IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 16, Issue 3, Ver. IX (May-Jun. 2014), PP 61-72
www.iosrjournals.org

Jamming-Aware Traffic Allocation for Multiple-Path Routing Using Portfolio Selection against DDOS.

Yogesh R. Surve[1], Nilashree N. Shirodkar[2], Ved B. Pansare[3], Sourabh V. Kolekar[4].
[1,2,3,4](Computer Engineering, DYPCOE, Ambi/ University of Pune, India)

Abstract: *Multiple-path routing protocols allow data source node to distribute the total traffic among available paths. We consider the problem of jamming-aware source routing in which the source node performs traffic allocation based on empirical jamming statistics at individual network nodes. We formulate this traffic allocation as a lossy network flow optimization problem using portfolio selection theory. This centralized optimization problem can be solved using a distributed algorithm based on decomposition in NUM. Along with this Network based attacks have become a serious threat to the critical information infrastructure. There are many security attacks in MANET e.g. DDOS. The effect of DDOS leads to routing load, packet drop rate, end to end delay in network. So considering these parameters we build secure IDS to detect such attack and block it. Identifying the source of the attackers is necessary to correlate the incoming and outgoing flows or connections. To resist attempts at correlation, the attacker may encrypt or otherwise manipulate the connection traffic. Timing based correlation approaches are subject to timing perturbations that may be deliberately introduced by the attacker at stepping stones. So watermark-based correlation scheme is proposed which is designed specifically to be robust against timing perturbations. Unlike most previous timing based correlation approaches, our watermark-.based approach is "active" in that it embeds a unique watermark into the encrypted flows by slightly adjusting the timing of selected packets.*

Keywords: *DDOS-Distributed Denial Of Service, IDS- Intrusion Detection System ,MANET-Mobile Ad-hoc Network ,NUM – Network Utility Maximization.*

I. Introduction

In ad hoc networks, routing protocols are responsible for delivering packets between nodes not within broadcast range. This requires the use of cooperative intermediate nodes that are able to act as routers in a distributed manner, thus allowing for data packets to be forwarded towards their destination. Ad hoc network routing protocols may be classified based upon how they determine routes into three groups: proactive, reactive and hybrid. In this section, routing protocols are briefly described with an emphasis on how they disseminate control information and perform route discovery.

Proactive routing was the first attempt at designing routing protocols for MANETs. Early generation proactive protocols such as DSDV and GSR were based on the traditional distance vector and link state algorithms, which were originally proposed for wired networks. These protocols periodically maintain and distribute route information to all nodes within the network. The disadvantage of these strategies was their lack of scalability due to exceedingly large overhead produced due to blind flooding. Blind flooding is shown to result in the Broadcast Storm Problem and is thus not efficient. Other proactive routing protocols such as Fisheye State Routing (FSR) limit the rate at which they update route information depending on the distance. Routes to closer nodes are maintained more regularly, whereas routes to remote nodes are maintained less regularly. Source-Tree Adaptive Routing (STAR) eliminates periodic dissemination of control information in favour of conditional dissemination, thus reducing the constant overhead. However, blind flooding is still required. In Cluster-head Gateway Switch Routing (CGSR) a hierarchy is created based upon node clustering. Cluster heads control the flow of route information within their cluster and between clusters, thus reducing the amount of route information and limiting the dissemination of route information. More recent attempts at reducing control overhead in proactive routing can be seen in protocols such as OLSR and TBRPF. These protocols attempt to reduce the control overhead by reducing the number of rebroadcasting nodes in the network through optimised flooding.

Reactive (on-demand) routing protocols attempt to reduce the amount of control overhead disseminated in the network by determining routes to a destination only when it is required. This is usually achieved through a two-phase route discovery process initiated by a source node. The first phase of route discovery starts by the propagation of Route Request (RREQ) packets throughout the network using a simple Blind flooding approach. The second phase is initiated when a RREQ packet reaches a node, which is the destination or has a route to the destination, in which case a Route Reply (RREP) packet is generated and transmitted back to the source node. Reactive routing protocols produce significantly lower amounts of routing overhead when compared with

proactive routing protocols when the numbers of flows in the network are low. However, for large number of flows reactive protocols experience a significant drop in data throughput. This is because routing control packets are usually blind flooded (globally) throughout the entire network to find a route to the destination resulting in the Broadcast Storm Problem.

