Measures and Countermeasures for Null Frequency Jamming of On-Demand Routing Protocols in Wireless Ad Hoc Networks

M. Balakrishnan, H. Huang, R. Asorey-Cacheda, S. Misra, S. Pawar, and Y. Jaradat

***Abstract*—Distributed network protocols operate similar to periodic state machines, utilizing internal states and timers for network coordination, which creates opportunities for carefully engineered radio jamming to target the protocol operating periods and disrupt network communications. Such periodic attacks targeting specific protocol period/frequency of operation is referred to as Null Frequency Jamming (NFJ). Our hypothesis is that NFJ is a pervasive phenomenon in dynamic systems, including wireless ad-hoc networks. This paper aims to test the hypothesis by investigating NFJ targeted at the on-demand routing protocols for ad-hoc networks. Our mathematical analysis and simulation results show substantial degradation in end-to-end network throughput at certain null periods/frequencies, where the jamming periodicity self-synchronizes with the route-recovery cycle. We also study an effective countermeasure, randomized route-recovery periods, for eliminating the presence of predictable null frequencies and mitigating the impact of NFJ. Our analytical model and simulation results validate the effectiveness of randomized route recovery with appropriately chosen randomization ranges.**

***Index Terms*—Wireless ad hoc networks, routing protocols, radio jamming, denial of service (DoS).**

I. Introduction

DISTRIBUTED radio jamming [1] using a network of embedded low-power wireless nodes can reduce the effectiveness of radio-triggered improvised explosive devices (IEDs) and can inflict severe disruptions to enemy battlefield communications, while avoiding self-interference. However, scalable and energy-efficient jamming solutions are required to realize the efficiency of distributed jamming networks [1]. Null Frequency Jamming (NFJ) provides one such solution.

NFJ is based on the fact that network protocols/systems operate like finite state machines, utilizing several internal states and timers for network coordination, and that periodic operation is common among most distributed network protocols. In NFJ, jammers emit low-rate low-power periodic jamming pulses, targeting specific protocol recovery time-cycles to cause severe performance degradation. The repetition frequency of jamming pulses targets the periodic operation of specific protocol mechanisms for disrupting network communications. NFJ is energy-efficient and hard-to-detect, since the jamming pulses are short and infrequent.

Our hypothesis is that most dynamic systems, including wireless ad-hoc networks, are vulnerable to NFJ. This paper provides evidence to support our hypothesis by investigating a NFJ method targeted at the dynamic (on-demand) route recovery mechanism [2] in ad-hoc networks. Our results have high practical significance because of the pervasive usage of timer-based periodic recovery mechanism in dynamic networks.

A. Background and Related Work

Numerous Denial of Service (DoS) attacks have been proposed for ad-hoc networks [3], [4], [5], [6], [7], [8] that require the malicious nodes to monitor the network or gain access to the network by impersonation for disseminating false information (e.g. fake routes). Although such attacks have high impact on the target network, they require network monitoring or even breaking the network authentication/encryption methods. Radio jamming represents another type of DoS attack, which denies channel access to legitimate nodes. Due to the broadcast nature of wireless medium, radio jamming is easy to launch, but the effectiveness of jamming would depend on the ability to maximize the impact with minimal complexity and energy expenditure.

The simplest jamming technique is to continually transmit interference signal [5] to degrade the capacity of wireless channel. Continual jamming is energy expensive. Intelligent jamming schemes that send intermittent pulses for targeting specific transmissions have been proposed in the literature [5], [6]. For example, the technique targeted at MAC protocols [6] sends jamming pulses immediately after detecting legitimate packet transmissions to jam the acknowledgments. Such *Reactive Jamming* [5], [6] techniques involve listening to the network transmissions, deciphering information, and reacting to the subsequent network state. Reactive jamming requires continuous idle-listening and, more importantly, the capability to decode packet information, which increases the difficulty of implementation. A number of jamming and DoS attacks

Manuscript received April 12, 2011; revised December 7, 2011 and August 8, 2012; accepted August 15, 2012. The associate editor coordinating the review of this paper and approving it for publication was J. Shea.

M. Balakrishnan, H. Huang, S. Pawar, and Y. Jaradat are with the Department of Electrical & Computer Engineering, Klipsch School of Electrical & Computer Engineering, New Mexico State University, P.O. Box 30001, MSC 3-O, Las Cruces, NM 88003-8001, USA (e-mail: {mabalakr, hhuang, sandyman, yousef77}@nmsu.edu).

R. Asorey-Cacheda is with the Centro Universitario de la Defensa, Escuela Naval Militar, Plaza de España 2, 36920 Marín, Spain (e-mail: rasorey@uvigo.es).

S. Misra is with the Department of Computer Science, New Mexico State University, P.O. Box 30001, MSC CS, Las Cruces, NM 88003, USA (e-mail: misra@cs.nmsu.edu).

Digital Object Identifier 10.1109/TWC.2012.12.110678

proposed for on-demand routing protocols [3], [4], [5], [6], [7], [8] fall under the reactive jamming category.

Scheduled Jamming refers to methods where jammers are programmed to transmit jamming bursts based on a preset schedule, while being inactive the rest of the time. Scheduled jamming does not require reactive intelligence or packet decoding capabilities and is highly energy-efficient, but the effectiveness would depend on the in-depth understanding of protocol mechanisms and exploiting known vulnerabilities. In this paper, we demonstrate a scheduled jamming method, NFJ, targeted at the dynamic route-recovery process.

The concept of scheduled low-rate DoS attack was first introduced for TCP [9], [10], where the adversary periodically injects DoS packets into the network for creating congestion and degrading TCP performance. The repetition frequency (null frequency or period) of the DoS transmissions was configured to exploit the TCP retransmission time-out (RTO) cycle [11], resulting in a behavior where TCP was pushed into perpetual time-out/slow-start mode. The TCP throughput was degraded by more than 80% with only 10% attack duty-cycle. The work in [9] also demonstrated the proposed DoS attack in real TCP/IP networks (Internet).

Our work demonstrates that similar phenomena can occur with radio jamming, not just with closed-loop (TCP) but also with open-loop (UDP) flow control methods, and not just in the transport layer but also in the network layer (routing protocols). Our work also demonstrates that NFJ should be targeted at routing protocols for maximizing the impact in ad-hoc networks, since targeting transport-layer (TCP) protocols would have negligible effect if the routes were recovered dynamically before the transport layer recovery initiation.

