Short Paper: Jamming-Resilient Multipath Routing Leveraging Availability-Based Correlation

Hossen Mustafa[a], Xin Zhang[b], Zhenhua Liu[a], Wenyuan Xu[a], Adrian Perrig[b]
[a] Dept. of CSE, University of South Carolina, Columbia, SC 29208, USA
{mustafah,liuz}@email.sc.edu, wyxu@cse.sc.edu
[b] CyLab, Carnegie Mellon University, Pittsburgh, PA 15213, USA
xzhang1@cs.cmu.edu, adrian@ece.cmu.edu

ABSTRACT

Jamming attacks are especially harmful to the reliability of wireless communication, as they can effectively disrupt communication. Existing jamming defenses primarily focus on repairing connectivity between adjacent nodes. In this paper, we address jamming at the network level and focus on restoring the end-to-end data delivery through multipath routing. As long as all paths do not fail concurrently, the end-to-end path availability is maintained. Prior work in multipath selection improves routing by choosing node-disjoint paths or link-disjoint paths. However, through our experiments on jamming effects using MicaZ nodes, we show that topological disjointness is insufficient for selecting fault-independent paths. Thus, we address multipath selection based on the knowledge of a path's *availability history*. Using Availability History Vectors (AHVs) of paths, we present an AHV-based Link-State (ALS) algorithm to select fault-independent paths. Our extensive simulation results validate that the ALS algorithm is effective in overcoming the jamming impact by maximizing the end-to-end availability of the selected paths.

Categories and Subject Descriptors: C.2.0 [Computer-Communication Networks]: General-Security and Protection

General Terms: Reliability, Security.

Keywords: Wireless Networks, Routing Algorithms, Jamming.

1. INTRODUCTION

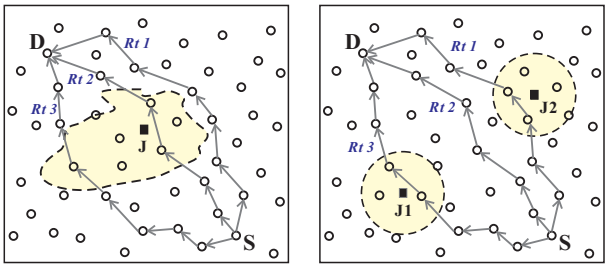
Wireless networks communicate through a shared medium and thus are vulnerable to jamming or radio interference. To cope with jamming, much research effort has focused on local repairing, i.e., restoring communication between adjacent nodes. Those anti-jamming measures include the conventional physical-layer techniques that rely on advanced transceivers [7] (e.g., frequency hopping) and MAC-layer mechanisms [5, 10] that adjust error correcting codes, channel adaptation [10], or physical location [3]. Although those techniques are important to defend against jamming, we take a different viewpoint and focus on defending against jamming at the network level, i.e., restoring reliability of end-to-end data delivery.

In this study, we examine multipath routing protocols that will react to communication disturbance on-demand. Particularly, a source node selects multiple paths for reaching the destination. When one of the paths fails, other working paths will be used to deliver packets and thereby maintain end-to-end availability, *as long as not all paths between the source and destination fail concurrently.* Such end-to-end availability provided by multiple paths between a pair of nodes is referred to as *multipath availability.* A crucial component of multipath routing is *multipath selection*, as the selection tactic and resulting path qualities will directly influence the effectiveness of multipath routing. In this study, we design multipath selection algorithms that optimize multipath availability even when one or more jammers disrupt network communication occasionally or continuously.

Most existing multipath selection algorithms [4, 9] choose node-disjoint paths or link-disjoint paths, i.e., paths without common nodes or shared links, in an attempt to minimize the probability that paths fail simultaneously. While such an approach is simple and intuitive, it relies on the assumption that the topological disjointness among multiple paths is sufficient to guarantee failure-independence. However, in a wireless network, disjoint paths can still be failure-correlated, especially in the presence of multiple interference sources. We illustrate this by two examples. Consider the two wireless ad hoc networks shown in Figure 1, where three paths between the source node S and the destination node D are disjoint. In Figure 1 (a), a stationary interferer J with an irregular jamming area becomes active occasionally. Upon turning on, J disturbs routes $Rt2$ and $Rt3$ and causes those two paths to fail concurrently. In Figure 1 (b), two interferers J_1 and J_2 are far away from each other but turning on or off synchronously. As a result, J_1 and J_2 turn routes $Rt1$ and $Rt3$ to be fault-correlated. In both cases, the disjointness is necessary but not sufficient to guarantee fault-independence between paths.