Network based attacker's makes a serious threat in a wide computer era, to stop or repel network-based attacks,

it is critical to be able to identify the source of the attack. Attackers, however, go to some lengths to conceal their identities and origin, using a variety of countermeasures. As an example, they may spoof the IP source address of the attack traffic. Methods of tracing spoofed traffic, generally known as IP trace back have been developed to address this countermeasure. Another common and effective countermeasure used by network- based intruders to hide their identity is to connect through a sequence of intermediate hosts, or stepping stones, before attacking the final target. For example, an attacker at host A may Telnet or SSH into host B, and from there launch an attack on host C. In effect, the incoming packets of an attack connection from A to B are forwarded by B, and become outgoing packets of a connection from B to C. The two connections or flows are related in such a case. The victim host C can use IP trace back to determine the second flow originated from host B, but trace back will not be able to correlate that with the attack flow originating from host A. In particular, the attacker can perturb the timing characteristics of a connection by selectively or randomly introducing extra delays when forwarding packets at the stepping stones. This kind of timing perturbation will adversely affect the effectiveness of any timing-based correlation. Timing perturbation can either make unrelated flows have similar timing characteristics, or make related flows exhibit different timing characteristics. This will increase the correlation false positive rate, or decrease the correlation true positive rate, respectively.

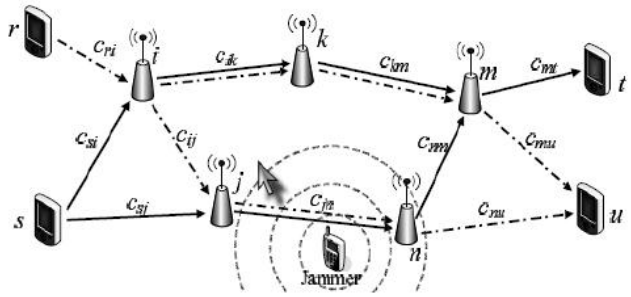


Fig. 1. An example network with sources $\mathcal{S} = \{r, s\}$ is illustrated. Each unicast link $(i, j) \in \mathcal{E}$ is labeled with the corresponding link capacity.

1.1 Problem Definition:

Particularly the fact that MP-DSR achieves a higher rate of majority of anti jamming techniques make use of diversity. For example, anti jamming protocols may employ multiple or multiple routing paths. Such diversity techniques help to curb the effects of the jamming attack by requiring the jammer to act on multiple resources simultaneously. In this paper, I consider the anti jamming diversity based on the use of multiple routing paths. Using multiple-path variants of source routing protocols such as Dynamic Source Routing (DSR) or Ad Hoc On-Demand Distance Vector (AODV) for example the MP-DSR protocol each source node can request several routing paths to the destination node for concurrent use. To make effective use of this routing diversity however, each source node must be able to make an intelligent allocation of traffic across the available paths while considering the potential effect of jamming on the resulting data through put. In order to characterize the effect of jamming on throughput, each source must collect information on the impact of the jamming attack in various parts of the network

So the problem with traditional anti jamming approach is that

Existing anti jamming techniques make use

-Only diversities techniques like multiple frequency bands, different MAC channels, or multiple routing paths.

-Lack of intelligent allocation of traffic across the available paths while considering the potential effect of jamming on the resulting data throughput.

-Lack of security in network

In ad hoc networks, routing protocols are responsible for delivering packets between nodes not within broadcast range. This requires the use of cooperative intermediate nodes that are able to act as routers in a

distributed manner, thus allowing for data packets to be forwarded towards their destination. Ad hoc network routing protocols may be classified based upon how they determine routes into three groups: proactive, reactive and hybrid. In this section, routing protocols are briefly described with an emphasis on how they disseminate control information and perform route discovery.

Proactive routing was the first attempt at designing routing protocols for MANETs. Early generation proactive protocols such as DSDV and GSR were based on the traditional distance vector and link state algorithms, which were originally proposed for wired networks. These protocols periodically maintain and distribute route information to all nodes within the network. The disadvantage of these strategies was their lack of scalability due to exceedingly large overhead produced due to blind flooding. Blind flooding is shown to result in the Broadcast Storm Problem and is thus not efficient. Other proactive routing protocols such as Fisheye State Routing (FSR) limit the rate at which they update route information depending on the distance. Routes to closer nodes are maintained more regularly, whereas routes to remote nodes are maintained less regularly. Source-Tree Adaptive Routing (STAR) eliminates periodic dissemination of control information in favor of conditional dissemination, thus reducing the constant overhead. However, blind flooding is still required. In Cluster-head Gateway Switch Routing (CGSR) a hierarchy is created based upon node clustering. Cluster heads control the flow of route information within their cluster and between clusters, thus reducing the amount of route information and limiting the dissemination of route information. More recent attempts at reducing control overhead in proactive routing can be seen in protocols such as OLSR and TBRPF. These protocols attempt to reduce the control overhead by reducing the number of rebroadcasting nodes in the network through optimized flooding.