This work does not consider robust physical-layer methods (e.g. dynamic frequency hopping [12], collaborative radio reception [13]) for improving packet reception probabilities in the presence of jamming. Given that typical communication methods are susceptible to jamming (even frequency hopping [14]), this paper deals with vulnerabilities at higher-layer network protocols.

B. Contributions

To the best of our knowledge, we are the first to investigate NFJ of on-demand routing protocols for wireless ad-hoc networks. Our research contributions are the following:

- We demonstrate the effectiveness of NFJ by targeting the dynamic route recovery procedure of on-demand routing protocols. Our investigation focuses on the two popular routing protocols, DSR [15] and AODV [16], with both TCP and UDP flows. The DSR and AODV schemes have uniform and non-uniform route recovery periodicities, respectively, and we demonstrate that NFJ can be designed for both methods. The scalability of NFJ is also evaluated in large network scenarios.
- We propose a countermeasure, based on route recovery randomization, for eliminating deterministic null frequencies and reducing the vulnerability to NFJ. We present an analytical model for predicting throughput with randomized recovery times and study the impact of randomization ranges and the type of probability distribution on the network performance.

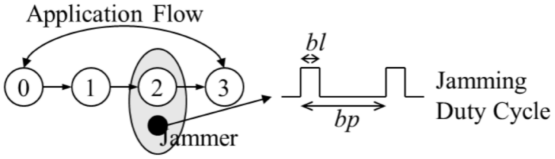


Fig. 1. Single flow scenario and jamming schedule.

Section II describes the NFJ attack in single-flow and large network scenarios. Section III studies the countermeasure by providing an analytical model of the throughput gains with randomized recovery times, along with simulation validation. Section IV concludes the paper.

II. The Routing-Targeted Null Frequency Jamming Attack

The basic idea of NFJ is to create a burst of packet losses, subsequently pushing the network protocols into the recovery mode. By synchronizing the jamming periodicity to the protocol recovery time-cycle, NFJ causes the protocol operation to be restricted in the recovery mode, which leads to network performance collapse. Higher-level network protocols are characterized by longer recovery periods and thus are more vulnerable to NFJ even with low jamming rates. This section describes NFJ at the routing layer using simulations and analytical modeling. We also demonstrate that dynamic route recovery can nullify the impact of transport layer jamming, and NFJ should be targeted at the routing protocols for substantial impact in ad-hoc networks.

A. NFJ at Network Layer: The Impact of Dynamic Route Recovery

Here, we demonstrate the NFJ of network protocols using ns-2 [17] simulations. A 3-hop single application flow is set up, which is interfered by a periodic jamming signal (Figure 1) characterized by burst-length bl and burst-period bp. The jamming is started at a random time after the application flow gets instantiated. The traffic-rate per flow is limited to 350 Kb/s. IEEE 802.11 DCF [18] is used for channel access with 7 long retries (LRC) [18] for declaring and notifying link errors to the routing protocol. As specified in [19], the bl should be long enough to create link errors, and a bl value of 100ms is large enough to accommodate up to 2×LRC retries.

Figure 2 depicts the end-to-end throughput of the application flow using the DSR protocol [15], under varying jamming periods. The results are shown with 90% confidence intervals. The performance without jamming ($bl = 0$) is also shown for baseline comparison. The DSR *RequestPeriod* [15] is set to 500ms as specified in the standard [15]. The *RequestPeriod* refers to the minimum duration between successive route discovery attempts by a node to the same destination and represents the recovery duration of the protocol. The null frequencies/periods (bp values causing significant throughput degradation) correspond to the DSR *RequestPeriod* and its factors (500ms, 250ms etc.). Figure 2 confirms that at certain jamming periods the jamming cycle self-synchronizes with the route recovery cycle to create null frequencies.

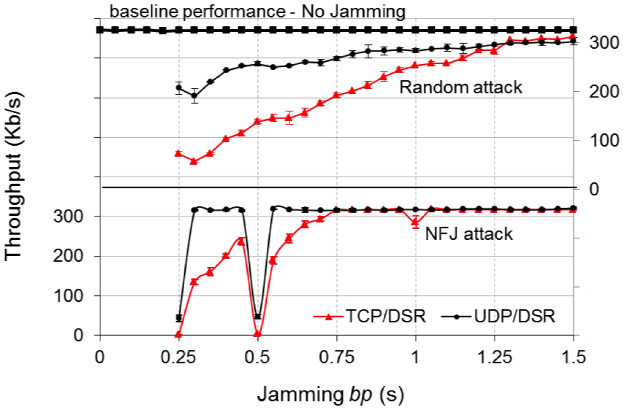


Fig. 2. Performance of DSR with TCP and UDP.

A notable behavior in Figure 2 is the absence of null frequency at the TCP minRTO value (1s), which is contradictory to the results reported in [9]. This is due to the impact of DSR route-recovery before TCP timeout. Upon detecting link error, a route-error (*RERR*) message is sent to Node-0 by Node-1 (referring to Figure 1). Node-0 immediately disseminates a route-request (*RREQ*) packet for discovering a new route to Node-3, and the data packets destined for Node-3 are queued up in DSR *SendBuffer* for *SendBufferTimeout* period of 30s [15]. The bl is long enough to jam the immediate *RREQ* broadcast from the source.

The next $RREQ$ is sent after the *RequestPeriod* and is successful, since the transmission is beyond the jamming burst. The routes are recovered shortly after 500ms, before the TCP timeout. At this point, packets from *SendBuffer* are sent to the destination, where out-of-order segments are detected. The destination node sends a sequence of TCP duplicate-acknowledgments [19], causing the TCP entity at Node-0 to retransmit the lost segments immediately, rather than waiting for the entire *minRTO* duration. The dynamic route-recovery ensures that TCP restarts after the *RequestPeriod* duration and that the 1s jamming period has no effect on TCP operation.

To show the influence of NFJ on DSR route-recovery cycle, results with UDP flows are also presented in Figure 2. The null periods occur precisely at the same bp values. Since UDP is used, it is confirmed that the null frequencies are exclusively caused by the DSR route-recovery cycle without any contribution from the transport layer.

