DoA estimation. The effective beam null border is a random function with the mean of intended border angle and standard deviation of $\sigma_{bn}$. Similarly for each node the observed DoA is a random function of mean at the actual DoA and standard deviation of $\sigma_{doa}$. Fig. 15b plots the performance of the network w.r.t. the error in beam nulling. The $x$-axis is $\sigma_{bn}$, while the simulations are repeated with several different values of $\sigma_{doa}$. With a $\sigma_{doa}$ of 0.1 that entails an error of 5.7$^o$ in DoA measurement, the connectivity still remains close to that of the no jamming scenario. The plots reflect that both the errors decrease the network performance significantly as the jammer is not tracked accurately. However, with a higher value of error in measurement, the proposed framework still performs better than the omnidirectional antenna case.

*4.4. Simulation for 3D environment*

The simulation is extended to evaluate the performance of our proposed framework in 3D space. The simulation area is a $10 \times 10 \times 4\,\text{km}^3$ volume where each node has a communication range of 3 km. Other system related parameters are the same as the 2D simulation discussed earlier. At each tick $m$, all nodes observe the DoA of the jammer $\theta_a^m, \phi_a^m$. With the history of the DoA of jammer, a node creates the null as described in Section 3.7. In the next phase, the modified beam is used for data transmission with neighbors. We observe the same system parameters as discussed earlier.

*4.4.1. Effect of node density*

Fig. 16a demonstrates the effect of jamming on networks with different node densities. Since the geographical area is fixed, changing number of nodes in the network effectively changes the node density. In this simulation, the jammer moves according to the Gauss–Markov mobility model. With lower node density, the network can be partitioned into multiple islands as seen in the benchmark scenario of no jammer. With increase in node density, the average number of islands is reduced. In the presence of jammer, the network is broken into multiple islands even though the node density is higher. With the proposed framework, networks are able to retain the average number of islands close to the benchmark scenario of no jamming. It is also observed that




(a) Changing node density (b) Changing jammer's mobility model

**Fig. 16.** Simulation results for 3D network.

the proposed framework results in a higher number of active links for the network.

*4.4.2. Effect of mobility model of jammer*

Fig. 16b illustrates the performance of the network under jamming with different mobility models. In this simulation, 100 nodes are scattered randomly over the volume. We can clearly see that for the case of omnidirectional antenna with jammer, average connectivity as well as average number of active links are lower for Gauss–Markov model compared to other mobility models. Again, the Gauss–Markov model creates more islands than the other mobility models. Thus we can say, in 3D space, a jammer with Gauss–Markov mobility model impacts the network most adversely. The proposed framework successfully avoids jamming by creating beam nulls. It can also be observed that although in the omnidirectional case, the jammer with Gauss–Markov mobility affects the network performance badly, the proposed framework provides almost the same performance for all the mobility models. So, we can conclude that the framework with dynamic $\alpha$ heuristic is effective in calculating a suitable null width regardless of jammer's mobility model.

*4.5. Simulation with upper layer protocols*

To ascertain the effects of the proposed mechanism on the upper network layer protocols, physical layer simulations are extended with network simulations in ns-3 [53], which is a discrete event simulator that provides reliable results when using complex networks with multiple protocol stacks. The simulations are focused towards the interoperability of the proposed framework with two ad hoc routing protocols, namely AODV and DSDV. For both routing protocols the IEEE 802.11b MAC is used. Ad hoc On-demand Distance Vector (AODV) is a reactive routing protocol with some active elements. In this scheme, the routes are discovered only when needed, but they are maintained for as long as possible. This can cause delay when there is data ready to be transferred by a node, but no route is stored in its routing table [56]. Destination-Sequenced Distance Vector (DSDV) is a proactive protocol, meaning it will regularly update the routing table, even when there is no data to be transmitted. DSDV requires new sequence numbers before the topology can converge again, making it's implementation in highly mobile networks undesirable [57].