To address the failure correlation between disjoint paths in the presence of jamming, one natural way is to mathematically model the impact of jamming on the network links. However, electromagnetic signals propagate in complex environments full of absorption, reflection, scattering and diffraction, and the resulting jamming impact on the network is highly irregular [2]. Figure 2 shows packet delivery contours of a sender in the presence of a jammer

WiSec'11, June 14–17, 2011, Hamburg, Germany.



(a) (b)

Figure 1: An illustration that disjoint paths are still correlated with regard to jamming (shaded areas represent jamming regions): (a) one non-isotropic jamming area and (b) two jamming areas far apart.

obtained using MicaZ nodes, and the pink (darker shades) areas within which a receiver can successfully receive messages exhibits high irregularity. This indicates that even given accurate information of jammers' locations and jamming power levels, it is still difficult, if even possible, to quantify their impacts with reasonable accuracy.

Rather than relying on inaccurate models, our key insight is to address multipath selection based on the knowledge of a path's *availability history*, granted that failure correlation between paths can be automatically derived from their availability history. Specifically, if two paths tend to exhibit a history of concurrent failures, we regard them as failure-correlated, otherwise failure-independent. Admittedly, by using the historical failure correlation to predict future correlation, our scheme is most resilient to the types of failures that can repeat themselves in the future, while we will show that it is still effective in improving network reliability when new types of failures emerge in the future. Our underlying rationale for leveraging availability history to exploit failure-correlation lies in the following: given the intricate causes to failure-correlation between different paths, the best we can do is to derive such correlation *after* the failures take place, while it is an open challenge to detect failure-correlation *before* failures happen. In our prior work [11], we have proposed the availability history approach for the Internet. In this paper, we demonstrate that availability history is particularly effective to defend against jamming attacks in wireless networks.

The rest of this paper is organized as follows. We specify our network and threat model in Section 2. Then, we present an Availability-History-Vector (AHV)-Based Link-State (ALS) algorithm that selects multiple paths based on AHVs in Section 3, and we evaluate the ALS algorithm in our customized simulator in Section 4. Finally, we conclude the paper in Section 5.

2. NETWORK AND THREAT MODELS

In this section, we summarize the network model and the threat model for our study, and overview our approach.

2.1 Network Model

To focus our effort in examining the resilience of multipath selection against jamming and radio-interference, we consider multi-hop wireless ad hoc networks with limited mobility (e.g., wireless mesh networks). That is, the link state is primarily affected by jamming but not by the mobility of network nodes. Further, we assume that each node will

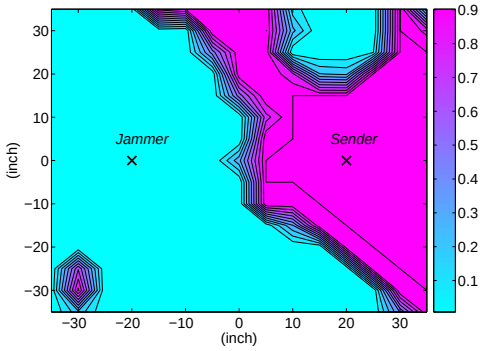


Figure 2: PDR contours of a sender located at (20,0) in the presence of a jammer located at (-20,0) to illustrate the irregularity of jamming effect in a real system. To obtain the PDR contours, a receiver was placed at the grid with a grid size of 5 inches in an indoor environment. The sender, receiver, and the jammer are all implemented on MicaZ nodes with the same transmission power levels.

maintain a neighbor table recording the link states between its neighbors and itself. Such a neighbor table is supported by most routing protocols and can be easily implemented by periodically broadcasting beacons.

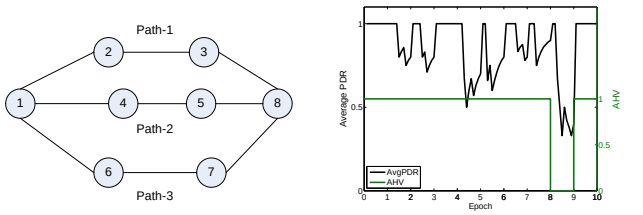
2.2 Threat Model

Besides jamming, our scheme is resilient to other types of failures or network dynamics that can change disjoint paths into fault-dependent ones. For instance, network nodes belonging to the same carrier may be far apart but leave/join the network at the same time. Nevertheless, we focus on studying the multipath selection problem under the threat of jamming. In particular, we examine the following representative jammers to mimic radio interference sources and malicious jammers.