Figure 2 also presents the results of a random jamming attack, where the bp is a random variable uniformly distributed within the interval $[p, bp]$. The same bl of 100ms has to be used for creating link-layer errors, and hence p is set to $2 \times bl$ to limit the jamming duty cycle to below 50% (larger duty cycles would result in unfair comparison, since NFJ's duty cycles are less than 40%). The throughput degradation is evident at smaller *bp* values, since certain amount of capacity is lost during jamming. However, the throughput increases as the bp is increased without any unexpected behavior. Also, UDP experiences minimal impact, since it does not slow down as a result of packet losses. The results validate that the NFJ attack has a higher impact on the network as it produces null periods (even with UDP) with much lesser jamming transmissions.

To summarize Figure 2, a jamming bp equal to *RequestPeriod* will represent the NFJ schedule for DSR protocol. The bl is long enough to jam the immediate *RREQ*dissemination, which means that the subsequent recovery attempts that will occur after a duration proportional to *RequestPeriod* will automatically align (self-synchronize) with the jamming period.

For NFJ to be effective, the jamming burst length (bl) is subject to the following constraint,

$$bl \geq T_{fd} + 2T_{rd} \tag{1}$$

where T_{fd} is the time required to determine link failures and T_{rd} is the time required for multi-hop dissemination of routing control packets between the source node and the jammed nodes. Since the bl should accommodate the dissemination of both *RERR* and *RREQ* packets, $2T_{rd}$ is used. An upper-bound estimate of T_{rd} based on the diameter of the network can be used, which will generically apply for both *RREQ* and *RERR* packets. Equation (1) provides a design guide for ensuring minimum required jamming burst-length and could be scaled up by a factor to encompass other timing factors (e.g. wait-time for non-propagating-*RREQ* [15], if supported).

B. *NFJ with Non-uniform Route Recovery Time*

In DSR, the wait duration for successive route discoveries is a multiple of *RequestPeriod* value, representing a uniform periodic recovery cycle. The NFJ period should be selected based upon the Greatest Common Divisor (GCD) of the wait durations for maximum jamming impact. In contrast to DSR, the AODV protocol [16] employs a Time To Live (TTL) expanding ring search method for route rediscoveries, generating non-uniform recovery periods. This subsection demonstrates a NFJ method for non-uniform route recovery cycles, using AODV as example.

The AODV specification [16] defines the following parameters: TTL_START, the initial propagation distance (in hop count) for *RREQ* packets, which is set to 1 or last recorded hop count. TTL_INCREMENT represents the search expansion parameter for successive *RREQ* transmissions and is set to 2 hops. TTL_THRESHOLD represents the maximum propagation/search distance for *RREQ* packets in the expanding ring search mode and is fixed at 7 hops. RING_TRAVERSAL_TIME (RTT_h) represents the wait duration for successive route discoveries in the expanding ring search mode and its value is a function of the propagation hop count (h = TTL). NET_TRAVERSAL_TIME (NTT) is a constant wait-duration in the network-wide search mode, and its value is based upon NET_DIAMETER (constant [16]). The estimation formulas for RTT_h and NTT are provided in the standard specification [16].

Upon notification of route failure, the source node broadcasts the first $RREQ$ packet with a TTL value of h = TTL_START + TTL_INCREMENT. The wait duration for route-reply is set to RTT_h [16]. The subsequent $RREQ$ broadcasts will increment the value of h by TTL_INCREMENT, thereby increasing the wait duration. If h exceeds the TTL_THRESHOLD value, then the subsequent attempts will be a network-wide search by setting the value of h to NET_DIAMETER and the wait duration to NTT [16]. During network-wide search, $RREQ$s will be sent every

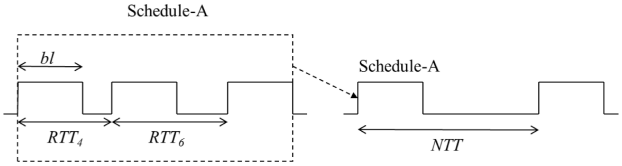


Fig. 3. NFJ schedule for AODV routing (non-uniform route recovery time).

NTT duration, and the value of NTT is doubled after every unsuccessful attempt.

Designing a NFJ schedule for the AODV protocol requires a prediction of possible route recovery periods. The first jamming burst causes route failures and also jams the immediately disseminated $RREQ$ packet. Let us consider the first jamming burst to be the route failure instant. Assuming that the routes were working before jamming, the TTL_START would be greater than or equal to 2 (minimum multi-hop route length is considered to be 2 hops) and thereby h for the first $RREQ$ packet will be greater than or equal to 4 hops. Thus, the wait duration (RTT_h) for second recovery attempt will be set to RTT_4 (480ms) or RTT_5 (560ms) or RTT_6 (640ms) or RTT_7 (720ms) or NTT (2.8s), depending on the recorded route length to the destination before the failure. If h is more than 7 hops, the NTT wait-time will be employed.

The values for RTT_h are computed using the formulations from the standard specification [16]. If no route replies were received, the second $RREQ$ packet will be sent after any one of the above-specified durations from the route failure instant. The value of h will be incremented by TTL_INCREMENT and the wait duration for third recovery attempt will be set to RTT_6 or RTT_7 or NTT. Thus, relative to the route failure instant (first burst), the third $RREQ$ packet will be disseminated after 1120ms (RTT_4 + RTT_6) or 1280ms (RTT_5 + RTT_7) or 2.8s (NTT). After the third route discovery attempt, all the cases converge to network-wide search mode where the wait duration is a multiple of NTT value.

The simplest NFJ strategy, which is to set the jamming bp to the GCD of the possible $RREQ$ broadcast instants, is an impractical solution for expanding ring search methods due to the non-uniformity in recovery periods. For example, the GCD of all the possible route recovery periods in AODV is 80ms, which is a non-feasible jamming period since 80ms bp with 100ms bl results in continuous jamming. However, a generic feature of expanding ring search algorithms is that the expansion factor converges to a constant threshold value after repeated failures. The AODV recovery process follows a non-uniform cycle during the ring search phase, but converges to the uniform NTT period after repeated unsuccessful attempts, which can be exploited using NFJ.