To simulate the proposed mechanism, a proof of concept antenna model [58] is used, which contains a few parameters: beamwidth, gain inside the beamwidth, gain outside the beamwidth, and orientation. In our case, the beamwidth corresponds to the nulled region, and the gain inside it is set to –60 dB, the gain outside is 0 db, and the orientation is defined as the direction towards the jammer. The traffic in the application layer is generated by 10 random source nodes and received by 10 random destination nodes. Details of the simulation parameters are presented in Table 2. Three different applications are used to send data at different rates. The amount of data to be transferred is randomly chosen by each source according to the probabilities shown in Table 3. Tables 4 and 5 provide the default parameters used for AODV and DSDV protocols.

The simulator provides four performance metrics: *packet delivery ratio (PDR), mean hop count, mean delay* and *bytes received*. PDR

**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 | $10,000 \times 10,000\,m^2$ |
| 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 |

**Table 3**
Application layer parameters.

| Application | Bytes generated | Probability |
|---|---|---|
| Text | 10,000 | 0.6 |
| Image | 500,000 | 0.3 |
| Video | 5,000,000 | 0.1 |

**Table 4**
Parameters for Simulating AODV.

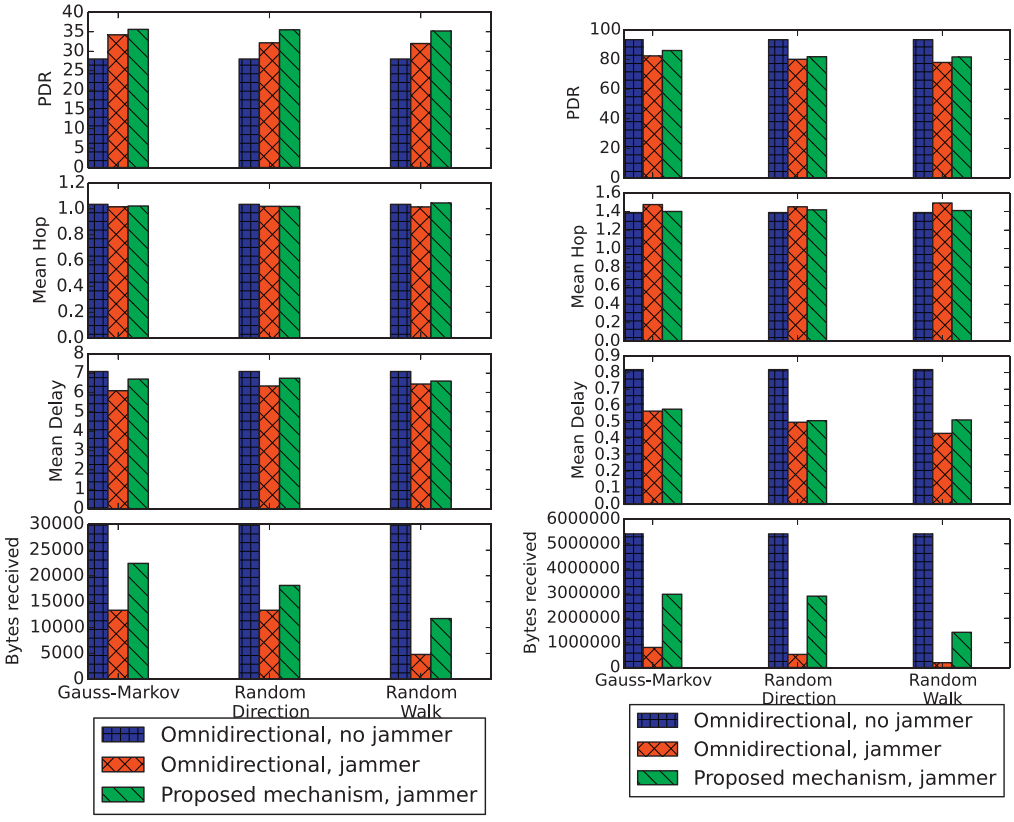
| Parameters | Values |
|---|---|
| Hello interval | 1 s |
| RREQ retries | 2 |
| RREQ rate limit | 10 per second |
| RERR rate limit | 10 per second |
| Node traversal time | 40 ms |
| Next hop wait | 50 ms |
| Active route timeout | 3 s |
| Net diameter | 35 |
| Max queue length | 64 packets |
| Max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