- **Stationary Jammers.** One or more stationary jammers alternate between on and off mode but do not move around. Specifically, when interferers are active, they emit energy to the channel without following the MAC protocol implemented by the network. When multiple interferers are present, they can start to emit signals simultaneously, independently, or in a manner such that at least one of them is active at any time instant. Regardless of their activation patterns, they will disturb the network communication occasionally.
- **Mobile Jammers.** A mobile jammer will move around in the network while emitting signals continuously, and it will disrupt the network communication in its vicinity. A mobile jammer can travel following a specific pattern or can walk randomly in the network. The distinction between this model and the stationary one is that a mobile jammer, especially the jammer moving randomly, affects a wider range of links but not at the same time. As a result, it is challenging to predict the future link states with the history information when a mobile jammer is present.

3. AHV-BASED LINK-STATE ALGORITHM

The success of multipath selection necessitates two components, namely, (a) a *metric* that can accurately reflect failure correlation between different paths, and (b) a *selec-*



(a) A simplified topology (b) PDR to AHV for the link 1➞2

Figure 3: An illustration of converting PDR to AHV with topology.

tion algorithm that can effectively leverage the metric to rule out failure-correlated paths from being selected together.

In Section 3.1 we present a new mechanism which not only can evaluate individual path availability, but can also derive a multipath availability metric even in the presence of failure correlation between links. Then in Section 3.2, we describe how our mechanism helps to select multiple paths based on AHVs.

We use the following standard notations: (a) "$\wedge$" is the logical AND bit-operation; (b) "$\vee$" stands for the logical OR bit-operation; (c) "$|\mathcal{X}|$" operation returns the cardinality of the set $\mathcal{X}$; and (d) "$\|\mathcal{X}\|$" operation returns the norm of the vector $\mathcal{X}$.

3.1 Modeling Availability History

To bypass the complexity caused by precisely predicting or modeling the correlation between different paths while still capturing the failure correlation between them, we propose a new mechanism called an *Availability History Vector* (AHV), to record path availability histories, from which the failure correlation between different paths can be learned. We first define an AHV on a per-link basis, from which path (multipath) availability can be then easily derived.

AHV of A Single Link. One natural metric to determine the availability of a wireless link is the Packet Delivery Ratio (PDR), i.e., the percentage of packets successfully delivered over the link. Recording the PDR time series directly requires at least 1 byte for each data point, and calculating the aggregated PDR of a path requires multiplication. To store and compute availability history efficiently, we utilize a binary vector for recording, and bitwise operations for calculating path availability.

In particular, we map a PDR to a 0-1 value, where '1' corresponds to the time instant when the link is *available* (acceptable PDR), while '0' corresponds to the time instant when the link is *unavailable* (unacceptable PDR). A threshold γ_0 is predefined to determine whether a PDR is acceptable. Furthermore, we divide time into epochs with fixed duration. At the lth epoch, let PDR_{ij}^l be the average PDR between node i and j, then the availability record of the link between node i and j at the lth epoch is

$$r_{ij}^l = \begin{cases} 1 & \text{if } PDR_{ij}^l \geq \gamma_0, \\ 0 & \text{otherwise.} \end{cases}$$

and the AHV of this link for e epochs is $\mathbf{a}_{ij} = [r_{ij}^1, r_{ij}^2, \ldots, r_{ij}^e]$.

To facilitate observation, we depict an AHV as a continuous line and illustrate an example of converting the PDR between node 1 to 2 (as shown in Figure 3 (a)) into the AHV with γ_0 being 0.6 in Figure 3 (b); except for Epoch 9, the availability of other epochs are '1'.