Reactive (on-demand) routing protocols attempt to reduce the amount of control overhead disseminated in the network by determining routes to a destination only when it is required. This is usually achieved through a two-phase route discovery process initiated by a source node. The first phase of route discovery starts by the propagation of Route Request (RREQ) packets throughout the network using a simple Blind flooding approach. The second phase is initiated when a RREQ packet reaches a node, which is the destination or has a route to the destination, in which case a Route Reply (RREP) packet is generated and transmitted back to the source node. Reactive routing protocols produce significantly lower amounts of routing overhead when compared with proactive routing protocols when the numbers of flows in the network are low. However, for large number of flows reactive protocols experience a significant drop in data throughput. This is because routing control packets are usually blind flooded (globally) throughout the entire network to find a route to the destination resulting in the Broadcast Storm Problem.

II. Methodology

In this paper, we consider the anti-jamming diversity based on the use of multiple routing paths. Using multiple- path variants of source routing protocols such as Dynamic Source Routing (DSR) or Ad-Hoc On-Demand Distance Vector (AODV), for example the MPDSR protocol , each source node can request several routing paths to the destination node for concurrent use. To make effective use of this routing diversity, however, each source node must be able to make an intelligent allocation of traffic across the available paths while considering the potential effect of jamming on the resulting data throughput. In order to characterize the effect of jamming on throughput, each source must collect information on the impact of the jamming attack in various parts of the network. However, the extent of jamming at each network node depends on a number of unknown parameters, including the strategy used by the individual jammers and the relative location of the jammers with respect to each transmitter-receiver pair. Hence, the impact of jamming is probabilistic from the perspective of the network1, and the characterization of the jamming impact is further complicated by the fact that the jammers' strategies may be dynamic and the jammers themselves may be mobile-In order to capture the non-deterministic and dynamic effects of the jamming attack, we model the packet error rate at each network node as a random process. At a given time, the randomness in the packet error rate is due to the uncertainty in the jamming parameters, while the time-variability in the packet error rate is due to the jamming dynamics and mobility. Since the effect of jamming at each node is probabilistic, the end-to-end throughput achieved by each source- destination pair will also be non-deterministic and, hence, must be studied using a stochastic framework. In this article, we thus investigate the ability of network nodes to characterize the jamming impact and the ability of multiple source nodes to compensate for jamming in the allocation of traffic across multiple routing paths. Our contributions to this problem are as follow:

· We formulate the problem of allocating traffic across multiple routing paths in the presence of jamming as a lossy network flow optimization problem. We map the optimization problem to that of asset allocation using portfolio selection theory.

· We formulate the centralized traffic allocation problem for multiple source nodes as a convex optimization problem.

· We show that the multi-source multiple-path optimal traffic allocation can be computed at the source

nodes using a distributed algorithm based on decomposition in network utility maximization (NUM).

· We propose methods which allow individual network nodes to locally characterize the jamming impact and aggregate this information for the source nodes.

· We demonstrate that the use of portfolio selection theory allows the data sources to balance the expected data throughput with the uncertainty in achievable traffic rates

III. System Module And Assumptions

The wireless network of interest can be represented by a directed graph G = (N, E). The vertex set N represents the network nodes, and an ordered pair (i, j) of nodes is in the edge set E if and only if node j can receive packets directly from node i. We assume that all communication is unicast over the directed edges in E, i.e. each packet transmitted by node i ! N is intended for a unique node j ! N with (i, j) ! E. The maximum achievable data rate, or capacity, of each unicast link (i, j) ! E in the absence of jamming is denoted by the predetermined constant rate cij in units of packets per second3. Each source node s in a subset S " N generates data for a single destination node ds ! N. We assume that each source node s constructs multiple routing paths to ds using a route request process similar to those of the DSR or AODV protocols. We let Ps = {ps1, . . . , psLs} denote the collection of Ls loop-free routing paths for source s, noting that these paths need not be disjoint as in MP-DSR. Representing each path ps! by a subset of directed link set E, the sub-network of interest to source s is given by the directed sub graph Gs =!
Ns ="Ls!=1 {j : (i, j) ! ps!}, Es = "Ls !=1ps!# of the graph G.
Figure 1 illustrates an example network with sources S ={r, s}. The subgraph Gr consists of the two routing paths
pr1 = {(r, i), (i, k), (k,m), (m, u)}
pr2 = {(r, i), (i, j), (j, n), (n, u)},
and the subgraph Gs consists of the two routing paths
ps1 = {(s, i), (i, k), (k,m), (m, t)}
ps2 = {(s, j), (j, n), (n,m), (m, t)}.