**Table 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 |

is the ratio between the number of packets received by the destination to the number of packets sent by the source. Mean hop count is the average number of hops taken by the packets in the simulation (including control packets) to reach their destinations. Mean delay is the average time taken for the packets (including control packets) to reach their destinations. Bytes received is the amount of bytes received by the destination nodes in the application layer. Fig. 17a and b illustrate the performance of a network under jamming using AODV and DSDV respectively. Three different mobility models are explored for the jammer: Gauss–Markov, Random Direction, and Random Walk.

The upper subplot in both Fig. 17a and b represent PDR value w.r.t different mobility models in the simulation. For both of the routing protocols, the proposed framework ensures enhanced PDR.




(a) AODV (b) DSDV

**Fig. 17.** Simulation results from ns-3 simulation.

It can be concluded that regardless of the routing protocol or the mobility pattern, the proposed framework is able to provide enhancement in the performance compared to the case of omnidirectional network under jamming.

The mean hop count plot shows a negligible difference for AODV, but in DSDV it is evident that the proposed framework improves this metric by keeping links active even when their corresponding nodes are inside the jammed region. In the presence of the jammer the hop count is higher than in the other cases.

For DSDV as in Fig. 17b, it can be seen that the mean delay in our proposed framework is lower compared to the benchmark scenario of omnidirectional without jammer. As beam nulling is used, nodes experience less interference from the neighbors, reducing the waiting time for packets in the queue, since there is less collisions due to medium access conflicts. With the reduced waiting time, the packets are transferred to the destination faster. This means that the proposed framework not only retains links in the jammed region but also reduces congestion on the links outside of the jammed region.

The last subplot illustrates the amount of data received by all destination nodes. It is observed that DSDV outperforms AODV by a large margin for all mobility models. This is partly due to the resolution of physical layer simulations, which cause the loss of some AODV messages during the network simulation. DSDV on the other hand is a proactive protocol, keeping the routes updated as link failures are detected. This characteristic plays an advantage and gives DSDV the better performance in the simulations.

### 4.6. 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 3.9. The simulated network consists of 100 nodes, the rest of the parameters are kept same as before as listed in Table 1. In Fig. 18 we plot the simulation results. Simulations are performed for different number of jammers in between 0 and 5, where 0 represents the case of no jammer as a bench mark of best case scenario.

Results show that using the adaptive beam nulling improves the network connectivity when compared with the same case with omnidirectional antenna. In comparison with the benchmark scenario of no jammer, the adaptive beam nulling approach shows a decrease of 39.94%, while using omnidirectional antenna decreased to 97.73%.

The number of active links and the number of islands also show improvements. The average number of active links decreased 72.05% with the proposed mechanism and 91.3% when no protection was used. Even with the apparent large decrease, the number of islands with the proposed beam nulling is 5.92 times lower than the omnidirectional case, demonstrating the feasibility of the proposed method for defense against multiple jammers.

## 5. Conclusion and future works

This paper proposes a framework for adaptive beam nulling in multihop ad hoc networks as a mitigation technique against a moving jammer. Performance of the proposed framework is studied through physical layer and full-stack network simulations of various network topologies and mobility models of the jammer in both 2-dimensional and 3-dimensional environments. Obtained results indicate that employing this framework leads to significant improvements in survivability of the links and connectivity over the performance of networks with omnidirectional interfaces. Also, to increase the accuracy of the simulated models for practical implementations, effects of varying inherent errors on the performance of a beam nulling ad hoc network is studied.