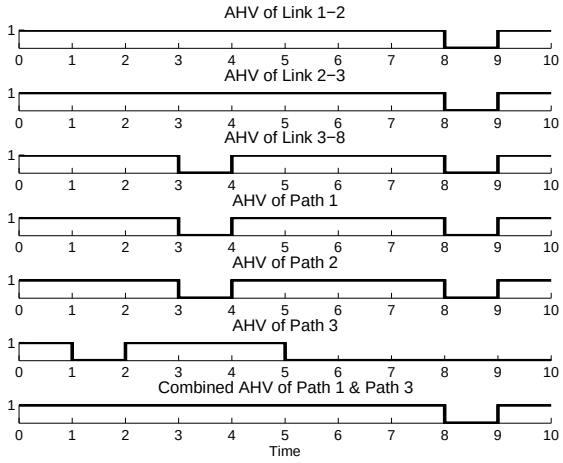


Figure 4: The AHVs for the network in Figure 3(a).

So far, AHV is used to characterize individual links. Now we present how to derive an AHV for an entire path consisting of concatenating links or sub-paths, using the following *series combination* operation.

AHV of One Path. The AHV of a complete path is computed as the logical AND bit operation of all AHVs of the links or sub-paths. The AHV of path p_i can be formulated as

$$\mathcal{A}_i = \mathbf{a}_{I_1 I_2} \wedge \mathbf{a}_{I_2 I_3} \wedge \ldots \wedge \mathbf{a}_{I_q I_{q+1}},$$

where I_q is the qth node ID on the path p_i.

For example, recall that Path-1, shown in Figure 3(a), consists of links 1➞2➞3➞8. Figure 4 illustrates the series combination for calculating its AHV. The top three lines present the AHVs of links 1➞2, 2➞3, and 3➞8, and the fourth line is the AHV of Path-1, computed as the AND bit operation of the first three AHVs.

AHV of Multiple Paths. Recall that in multipath routing, we aim at selecting multiple paths that provide the highest multipath availability; thus we derive the AHV of a given set of k paths using the following *parallel combination* operation.

Let M be the set of k paths between a source-destination pair. The AHV of M is computed as the logical OR bit operation of all AHVs of the paths, denoted as

$$\mathcal{A}_M = \mathcal{A}_1 \vee \mathcal{A}_2 \vee \ldots \vee \mathcal{A}_k.$$

Figure 4 shows an example of the AHVs of three paths along with the combined AHV of Path-1 and Path-3, obtained by a logical OR bit operation on Path-1 and Path-3's AHVs.

3.2 AHV-Based Multipath Selection

The AHV-Based Link-State (ALS) algorithm selects a few failure-independent paths utilizing the individual path AHV information. Specifically, each node maintains a history table recording the AHVs between its neighbors and itself, and such history tables for every link are accessible by the ALS algorithm. One challenge facing the ALS algorithm is that a huge number of possible paths may exist between nodes in multi-hop wireless networks. As the number of nodes increases, the number of paths will increase exponentially. Selecting multiple paths from a large number of candidate paths can be computationally prohibitive. To address this issue, we propose a two-stage framework to select multiple

paths, (a) the *path pre-selection* stage and (b) the *greedy multipath selection* stage.

In particular, we model the network as a weighted graph $G = (N, E, W)$ with N being the node set, E being the link set, and W being the map from edges to weights. Given the network graph G, h paths are selected as the candidates in the *path pre-selection* stage. Out of h candidate paths, k paths are chosen in *greedy multipath selection* stage that produce a high level of availability according to the AHVs.

3.2.1 Path Pre-Selection

The quality of the h candidate paths directly affects the achievable multipath availability to be obtained in the greedy multipath selection stage. Denote the quality of path p_i as $w(p_i)$ and denote $\mathcal{N}_i$ as the node set on the path p_i, the candidate path set H will satisfy the following requirements:

1. $|H| \leq h$,
2. $\forall p_j, p_i \in H, \frac{|\mathcal{N}_i \cap \mathcal{N}_j|}{\min(|\mathcal{N}_i|, |\mathcal{N}_j|)} \leq \rho$,
3. $\forall p_u \notin H, p_i \in H, w(p_i) \geq w(p_u)$ `OR` $\frac{|\mathcal{N}_i \cap \mathcal{N}_u|}{\min(|\mathcal{N}_i|, |\mathcal{N}_u|)} \geq \rho$,

where ρ is a threshold and $\rho \in [0, 1]$.