In this article, we assume that the source nodes in S have no prior knowledge about the jamming attack being performed. That is, we make no assumption about the jammer's goals, method of attack, or mobility patterns. We assume that the number of jammers and their locations are unknown to the network nodes. Instead of relying on direct knowledge of the jammers, we suppose that the network nodes characterize the jamming impact in terms of the empirical packet delivery rate. Network nodes can then relay the relevant information to the source nodes in order to assist in optimal traffic allocation. Each time a new routing path is requested or an existing routing path is updated, the responding nodes along the path will relay the necessary parameters to the source node as part of the reply message for the routing path. Using the information from the routing reply, each source node s is thus provided with additional information about the jamming impact on the individual nodes.

3.1 Estimating Local Packet Success Rate

We let xij (t) denote the packet success rate over link (i, j) ! E at time t, noting that xij (t) can be computed analytically as a function of the transmitted signal power of node i, the signal power of the jammers, their relative istances from node j, and the path loss behavior of the wireless medium. In reality, however, the locations of mobile jammers are often unknown, and, hence, the use of such an analytical model is not applicable. Due to the uncertainty in the jamming impact, we model the packet success rate xij (t) as a random process and allow the network nodes to collect empirical data in order to characterize the process. We suppose that each node j maintains an estimate μij (t) of the packet success rate xij (t) as well as a variance parameter !2 ij (t) to characterize the estimate uncertainty and process variability4.

We propose the use of a recursive update mechanism allowing each node j to periodically update the estimate μij(t) as a function of time. As illustrated in Figure 3, we suppose that each node j updates the estimate μij(t) after each update period of T seconds and relays the estimate to each relevant source node s after each update relay period of Ts # T seconds. The shorter update period of T seconds allows each node j to characterize the variation in xij (t) over the update relay period of Ts seconds, a key factor in !2ij (t). We propose the use of the observed packet delivery ratio (PDR) to compute the estimate μij(t). While the PDR incorporates additional factors such as congestion, it has been shown by extensive experimentation [8] that such factors do not affect the PDR in a similar manner. Furthermore, we propose to average the empirical PDR values over time to smooth out the relatively short-term variations due to noise or fading. During the update period represented by the time interval [t − T, t], each node j can record the number rij ([t − T, t]) of packets received over link (i, j) and the number vij ([t−T, t]) % rij ([t−T, t]) of valid packets which pass an error detection check5. The PDR over link (i, j) for the update period [t−T, t], denoted PDRij ([t−T, t]), is thus equal to the ratio PDRij ([t − T, t]) =vij ([t − T, t])/rij ([t − T, t]) . (1)

This PDR can be used to update the estimate μij (t) at the end of the update period. In order to prevent significant variation in the estimate μij (t) and to include memory of the jamming attack history, we suggest using an exponential weighted moving average (EWMA) to update the estimate μij(t) as a function of the previous estimate μij (t − T) as

μij (t) = "μij (t − T) + (1 − ")PDRij ([t − T, t]), (2)

where " ! [0, 1] is a constant weight indicating the relative preference between current and historic samples.

We use a similar EWMA process to update the variance μ2ij (t) at the end of each update relay period of Ts seconds. Since this variance is intended to capture the variation in the packet success rate over the last Ts seconds, we consider the sample variance Vij ([t − Ts, t]) of the set of packet delivery ratios computed using (1) during the interval [t − Ts, t] as

Vij ([t − Ts, t]) =V ar {PDRij ([t − kT, t − kT + T]) :

k = 0, . . . , &Ts/T' − 1} . (3)

The estimation variance !2ij (t) is thus defined as a function of the previous variance !2ij(t − Ts) as !2ij (t) = #!2ij (t − Ts) + (1 − #)Vij ([t − Ts, t]), (4)

where # ! [0, 1] is a constant weight similar to " in (2). The EWMA method is widely used in sequential estimation processes, including estimation of the round-trip time (RTT) in TCP. We note that the parameters " in (2) and # in (4) allow for design of the degree of historical content included in the parameter estimate updates, and these parameters can themselves be functions "(t) and #(t) of time.