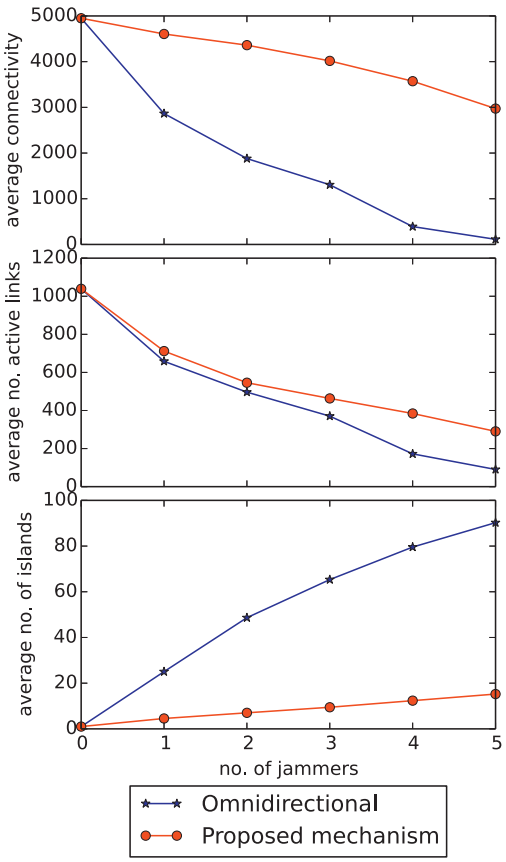



**Fig. 18.** Simulation results for multiple jammer.

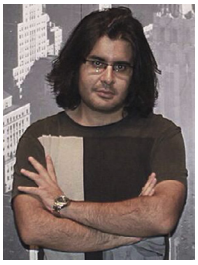
With the promising performance of the proposed framework, investigating potential avenues for its further development may lead to an even better enhancement in mitigation capabilities of this approach. Our proposed framework proposes a seamless integration of beam nulling with the network protocol stack, but customization and optimization of upper network layers such as MAC and routing may result in an improved network performance. Also, the method used for prediction of the jammer's trajectory can be built upon by applying stochastic methods and learning algorithm to generate more accurate predictions.

## References

[1] S. Bhunia, V. Behzadan, P.A. Regis, S. Sengupta, Performance of adaptive beam nulling in multihop ad-hoc networks under jamming, in: High Performance Computing and Communications (HPCC), 2015 IEEE 7th International Symposium on Cyberspace Safety and Security (CSS), 2015 IEEE 12th International Conferen on Embedded Software and Systems (ICESS), 2015 IEEE 17th International Conference on, IEEE, 2015, pp. 1236–1241.
[2] L. De Filippis, G. Guglieri, F. Quagliotti, Path planning strategies for UAVs in 3D environments, J. Intel. Robot. Syst. 65 (1–4) (2012) 247–264.
[3] A. Farago, Graph theoretic analysis of ad-hoc network vulnerability, WiOpt'03: Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks, 2003.
[4] A. Yahya, O. Sidek, J. Mohamad-Saleh, C.M.D.E. Center, Performance analyses of fast frequency hopping spread spectrum and jamming systems., Int. Arab J. Inf. Technol. 5 (2) (2008) 115–119.
[5] L.B. Milstein, Interference rejection techniques in spread spectrum communications, Proc. IEEE 76 (6) (1988) 657–671.
[6] B.D. Van Veen, K.M. Buckley, Beamforming: A versatile approach to spatial filtering, IEEE ASSP Mag. 5 (2) (1988) 4–24.
[7] K.-B. Yu, Adaptive beamforming for satellite communication with selective earth coverage and jammer nulling capability, Sig.l Process. IEEE Trans. 44 (12) (1996) 3162–3166.
[8] W.C. Cummings, An adaptive nulling antenna for military, Lincoln Lab. J.,1992, 5 (2) (1992).
[9] G. Okamoto, Jammer nulling via low complexity blind beamforming algorithm, in: Antennas and Propagation Society International Symposium, 2007 IEEE, IEEE, 2007, pp. 25–28.