The second condition requires that any pair of paths belonging to H should have less than ρ percent of shared nodes. The parameter ρ controls the level of overlapping between paths in H. When $\rho = 0$, H only contains node-disjoint paths, while when $\rho = 1$, H consists of any paths without the disjointness restriction. Setting $\rho = 0$ may sound appealing at first glance, but strictly choosing h disjoint paths can filter out 'good' candidates, some of which in combination can be highly failure-independent. At the other extreme, setting $\rho = 1$ can include several candidate paths that are highly correlated with each other, reducing the diversity of H. Thus, ρ serves as a tunable value to strike the balance between those extreme cases. In our study, we choose $\rho = 0.8$.

The third condition requires that a top-ranked candidate path should be selected, unless many of its nodes overlap with higher ranking ones. The paths are ranked with regard to the link quality. We continue to use PDR as the link quality indicator, and the PDR of a path that connects node i and node j by q links is,

$$PDR_{ij} = \prod_{i=1}^{q} PDR_{I_i I_{i+1}}$$

To obtain h paths, we utilized Yen's ranking algorithm [6], a classic algorithm to determine the K shortest paths. A few issues arise when applying Yen's ranking algorithm to obtain H: (a) it assumes the end-to-end weight of two consecutive links equals the *sum* of individual link weight while the end-to-end PDR is the *product* of individual $PDRs$; (b) it returns paths with the top *minimum* weight while we are interested in top *maximum* weighted ones, and (c) it returns paths regardless of the number of shared nodes among them. To address those issues, we define the weight of a path from nodes i to j as

$$w_{ij} = -\log(PDR_{ij}) = -\sum_{i=1}^{q} \log(PDR_{I_i I_{i+1}})$$

and simply discard a path from H if it shares more than ρ percent of nodes with a higher ranking path.

3.2.2 Greedy Multipath Selection

From the availability history carried by AHVs, we can infer that two paths are highly correlated if they tend to fail at the same time in their AHVs, and vice versa. Several algorithms have been proposed to leverage such history records to cluster correlated links into groups [8]. For our purpose of selecting failure-independent routing paths, we derive a multipath availability metric from AHVs and employ a selection scheme that can preclude failure-correlated paths from being selected together without additional clustering efforts.

Multipath Availability Metric (MAM). It is computed as the number of 1-epochs (i.e., availability bit equals '1' in that epoch) in the AHV of multiple paths between a source-destination pair. Specifically, the availability of a multipath set M is denoted by $\theta(M) = |\mathcal{A}_M|$. Accordingly, our algorithm will select the k AHVs that can produce the largest MAM, by which it can ensure that failure-correlated paths are bound to be less likely chosen together.

Formally, let H be the output of path pre-selection stage. The multipath selection problem can be defined as the following:

DEFINITION 1.

$$\underset{M}{maximize} \quad \theta(M)$$
$$subject\ to \quad |M| \leq k, M \subseteq H.$$

The multipath selection problem in Definition 1 is NP-complete, according to our prior work [11]. Thus, rather than designing an algorithm to search for the optimal solution, we use an approximation algorithm (as shown in Algorithm 1) to solve the multipath selection problem. Essentially, `AHVSelect()` is a greedy algorithm. In each iteration, the algorithm greedily selects the path that can maximize the multipath availability accumulated so far, and it has a time complexity of $O(hk)$.

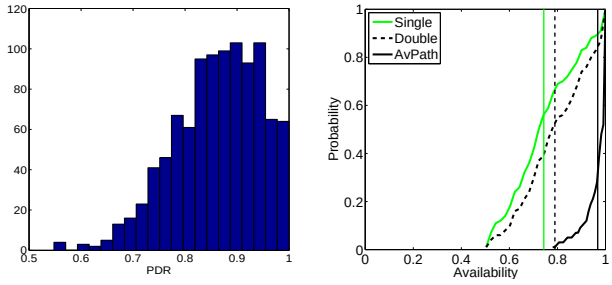
To illustrate, recall the example topology shown in Figure 3(a) with the AHV of each path given in Figure 4. Suppose Path-1 has already been selected. If we further select Path-2 which is failure-correlated with Path-1, the resulting combined AHV gains no increase in the total number of 1-epochs. In contrast, if we parallel-combine Path-3, which is failure-independent with Path-1, the resulting combined AHV benefits from a significant increase in MAM; thus it is chosen over Path-2 in our selection mechanism.