The NFJ schedule for the AODV protocol is shown in Figure 3. Irrespective of the initial state in the expanding ring search process, schedule-A is designed to push all flows into the NTT timeout cycle and thereby allowing a jamming period (bp) of NTT duration to synchronize with the route recovery period. The first burst jams the immediate $RREQ$ packet. The second burst should start from the earliest possible instant (RTT_4) when the second $RREQ$ could be sent and the bl should be long enough to jam the farthest possible transmission instant (RTT_7) of the second $RREQ$ in the ring search mode. The design of third burst follows similar approach depending on the possible transmission instants of third $RREQ$ packet. By the end of the third burst all flows would have progressed into the network-wide search mode. Repeating schedule-A every NTT would produce NFJ. The bl formulation is summarized below:

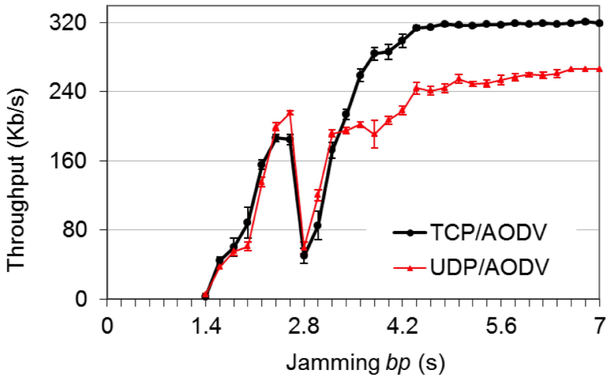


Fig. 4. AODV jamming: null frequency corresponds to NTT.

bl to jam the second $RREQ$ transmission,

$$bl_1 = RTT_7 + T_{fd} + 2T_{rd}$$

bl to jam the third $RREQ$ transmission,

$$bl_2 = (RTT_5 + RTT_7) - (RTT_4 + RTT_6) + T_{fd} + T_{rd}$$

For generalization pick the largest bl value,

$$bl = \max(bl_1, bl_2) \tag{2}$$

T_{fd} and T_{rd} are the primary constraints (explained before) and should not be ignored. Applying the standard values [16] for RTT_h in equation (2), the bl value is estimated to be approximately 300ms. The entire schedule-A (3 bursts) has to be repeated every NTT duration, since the instant when a random flow transitions into the network-wide search mode cannot be predicted.

Figure 4 depicts the performance of the AODV protocol with NFJ. Substantial throughput degradation occurs at 2.8s (NTT) and 1.4s (factor of NTT), confirming that the proposed NFJ schedule pushes the AODV route recovery process into the long timeout mode and produces null periods corresponding to NTT. The results with both TCP and UDP flows also validate that the AODV protocol can be targeted independently, irrespective of the transport layer mechanism.

C. NFJ in Large Networks

The NFJ method could be implemented as a distributed jamming network, where networks of jammers will synchronously execute the NFJ schedule for targeting large networks. However, the availability of numerous alternate routes in large networks will alter the requirements for routing-targeted NFJ. Figure 5 shows a 10×10 grid network with jamming nodes deployed either in a 1-cut (only the first cut) or 2-cut (both cuts) arrangement. Five simultaneous application flows were configured with a traffic rate of 80 to 100 Kb/s per flow. The

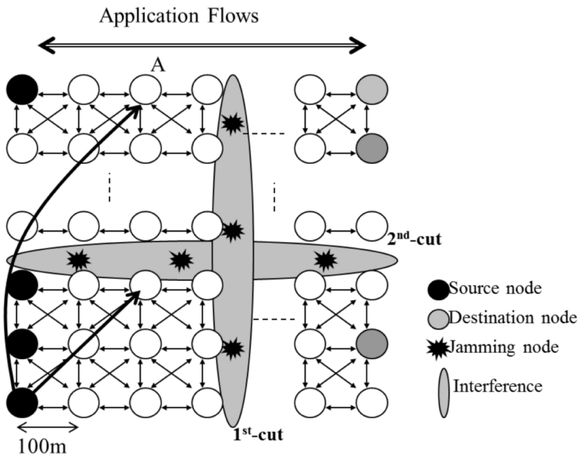


Fig. 5. Distributed network with grid topology.

source and destination nodes were selected randomly from the left and right halves of the network, respectively, such that the traffic-flows travel across the network. The path lengths varied between 5 to 10 hops.

The proposed NFJ schedules for DSR and AODV did not produce null periods (results not shown here) in the grid network scenario; the reason being insufficient bl. Referring to equation (1), the jamming bl is a function of T_{rd}, which would be substantially larger in large networks due to the availability of longer alternate routes. For example, in the scenario with only 1 cut, a flooded $RREQ$ message can traverse up to 10 hops before reaching the jammed region (location A in Figure 5). An insufficient jamming bl implies that an $RREQ$ message can take longer routes and escape the jamming burst, thereby resulting in route recovery. Although longer routes are used, null frequencies and significant performance degradation are avoided.

An upper-bound estimate of T_{rd}, for ensuring efficient NFJ in large networks, is given by

$$T_{rd} = T_{cMAC} D \tag{3}$$

where D and T_{cMAC} represent the network diameter (in hops) and one-hop delay for control packet transmissions, respectively. In 2-cut scenarios, the longest propagation path of a packet to the jammed region will be halved, and hence D in equation (3) is replaced by $D/2$.

Using ns-2 simulations, the 99^{th} percentile of T_{cMAC} and T_{fd} was estimated to be 7ms and 40ms respectively, generating an upper-bound bl estimate of 180ms and 110ms for 1-cut and 2-cut scenarios, respectively, for jamming the DSR protocol. For AODV, the jamming bl within schedule-A is proportionally increased to account for the increase in T_{rd} estimate. Figure 6 validates that, with appropriate bl, the NFJ schedule is effective in large networks and produces the same null frequencies as exhibited before. The results depict the accumulated throughput of all the flows, which validates that the AODV NFJ schedule works for all flows and is independent of the initial state of the expanding ring search process.

The impact of distributed NFJ in random network topologies is depicted in Figure 7. A hundred nodes, including 10 source/destination pairs, are deployed randomly (uniform distribution) in a 1000×1000m region. Assuming that the deployment of jammers is controllable, the impact of different jamming topologies is evaluated. In the random-jammers scenario, 10 jammers are randomly deployed in the target region. All the simulations were repeated 15 times, generating 15 different random network topologies for each NFJ scenario. The performance has high variance due to the random topologies and the location of source-destination pair (e.g. NFJ has no impact on a flow if there is no jammer in the path). Evidently, the 2-cut deployment has significantly higher impact on the network, implying higher NFJ coverage. However, the more important result is that the random NFJ deployment produced null periods in a random network. Even though the random jammers do not create a network cut, they create several routing bottlenecks (certain regions free of jamming) in the network, thereby producing null periods. This behavior expands the validity of NFJ to real networks where the topologies may not be known and the only option would be to uniformly distribute the jammers throughout the target region.