[10] W. Zhu, B. Daneshrad, J. Bhatia, H.-S. Kim, D. Liu, K. Mohammed, R. Prabhu, S. Sasi, A. Shah, MIMO systems for military communications, in: Military Communications Conference, 2006. MILCOM 2006. IEEE, IEEE, 2006, pp. 1–7.
[11] D. Mingjie, P. Xinjian, Y. Fang, L. Jianghong, Research on the technology of adaptive nulling antenna used in anti-jam GPS, in: Radar, 2001 CIE International Conference on, Proceedings, IEEE, 2001, pp. 1178–1181.
[12] G.K. Rao, R.S.H. Rao, Status study on sustainability of satellite communication systems under hostile jamming environment, in: India Conference (INDICON), 2011 Annual IEEE, IEEE, 2011, pp. 1–7.
[13] J.C. Liberti, T.S. Rappaport, Smart antennas for wireless communications: IS-95 and third generation CDMA applications, Prentice Hall PTR, 1999.
[14] W. Xu, K. Ma, W. Trappe, Y. Zhang, Jamming sensor networks: attack and defense strategies, Netw. IEEE 20 (3) (2006) 41–47.
[15] D.S. Berger, F. Gringoli, N. Facchi, I. Martinovic, J. Schmitt, Gaining insight on friendly jamming in a real-world IEEE 802.11 network, in: Proceedings of the 2014 ACM conference on Security and Privacy in Wireless & Mobile Networks, ACM, 2014, pp. 105–116.
[16] M. Strasser, B. Danev, S. Čapkun, Detection of reactive jamming in sensor networks, ACM Trans. Sensor Netw. (TOSN) 7 (2) (2010) 16.
[17] K. Pelechrinis, M. Iliofotou, S.V. Krishnamurthy, Denial of service attacks in wireless networks: The case of jammers, Commun. Surv. Tut. IEEE 13 (2) (2011) 245–257.
[18] D. Giustiniano, V. Lenders, J.B. Schmitt, M. Spuhler, M. Wilhelm, Detection of reactive jamming in DSSS-based wireless networks, in: Proceedings of the Sixth ACM Conference on Security and Privacy in Wireless and Mobile Networks, ACM, 2013, pp. 43–48.
[19] W. Xu, W. Trappe, Y. Zhang, T. Wood, The feasibility of launching and detecting jamming attacks in wireless networks, in: Proceedings of the 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing, ACM, 2005, pp. 46–57.
[20] M. Wilhelm, V. Lenders, J.B. Schmitt, On the reception of concurrent transmissions in wireless sensor networks, Wireless Commun. IEEE Trans. 13 (12) (2014) 6756–6767.
[21] J.T. Chiang, Y.-C. Hu, Cross-layer jamming detection and mitigation in wireless broadcast networks, IEEE/ACM Trans. Netw. (TON) 19 (1) (2011) 286–298.
[22] C. Sorrells, L. Qian, H. Li, Quickest detection of denial-of-service attacks in cognitive wireless networks, in: Homeland Security (HST), 2012 IEEE Conference on Technologies for, IEEE, 2012, pp. 580–584.
[23] M. Spuhler, D. Giustiniano, V. Lenders, M. Wilhelm, J.B. Schmitt, Detection of reactive jamming in DSSS-based wireless communications, Wireless Commun. IEEE Trans. 13 (3) (2014) 1593–1603.
[24] C. Pöpper, M. Strasser, S. Čapkun, Anti-jamming broadcast communication using uncoordinated spread spectrum techniques, Selected Areas Commun. IEEE J. 28 (5) (2010) 703–715.
[25] C. Sorrells, P. Potier, L. Qian, X. Li, Anomalous spectrum usage attack detection in cognitive radio wireless networks, in: IEEE International Conference on Technologies for Homeland Security (HST), 2011, IEEE, 2011, pp. 384–389.
[26] S. Bhunia, S. Sengupta, F. Vazquez-Abad, CR-Honeynet: A learning & decoy based sustenance mechanism against jamming attack in CRN, in: Military Communications Conference (MILCOM), 2014 IEEE, IEEE, 2014, pp. 1173–1180.
[27] S. Bhunia, S. Sengupta, F. Vzquez-Abad, Performance analysis of CR-Honeynet to prevent jamming attack through stochastic modeling, Perv. Mob. Comput. 21 (2015) 133–149.
[28] C.A. Balanis, Antenna Theory: Analysis and Design, John Wiley & Sons, 2012.
[29] H.L. Van Trees, Detection, Estimation, and Modulation Theory, John Wiley & Sons, 2004.
[30] C. Vaidyanathan, K.M. Buckley, Performance analysis of the MVDR spatial spectrum estimator, Sig. Process. IEEE Trans. 43 (6) (1995) 1427–1437.
[31] V. Krishnaveni, T. Kesavamurthy, et al., Beamforming for direction-of-arrival (DOA) estimation: A survey, Int. J. Comput. Appl. 61 (11) (2013) 4–11.
[32] J. Volakis, Antenna Engineering Handbook, fourth edition, McGraw-Hill Companies,Incorporated, 2007.
[33] J.M. Becker, Dynamic beamforming optimization for anti-jamming and hardware fault recovery, CARNEGIE MELLON UNIVERSITY, 2014 Ph.D. thesis.