Algorithm 1 AHVSelect: AHV-Based Multipath Selection

Require: INPUT:
$H, k, \{\mathcal{A}_i\}_{i \in H}$
OUTPUT:
M;
PROCEDURES:
1: $M = \emptyset$, $\theta(M) = 0$
2: **while** $|M| \leq k$ **do**
3: *select* a path $p \in H$ that maximizes $\theta(M \cup p)$
4: *add* p to M, *update* $\theta(M) = \theta(M \cup p)$
5: **end while**

4. ALGORITHM EVALUATION

In this section, we evaluate the performance of the AHV-based Link-State algorithm in our customized simulator, which provides the flexibility of adopting various physical propagation models and hardware models for decoding pack-



(a) histogram of PDR_{SD} (b) CDF

Figure 5: The end-to-end PDR and availability in the no-jammer case.

ets. The focus is to validate the algorithm performance without the influence of network traffic.

4.1 Simulation Methodology

Propagation Model. To prepare for the extensive performance study, we chose a simple yet representative model that captures the essence of signal propagation without using computer-aided modeling tools, i.e., log-normal shadowing model [1]. Our model captures both path loss versus distance and the random attenuation due to blockage from objects in the signal path, and it has the following form,

$$PL(d) = PL(d_0) - 10 \cdot \eta \cdot \log(\frac{d}{d_0}) + X_\sigma,$$

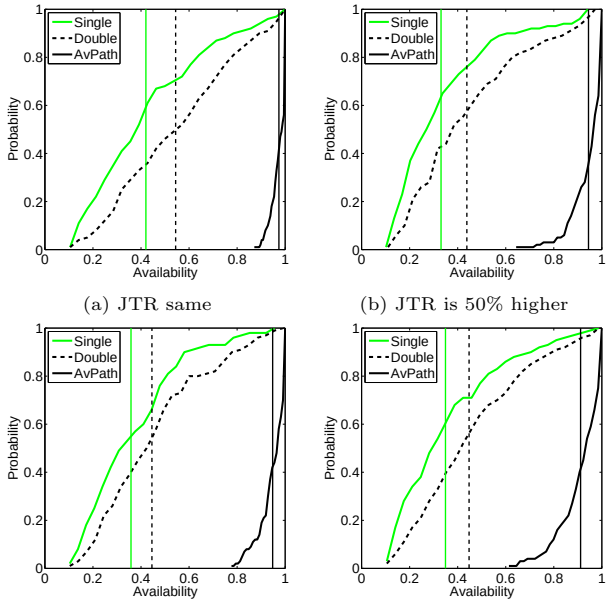
where $PL(d)$ is the path loss at distance d, $PL(d_0)$ is the known path loss at a reference distance d_0, η is the Path Loss Exponent, and X_σ is a Gaussian zero-mean random variable with standard deviation σ. To emulate a real environment, we tune the variables obtained from our prior empirical study [2]: $\eta = 2.11$, $\sigma = 1.8$, and $PL(d_0) = 33$dB.

Essentially, the link quality (PDR) between a pair of directly connected nodes is determined by the physical-layer metric, i.e., signal-to-noise ratio (SNR) at the receiver. When the SNR is larger than a threshold value ξ_o, a message can be decoded and will be received successfully, otherwise it will not. We measured the PDRs by examining the SNR for each link periodically while setting ξ_o to 0dB.

Network Setup and Scenarios. We simulated a random wireless network consisting of 1000 nodes in a 700-by-700m square. The nodes were deployed with a uniform density and each node had a transmission range of about $40m$, which resulted in approximately 10 neighbors per node.

We evaluated our ALS algorithm with $k = 2$ and compared it with two other baseline algorithms: (a) single path: selecting the path with the highest average end-to-end PDR; (b) double disjoint paths: selecting two paths that are disjoint and have the top $PDRs$. We denote those three algorithms as `AvPath`, `single`, and `double`, respectively.

We studied the following scenarios: no jammer, one stationary jammer, two stationary jammers, and a mobile jammer with two types of moving patterns. For each scenario, we ran our experiment 100 times to collect the statistical characteristics. For each simulation run, the nodes built their neighbor tables and history tables prior to time $t_1 = 600s$. At time t_1, all algorithms selected paths based on the link information observed so far. Then the average availability between a pair of nodes (S and D) that were approximately 20 hops apart was measured for 600 seconds, and the normalized availability is depicted.