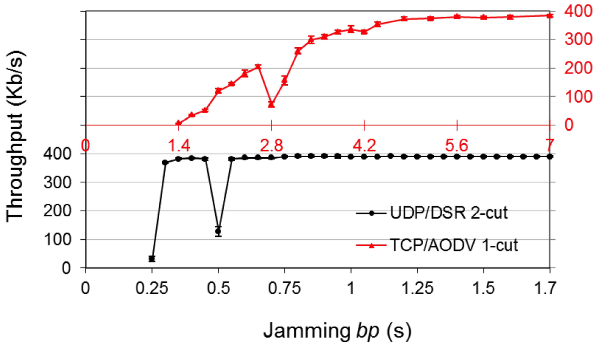


Fig. 6. Performance of NFJ in large networks.

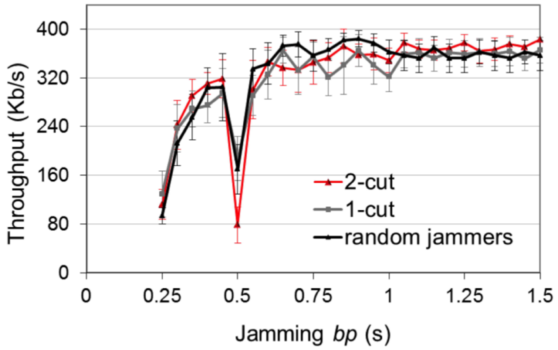


Fig. 7. NFJ in large networks: random topologies and source/destination locations.

The impact of NFJ in real networks depends on a safe (rather than accurate) upper-bound estimate of T_{rd} that will ensure a sufficiently large jamming bl to counter all timing errors (e.g recovery-time drifts, route changes). In large networks, the T_{rd} estimate is more dependent on the network diameter. In such cases, an approximate upper-bound estimate of network diameter, say M, combined with more jamming-cuts will still result in efficient NFJ.

The jammers in the NFJ network should operate synchronously. In our simulations, the network time is slotted

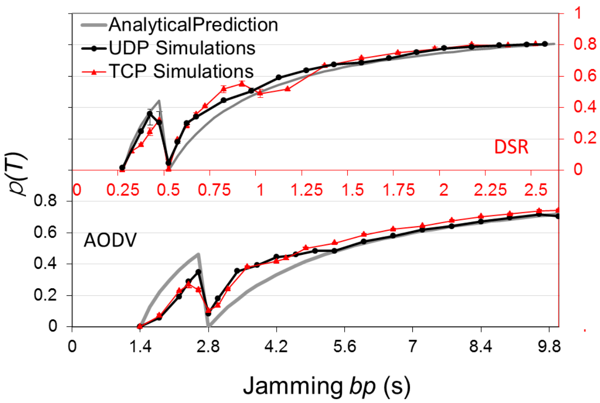


Fig. 8. Normalized network throughput: analytical vs. simulation results.

and all the jamming nodes are tied to the simulator clock (no drift). However, in real networks, the jammers should exchange timing information (e.g. using beacons) to eliminate their clock drifts, which should be reasonably simple due to the small size of the NFJ network. Also, no synchronization is required between the jammers and the target network transmissions. A jamming event will cause route failures and the network will try to recover immediately, which will automatically synchronize the network recovery cycle with the NFJ attack.

Though we have used DSR and AODV to demonstrate the attack, NFJ can be generalized to collaborative jamming-network. All the jammers in a NFJ network do not need to have a single NFJ schedule for all the protocols. In a collaborative jamming-network, different sets of jammers could run different schedules targeting different protocols. Thus, we can develop NFJ strategies for different ad-hoc protocols independently and use collaborative jamming to diminish the performance of a target network, without being confined to any specific protocol.

D. *NFJ Analytical Modeling*

Our model is an extension of the model provided in [9], and is based on the assumption that it takes a network R seconds to recover from a failure, which applies to any practical dynamic network. Suppose, the jamming process has a period of T seconds, then the number of jamming cycles N between two successful jamming events (i.e., the jamming burst hits traffic) is $\lceil R/T \rceil$. Within a time frame of NT, the network is in recovery state for R seconds, after which the network starts to send traffic until another successful jamming event at the end the time frame. So, the normalized throughput of the network with T-periodic jamming, $\rho(T)$, defined as the proportion of time that the network is functioning (roughly the ratio of maximum throughput with jamming to maximum throughput without jamming), is given by

$$\rho(T) = \frac{\left\lceil \frac{R}{T} \right\rceil T - R}{\left\lceil \frac{R}{T} \right\rceil T} \tag{4}$$

T is equivalent to the jamming burst-period (bp). For DSR, $R = RequestPeriod$, and the bl should satisfy equation (2) and for AODV, $R = NTT$, and the bl should follow schedule-A. Figure 8 compares the ns-2 simulation results with analytical predictions. The simulations were performed with the single flow topology, and the throughput results with jamming were normalized to the maximum achieved throughput without jamming. The normalized-throughput metric allows for an unbiased comparison of different protocol combinations. The analytical and simulation results agree quite well with the occurrence of null frequencies and the drops in throughput performance. Dynamic features such as packet-salvaging and the doubling of wait duration after every unsuccessful recovery attempt are not incorporated in the analytical model, which result in the differences from the simulation results. The performances converge as the jamming rate decreases (large bp) and as the effect of jamming diminishes. Despite the simplicity of our analytical model, it is able to accurately capture the high-level behavior of the network.

III. Countermeasure: Randomized Route Recovery Time

Since the NFJ attack targets the periodic route recovery process, an effective countermeasure would be to randomize the recovery periods such that the null frequencies are non-deterministic and less predictable. The TCP-targeted DoS work in [9] proposed a minRTO randomization technique for thwarting low-rate DoS attacks, but concluded that the strategy will not be effective for practical TCP networks (Internet) dominated by short-lived HTTP flows. In contrast to the conclusions made in [9], we demonstrate in this section that the randomization of route recovery wait-duration is an effective strategy for mitigating the impact of routing-targeted NFJ in wireless ad-hoc networks, with appropriate choice of randomization ranges.