[34] L. Lei, X. Rongqing, L. Gaopeng, Robust adaptive beamforming based on generalized sidelobe cancellation, in: Radar, 2006. CIE'06. International Conference on, IEEE, 2006, pp. 1–4.
[35] Y. Xu, Z. Liu, Noncircularity-rate maximization: a new approach to adaptive blind beamforming, in: Wireless Communications, Networking and Mobile Computing, 2009. WiCom'09. 5th International Conference on, IEEE, 2009, pp. 1–4.
[36] S. Chen, L. Hanzo, N.N. Ahmad, A. Wolfgang, Adaptive minimum bit error rate beamforming assisted receiver for QPSK wireless communication, Dig. Sig. Process. 15 (6) (2005) 545–567.
[37] R. Haupt, H. Southall, Experimental adaptive nulling with a genetic algorithm, Microwave J. Euroglobal Ed. 42 (1999) 78–89.
[38] A. Massa, M. Donelli, F.G. De Natale, S. Caorsi, A. Lommi, Planar antenna array control with genetic algorithms and adaptive array theory, Antennas Prop. IEEE Trans. 52 (11) (2004) 2919–2924.
[39] Y.-j. Lee, J.-W. Seo, J.-K. Ha, D.-c. Park, Null steering of linear phased array antenna using genetic algorithm, in: Microwave Conference, 2009. APMC 2009. Asia Pacific, IEEE, 2009, pp. 2726–2729.
[40] D.J. Sadler, Planar array design for low ambiguity, in: Antennas & Propagation Conference, 2009. LAPC 2009. Loughborough, IEEE, 2009, pp. 713–716.
[41] H. Evans, P. Gale, B. Aljibouri, E. Lim, E. Korolkwiwicz, A. Sambell, Application of simulated annealing to design of serial feed sequentially rotated $2 \times 2$ antenna array, Electron. Lett. 36 (24) (2000) 1987–1988.
[42] S. Ram, A study of adaptive beamforming techniques using smart antenna for mobile communication, National Institute of Technology Rourkela, 2007 Ph.D. thesis.
[43] Z. Szalay, 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, 2015, pp. 1–4.
[44] 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, 2012, pp. 212–216.
[45] F. Akbari, S. Moghaddam, V. Vakili, MUSIC and MVDR DOA estimation algorithms with higher resolution and accuracy, in: Telecommunications (IST), 2010 5th International Symposium on, 2010, pp. 76–81.
[46] O. Bazan, M. Jaseemuddin, A survey on mac protocols for wireless adhoc networks with beamforming antennas, Commun. Surv. Tut. IEEE 14 (2) (2012) 216–239.
[47] A. AhmadAnsari, Z. Hasan, K. Mohammad Athar Siddique, M. Jahangeer Alam, Performance comparison for omnidirectional and directional MAC protocols for ad hoc network, Int. J. Comput. Appl. 70 (21) (2013) 26–31.
[48] J. Niu, R. Zhang, L. Cai, J. Yuan, A fully-distributed directional-to-directional MAC protocol for mobile ad hoc networks, in: Computing, Networking and Communications (ICNC), 2015 International Conference on, IEEE, 2015, pp. 766–770.
[49] M.S. Ullah, F.N. Nur, N.N. Moon, Optimization of wireless ad-hoc networks using an adjacent collaborative directional MAC (ACDM) protocol, Int. J. Comput. Appl. 114 (3) (2015) 39–45.
[50] A. Nasipuri, S. Ye, J. You, R.E. Hiromoto, A MAC protocol for mobile ad hoc networks using directional antennas, in: Wireless Communications and Networking Confernce, 2000. WCNC. 2000 IEEE, volume 3, IEEE, 2000, pp. 1214–1219.
[51] W.-D. Wirth, Radar techniques using array antennas (FEE radar, sonar, navigation & avionics series), volume 10, IET, 2001.
[52] S. Chandran, Adaptive Antenna Arrays: Trends and Applications, Springer Science & Business Media, 2013.
[53] ns-3 Consortium, Network simulator, ns-3, (https://www.nsnam.org/).
[54] F. Bai, A. Helmy, A survey of mobility models, Wireless Ad hoc Netw. Univ. South. California, USA 206 (2004).
[55] O. Bouachir, A. Abrassart, F. Garcia, N. Larrieu, A mobility model for UAV ad hoc network, in: ICUAS 2014, International Conference on Unmanned Aircraft Systems, 2014, pp. pp–383.
[56] C. Perkins, E. Belding-Royer, S. Das, Ad Hoc on-demand distance vector (AODV) routing, Technical Report, United States, 2003.
[57] C.E. Perkins, P. Bhagwat, Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers, in: Proceedings of the Conference on Communications Architectures, Protocols and Applications, in: SIGCOMM '94, ACM, New York, NY, USA, 1994, pp. 234–244.
[58] Directional antenna patch for ns-3, (https://codereview.appspot.com/6620057/#ps1).