(a) JTR same (b) JTR is 50% higher

(c) JTR is 100% higher (d) JTR is 200% higher

Figure 6: CDF of Availability for one jammer with different Jammer Transmission Range(JTR).

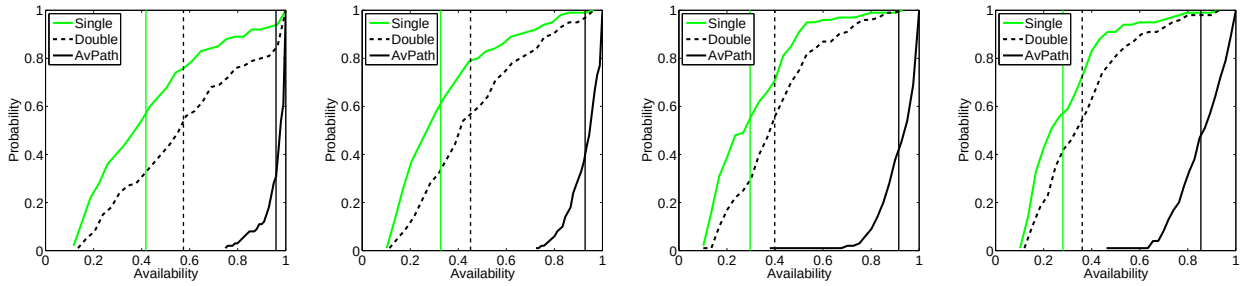
4.2 Results

No Jammer. Figure 5 shows the histogram of the end-to-end PDR and CDF (Cumulative Distribution Function) of the normalized availability between node S and node D with no active jammers. The vertical lines are the average normalized availabilities for the three cases. Overall, selecting two disjoint paths did increase the average availability slightly: by 4%. However, by selecting two fault-independent paths, our ALS algorithm boosted the average availability more: from 75% to 98%.

Stationary Jammers. In this set of experiments, we studied two scenarios: a one-stationary-jammer scenario and a two-stationary-jammer scenario. In both scenarios, the stationary jammers were present at the beginning of the simulations, and they alternated between ON and OFF for random amount of time, e.g., a random duration uniformly distributed between 5 and 20 seconds. The jammers were placed somewhere on the shortest path between nodes D and S, so that it would affect the shortest path between the source and the destination. We assumed that the jammers were capable of transmitting at a higher power level than the network nodes, and we evaluated cases when the jammer had (i) the same, (ii) 50% more, (iii) 100% more, and (iv) 200% more transmission range than the network nodes. The average availability between S and D for the one jammer case is depicted in Figure 6, and two jammer case is depicted in Figure 7. In all cases, the ALS algorithm outperforms the other two baseline algorithms by $60\% - 70\%$.

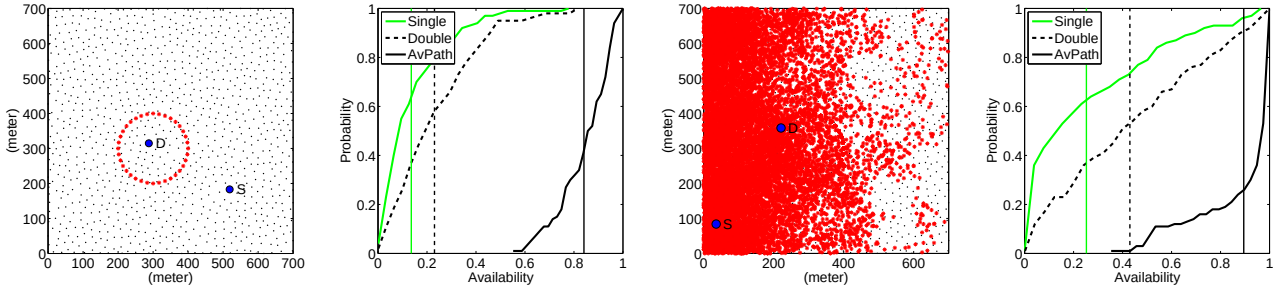
Additionally, the average availability of all algorithms decreases as the transmission range of the jammer increases, since a larger jamming range will affect more paths and will reduce the end-to-end availability.