A. *Analytical Model with Randomized Recovery Time*

If the route recovery time (RequestPeriod in DSR or NTT in AODV) is represented as a random variable x, then the original equation (4) can be written as

$$\rho(T) = \frac{\left\lceil \frac{x}{T} \right\rceil T - x}{\left\lceil \frac{x}{T} \right\rceil T} = 1 - \frac{x}{\left\lceil \frac{x}{T} \right\rceil T}, \ a \leq x \leq b \text{ and } 0 < a \leq b \tag{5}$$

The randomization range is defined by $[a, b]$. Let us denote $F(x)$ and $f(x)$ as the cumulative distribution function (cdf) and probability density function (pdf) of x respectively. Let us also define an integral function $g(u)$ as

$$g(u) = \int_0^a x f(x) dx, \ a \leq x \leq b \text{ and } a \leq u \leq b \tag{6}$$

Given a jamming bp of T seconds, time could be split into slots of T seconds each. Assume at the beginning of slot 1 a successful jamming event occurs. Given that the recovery time is a random variable, the probability that the next route recovery attempt occurs in any slot i is given by

$$P_i = F(iT) - F((i-1)T) \tag{7}$$

In the following, we evaluate $\rho(T)$ based on the probability distribution of the recovery time, i.e., the slot in which the network recovers. The recovery event can occur only at certain slots, since x is bounded by the interval $[a, b]$. If we define $k_1 = \lfloor a/T \rfloor$ and $k_2 = \lfloor b/T \rfloor$, then i is bounded between $[k_1+1, k_2+1]$. Equation (5) can be transformed by slot-wise integration as

$$\begin{gathered}\rho(T) = \int_a^b \left(1 - \frac{x}{\left\lceil \frac{x}{T} \right\rceil T}\right) f(x)dx \Leftrightarrow \\ \sum_{i=k_1+1}^{k_2+1} P_i \left(1 - \frac{g(iT) - g((i-1)T)}{P_i} \frac{1}{iT}\right)\end{gathered} \tag{8}$$

The expected value of x within slot i is given by $g(iT) - g((i-1)T)$, and is conditioned upon the probability of being in slot i. Expanding the first and last terms in equation (8), we get

$$\begin{gathered}\rho(T) = 1 - \frac{1}{T}\left(\frac{g(b)}{k_2+1} + \frac{g((k_1+1)T)}{k_1+1} - \frac{g(k_2T)}{k_2+1} + \right. \\ \left. \sum_{i=k_1+2}^{k_2} \frac{g(iT) - g((i-1)T)}{i}\right)\end{gathered} \tag{9}$$

$g(k_1T)$ and $g((k_2+1)T)$ is replaced by $g(a)$ and $g(b)$ respectively, since $f(x)$ is invalid outside the bound $[a, b]$. By substituting 0 for $g(a)$, we get equation (9). Further expansion and generalization of equation (9) yields equations (10) and (11).

$$\begin{aligned}\rho(T) =& 1 - \frac{1}{T}\left(\frac{g(b)}{k_2+1} + \frac{g((k_1+1)T)}{(k_1+1)(k_1+2)}\right. \\ &+ \frac{g((k_1+2)T)}{k_1+2} + \frac{g(k_2T)}{k_2(k_2+1)} - \frac{g((k_2-1)T)}{k_2} \\ &\left. + \sum_{i=k_1+3}^{k_2-1} \frac{g(iT) - g((i-1)T)}{i}\right)\end{aligned} \tag{10}$$

$$\rho(T) = 1 - \frac{1}{T}\left(\frac{g(b)}{k_2+1} + \sum_{i=k_1+1}^{k_2} \frac{g(iT)}{i(i+1)}\right) \tag{11}$$

Given the probability distribution for x, $g(x)$ can be obtained. Equation (11) provides a model for predicting $\rho(T)$ with route recovery time randomized within the range $[a, b]$ and with arbitrary distribution, which can be very useful for studying different randomization strategies.

B. Performance Evaluation of NFJ Countermeasure

We study the effectiveness of the proposed countermeasure by applying the technique to DSR protocol as an example. Other cases produce similar results and are not repeated here. We also implemented the randomized route recovery time in ns-2 simulations. The wait duration for successive $RREQ$ transmissions is a random variable, x, bounded by the interval $[a, b]$, with $a = RequestPeriod$ and b taking varying values. The value of a is set to the standard value of 500ms [15] and so the interval size is determined by the value of b. In simulations, the wait-duration is doubled (a is doubled) after each unsuccessful route discovery attempt and b is proportionally increased to keep the randomization range constant.

To quantify the influence of the probability distribution of x, results with uniform and exponential distributions are obtained. Given the interval size, T, and $f(x)$, equation (11) can be evaluated. Unlike uniform distribution, exponential distribution is unbounded ($x \in [0, \infty)$), so we use the truncated-exponential distribution with the following characteristics

$$\begin{gathered}f(x) = \left(\frac{\lambda}{\alpha}\right) e^{-\lambda(x-\alpha)}, a \leq x \leq b = a - \left(\frac{1}{\lambda}\right) \ln(1-\alpha) \\ \text{and } 0 < \alpha \leq 1 \\ F(x) = \left(\frac{1}{\alpha}\right)\left(1 - e^{-\lambda(x-\alpha)}\right)\end{gathered} \tag{12}$$

The truncation-parameter α can be computed for given values of λ, a, and b. Truncated exponential distribution was generated in the simulations using the inverse transformation method, with parameter λ set to $2/(a+b)$.

Figure 9 shows the analytical and simulation results with randomized recovery times, under different probability ranges and distributions. It is evident that the randomization range has a perceivable influence on the occurrence of null-frequencies, since the steep reduction in network performance at null periods is avoided with large randomization ranges. The effectiveness of NFJ is reduced to that of a random jamming attack (Figure 2). In contrast, the type of probability distribution has negligible impact, since the uniform and exponential curve patterns are similar in every graph. There are two important design considerations:

- Randomizing within a small range is ineffective since the jamming burst length (*bl*) is designed to compensate for the variations in $RREQ$ transmission times (due to MAC layer random back-off wait [18], multi-hop propagation, route changes). For avoiding null frequencies, the randomization range of route recovery time should be substantially larger in comparison with the lower-layer operating time scales. Equation (1) can also be a design guide for choosing the randomization range in proportion with the *bl* value. Also, predicting the null frequencies will be harder with larger randomization ranges.
- The value of a represents the minimum route recovery delay. For higher throughput, the randomization range should be significantly larger in comparison with the value of a. Consider the case where the jamming periodicity (*bp*) is equivalent to b, i.e., route failures occur every b time units. In this case, a smaller interval ($b - a \ll a$) implies immediate route breakages after long route recovery delays, which will cause significant degradation in the network throughput. The design approach should consider $b - a \geq a$ or $b \geq 2a$.