**Suman Bhunia** is a Ph.D. candidate in the department of Computer Science at the University of Nevada, Reno. His research area is security and vulnerabilities in wireless communications, with focus on sustenance of wireless communication in the event of jamming attacks. He has also worked on other research areas such as neighbor discovery in directional antenna systems and free space optical communication, maintaining quality of voice traffic over wireless, etc. Suman aims to defend his Ph.D. thesis in Fall 2016.



**Vahid Behzadan** is a Ph.D. student in the department of Computer Science and Engineering at the University of Nevada, Reno. His area of research is security of wireless communications with specific focus on vulnerabilities and security of airborne and mobile ad hoc wireless networks. Vahid's background includes research and commercial experience in design and implementation of microwave radio interfaces and reconfigurable antennas for airborne and satellite applications.

**Paulo Alexandre Regis** is a Ph.D. Student at the University of Nevada, Reno. Paulo obtained his BS in Telecommunication Engineering in 2010 at the Regional University of Blumenau, Brazil. His research interests are optimization, security and application of mobile ad hoc networks. His background experience includes mobile development and network systems administration.

**Dr. Shamik Sengupta** is an Assistant Professor in the Department of Computer Science and Engineering at University of Nevada, Reno (UNR). Dr. Sengupta received his Ph.D. degree from the School of Electrical Engineering and Computer Science, University of Central Florida in 2007. Prior to that, he completed his B.E. degree (First class with Hons.) in Computer Science & Engineering from Jadavpur University, India in 2002. Before joining University of Nevada, Reno, Dr. Sengupta worked as a Post-Doctorate Researcher at Stevens Institute of Technology and an Assistant Professor at City University of New York (John Jay College of Criminal Justice). He is a recipient of the NSF CAREER award. He has also held visiting researcher position at AFRL, Rome, NY.