Mobile Jammers. In our experiment, we studied two types of mobile jammers: one traveling in a circle (CW) and the other moving randomly (RW), as illustrated in Figure 8(a) and (c). Regardless of their moving patterns, both mobile jammers' transmission range is 100% more than net-



(a) JTR same (b) JTR is 50%higher (c) JTR is 100% higher (d) JTR is 200% higher

Figure 7: CDF of Availability for two jammers with different Jammer Transmission Range(JTR).



(a) jammer's positions (CW) (b) performance (CW) (c) jammer's positions(RW) (d) performance (RW)

Figure 8: Two types of mobile jammers: a circular-walk jammer (CW) and a random-walk jammer (RW).

work nodes, and both jammers remained active throughout the simulation. Specifically, the circular-walk jammer constantly disturbed communication between node D and node S, as it hovered around the destination node D (affecting node D about 40% of the time). As a result, the single-path algorithm generated a path that is only available 18% of the time, on average. The double-disjoint-path algorithm selected two paths that in combination have slightly higher availability than the single path, even though twice the number of paths were used. In comparison, the `AvPath` also selected two paths but two fault-independent paths, and thus the availability of those paths is 60% more than the other two algorithms. Finally, the RW-mobile jammer represents a jammer whose behavior is not fully captured in the availability history prior to multipath selection, i.e., it creates new 'future' failures. The simulation results in this case show that the ALS algorithm can improve the availability even when unexpected new faults may appear after multipath selection is done. This is because that both failure-dependence and the impact of jamming are partially affected by wireless network factors, such as network topologies and radio propagation environments. Although the 'future' failure has not occurred yet, its impact has already been partially encoded in the historical failure correlation implicitly.

5. CONCLUSION

We have addressed the problem of multipath selection with the goal of improving jamming resilience in wireless networks. Our key insight is to select multiple paths that are unlikely to fail concurrently, based on the knowledge of paths' availability histories. The availability histories of paths are efficiently recorded and calculated via availability history vectors (AHVs). Leveraging AHVs, we have presented AHV-based Link-State multipath selection algorithms. Our extensive simulation results have validated that (1) selecting disjoint paths is insufficient to improve end-to-end availability in the presence of jamming; and (2) AHV-based algorithms can effectively identify multiple paths that provide high end-to-end availability, even in the presence of a new jammer that has not yet affected the AHVs prior to path selection.

In summary, our AHV-based algorithms can greatly improve the end-to-end message delivery in the presence of a wide variety of jamming attacks.

6. REFERENCES

[1] A. Goldsmith. *Wireless Communications*. Cambridge University Press, New York, USA, 2005.

[2] Z. Liu, H. Liu, W. Xu, and Y. Chen. Exploiting jamming-caused neighbor changes for jammer localization. *Accepted to IEEE Transactions on Parallel and Distributed Systems (TPDS)*.

[3] K. Ma, Y. Zhang, and W. Trappe. Mobile network management and robust spatial retreats via network dynamics. In *Proceedings of workshop on Resource Provisioning and Management in Sensor Networks (RPMSN05)*, 2005.

[4] S. Mueller, R. Tsang, and D. Ghosal. Multipath routing in mobile ad hoc networks: Issues and challenges. In *Performance Tools and Applications to Networked Systems*, 2004.

[5] G. Noubir and G. Lin. Low-power DoS attacks in data wireless lans and countermeasures. *SIGMOBILE Mob. Comput. Commun. Rev.*, 7(3):29–30, 2003.

[6] M. Pascoal and E. Martins. A new implementation of yen's ranking loopless paths algorithm. *4OR: A Quarterly Journal of Operations Research*, 1(2):121–133, 2003.

[7] J. Proakis. *Digital Communications*. McGraw-Hill, Columbus, OH, 2000.

[8] A. Tachibana, S. Ano, T. Hasegawa, M. Tsuru, and Y. Oie. Locating congested segments on the internet by multiple paths' delay performance clustering. In *IEEE International Conference on Communications*. IEEE, 2007.

[9] K. Wu and J. Harms. On-demand multipath routing for mobile ad hoc networks. In *Networks EPMCC*, pages 1–7, 2001.

[10] W. Xu, W. Trappe, and Y. Zhang. Channel surfing: defending wireless sensor networks from interference. In *Proceedings of conference on Information Processing in Sensor Networks (IPSN)*, pages 499–508, 2007.

[11] X. Zhang and A. Perrig. Correlation-resilient path selection for multi-path routing. In *IEEE Globecom*, 2010.