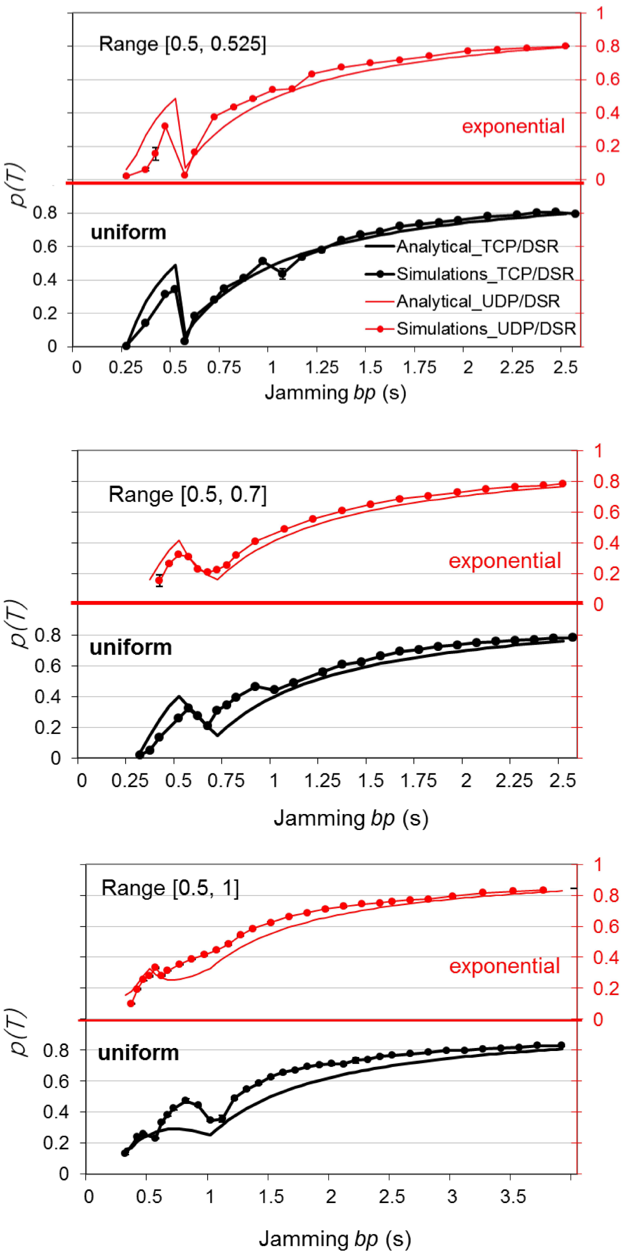


Fig. 9. Performance of DSR with randomized recovery time.

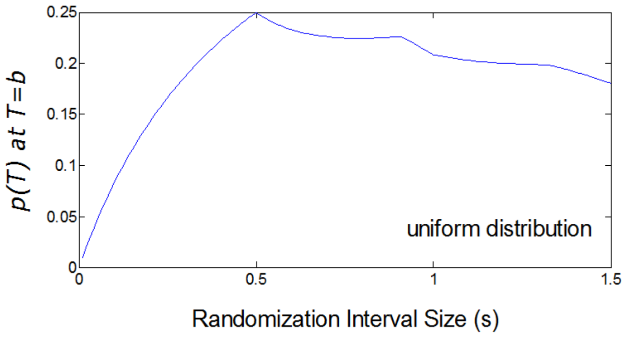


Fig. 10. Throughput at null frequency with different randomization ranges.

The analytical and simulation results in Figure 9 match quite well in terms of the overall behavior, though the values do not match precisely for the same reasons as in Figure 8. Further, recovery time randomization introduces a timing jitter relative to the jamming periodicity. The relative time difference between the jamming instant and the $RREQ$ transmission instant will cumulatively increase with subsequent $RREQ$ transmissions. The simulations replicate this drift phenomenon and thus the probability of jamming an $RREQ$ is lower in the simulations, producing higher throughput relative to the mathematical model. This phenomenon is more pronounced with larger randomization intervals due to larger time variations.

Figure 9 also depicts a marginal throughput reduction with larger randomization ranges; the reason being increased route recovery delays. At higher jamming rates, routes are broken frequently and the higher recovery delay contributes for up to 10% reduction in maximum achievable throughput. However, larger randomization ranges eliminate the null frequency performance. In the context of NFJ, the possibility of avoiding predictable null frequencies outweighs the possibility of marginal throughput reduction due to route recovery delays.

C. Selecting the Upper Limit of Randomization Range

As per Figure 9, a jamming bp equivalent to b produces the lowest throughput and indicates the location of the null frequency. The maximum network throughput at $bp = b$ increases with larger randomization ranges. However, larger ranges result in longer delays and slower route recovery process. Thus, given these two opposing effects, we identify the randomization range limit beyond which the route recovery delays will have a negative impact on the network throughput.

Figure 10 shows the value of $\rho(T)$ at the null frequency ($bp = b$), under increasing randomization ranges. After a steady rise, $\rho(T)$ starts decreasing beyond an interval size of 500ms, which represents the threshold point beyond which the route-recovery delays will cause perceivable degradation to the network throughput. In the specific case of DSR, the range $[0.5, 1]$, i.e $b = 2a$, seems to the ideal randomization range for achieving the maximum possible throughput with b-periodic jamming. This effect of randomization range is independent of the targeted protocol, though the optimal range might vary depending on the recovery duration of the protocols.

IV. Conclusions

This paper introduced a low-rate radio jamming method, NFJ, for wireless ad-hoc networks that targeted the on-demand routing protocols, independent of transport layer mechanisms. As we demonstrated, NFJ caused significant throughput losses in the network without using sophisticated intelligence. We also showed that, in ad-hoc networks, targeting transport-layer protocols would have minimal impact due to dynamic route recovery and that routing protocols should be targeted for substantial impact. NFJ has high practical significance because of the pervasive usage of timer-based periodic recovery mechanisms in dynamic networks. This paper further studied a countermeasure for NFJ, randomized route recovery time, and identified the need to randomize protocol time constants over appropriate ranges to eliminate the presence of predictable null frequencies and to reduce the vulnerability to NFJ attacks.

ACKNOWLEDGMENTS

This work was supported in part by the U.S. National Science Foundation under Grant 1248109 and the project CALM (TEC2010-21405-C02-01), funded by MICINN, Spain.

REFERENCES

[1] N. Ahmed and H. Huang, "Distributed jammer networks: impact and characterization," in *Proc. 2009 IEEE Military Communications Conference*.

[2] C. S. R. Murthy and B. S. Manoj, "Routing protocols for ad hoc wireless networks," *Ad Hoc Wireless Networks: Architectures and Protocols*. Prentice Hall PTR, 2008, ch. 7, pp. 300–333.

[3] K. Pelechrinis, M. Iliofotou, and S. V. Krishnamurthy, "Denial of service attacks in wireless networks: the case of jammers," *IEEE Commun. Surveys and Tutorials*, vol. PP, no. 99, pp. 1–13, May 2010.

[4] D. M. Shila, Y. Cheng, and T. Anjali, "Mitigating selective forwarding attacks with a channel-aware approach in WMNs," *IEEE Trans. Wireless Commun.*, vol. 9, no. 5, pp. 1661–1675, May 2010.

[5] W. Xu, W. Trappe, Y. Zhang, and T. Wood, "The feasibility of launching and detecting jamming attacks in wireless networks," in *Proc. 2005 ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pp 46–57.

[6] E. Bayrataroglu, *et al.*, "On the performance of IEEE 802.11 under jamming," in *Proc. 2008 IEEE International Conference on Computer Communications*, pp. 1265–1273.

[7] R. H. Jhaveri *et al.*, "MANET routing protocols and wormhole attack against AODV," in *International J. Computer Science and Network Security*, vol. 10, no. 4, Apr. 2010.

[8] M. Khabbazian, H. Mercier, and V. K. Bhargava, "Severity analysis and countermeasure for the wormhole attack in wireless ad hoc networks," *IEEE Trans. Wireless Commun.*, vol. 8, no. 2, pp. 736–745, Feb. 2009.

[9] A. Kuzmanovic and E. Knightly, "Low-rate TCP-targeted denial of service attacks and counter strategies," *IEEE/ACM Trans. Networking*, vol. 14, no. 4, pp. 683–696, Aug. 2006.

[10] I. Aad, J.-P. Hubaux, and E. Knightly, "Impact of denial of service attacks on ad hoc networks," in *IEEE/ACM Trans. Networking*, vol. 16, no. 4, pp. 791–802, Aug. 2008.

[11] IETF TCP Retransmission Timer Specification, RFC 2988, Nov. 2000.

[12] H. Wang, L. Zhang, T. Li, and J. Tugnait, "Spectrally efficient jamming mitigation based on code-controlled frequency hopping," *IEEE Trans. Wireless Commun.*, vol. 10, no. 3, pp. 1536–1276, Mar. 2011.

[13] J. Moon, J. M. Shea, and T. F. Wong, "Collaborative mitigation of partial-time jamming on nonfading channels," in *IEEE Trans. Wireless Commun.*, vol. 5, no. 6, pp. 1371–1381, June 2006.

[14] K. Pelechrinis, C. Koufogiannakis, and S. V. Krishnamurthy, "On the efficacy of frequency hopping in coping with jamming attacks in 802.11 networks," in *IEEE Trans. Wireless Commun.*, vol. 9, no. 10, pp. 3258–3271, Oct. 2010.

[15] IETF Dynamic Source Routing Specification, RFC 4728, Feb. 2007.

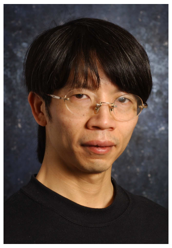
[16] IETF Ad hoc On-Demand Distance Vector Routing Specification, RFC 3561, July 2003.

[17] Network Simulator, http://www.isi.edu/nsnam/ns.

[18] IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, 2007.

[19] IEFT TCP Congestion Control Specification, RFC 2581, Apr. 1999.

Manikanden Balakrishnan received his Ph.D. in 2007 from New Mexico State University. His expertise includes computer and wireless networks, software development, protocols, and standards. His previous research works in wireless and sensor networks were related to applications in military HF radio, home-multimedia, radio jamming, and health monitoring. He is currently a staff systems engineer within the research department at ASSIA Incorporated, where he is investigating dynamic management and performance optimization solutions for Wi-Fi networks.



Hong Huang received his B.E. degree from Tsinghua University, Beijing, China, and M.S. and Ph.D. degrees from Georgia Institute of Technology in 2000 and 2002, respectively, all in electrical engineering. He is currently an associate professor with the Klipsch School of Electrical and Computer Engineering at the New Mexico State University. His current research interests include wireless sensor networks, mobile ad hoc networks, network security, and optical networks. He is a member of IEEE.

Rafael Asorey-Cacheda received the M.Sc. degree in Telecommunication Engineering (major in Telematics and Best Master Thesis Award) and the Ph.D. Degree (cum laude and Best PhD Thesis Award) in Telecommunication Engineering from the University of Vigo, Spain, in 2006 and 2009 respectively. He was a researcher with the Information Technologies Group, University of Vigo, Spain until 2009. Between 2008 and 2009 he was also R&D Manager at Optare Solutions, a Spanish telecommunications company. Between 2009 and 2012 held an Ángeles Alvariño position, Xunta de Galicia, Spain. Currently, he is a professor at the Defense University Center, University of Vigo. His interests include content distribution, high-performance switching, video transcoding, peer-to-peer networking and wireless networks.

Satyajayant Misra (SM'2005, M'2009) received his Ph.D. in Computer Science from Arizona State University in 2009 and is an assistant professor in Computer Science at New Mexico State University. His research interests include anonymity, security, and survivability in wireless networks, and protocol design for supercomputing and smart grid architectures. He serves on the editorial boards of the IEEE COMMUNICATIONS SURVEYS AND TUTORIALS and the *IEEE Wireless Communications Magazine*; he was the TPC Vice-Chair of Information Systems for IEEE INFOCOM 2012, and has served on the executive committees of IEEE SECON 2011 and IEEE IPCCC 2010. He has published over 30 peer-reviewed IEEE journals and conference articles. One of his co-authored papers was a runner-up to the best-paper award at IEEE ICNP 2010.

Sandeep Pawar and **Yousef Jaradat** are Ph.D. students at New Mexico State University doing research in wireless ad hoc networks and wireless sensor networks.