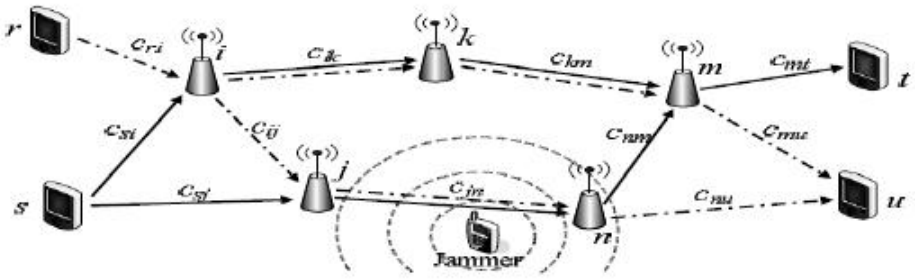


Fig. 1. An example network with sources $\mathcal{S} = \{r, s\}$ is illustrated. Each unicast link $(i, j) \in \mathcal{E}$ is labeled with the corresponding link capacity.

3.2 Characterizing The Impact Of Jamming

In this section, we propose techniques for the network nodes to estimate and characterize the impact of jamming and for a source node to incorporate these estimates into its traffic allocation. In order for a source node s to incorporate the jamming impact in the traffic allocation problem, the effect of jamming on transmissions over each link (i, j) ! Es must be estimated and relayed to s. However, to capture the jammer mobility and the dynamic effects of the jamming attack, the local estimates need to be continually updated. We begin with an example to illustrate the possible effects of jammer mobility on the traffic allocation problem and motivate the use of continually updated local estimates.

We assume that this capacity is an available constant which corresponds to the maximum packet rate for reliable transport over each wireless link. We do not address the analysis or estimation of this link capacity parameter.

Illustrating the Effect of Jammer Mobility on Network Throughput Figure 2 illustrates a single-source twork with three routing paths

p1 = {(s, x), (x, b), (b, d)},

p2 = {(s, y), (y, b), (b, d)} and

p3 = {(s, z), (z, b), (b, d)}.

The label on each edge (i, j) is the link capacity cij indicating the maximum number of packets per second (pkts/s) which can be transported over the wireless link. In this example, we assume that the source is generating data at a rate of 300 pkts/s. In the absence of jamming, the source can continuously send 100 pkts/s over each of the three paths, yielding a throughput rate equal to the source generation rate of 300 pkts/s. If a jammer near node x is transmitting at power, the probability of successful packet reception, referred to as the packet success rate, over the link (s, x) drops to nearly zero, and the traffic flow to node d reduces to 200 pkts/s. If the source node becomes aware of this effect, high the allocation of traffic can be changed to 150 pkts/s on each of paths p2 and p3, thus recovering from the jamming attack at node x. However, this one-time re-allocation by the

source node s does not adapt to the potential mobility of the jammer. If the jammer moves to node y, the packet success rate over (s, x) returns to one and that over (s, y) drops to zero, reducing the throughput to node d to 150 pkts/s, which is less than the 200 pkts/s that would be achieved using the original allocation of 100 pkts/s over each of the three paths. Hence, each node must relay an estimate of its packet success rate to the source node s and the source must use this information to reallocate traffic in a timely fashion if the effect of the attack is to be mitigated. The relay of information from the nodes can be done periodically or at the instants when the packet success rates change significantly. These updates must be performed at a rate comparable to the rate of the jammer movement to provide an effective defense against the mobile jamming attack. Next, suppose the jammer continually changes position between nodes x and y, causing the packet success rates over links (s, x) and (s, y) to oscillate between zero and one. This behavior introduces a high degree of variability into the observed packet success rates, leading to a less certain estimate of the future success rates over the links (s, x) and Fig. 3. The estimation update process is illustrated for a single link. The estimate µij (t) is updated every T seconds, and the estimation variance !2ij (t) is computed only every Ts seconds. Both values are relayed to relevant source nodes every Ts seconds. (s, y). However, since the packet success rate over link (s, z) has historically been more steady, it may be a more reliable option. Hence, the source s can choose to fill p3 to its capacity and partition the remaining 100 pkts/s equally over p1 and p2. This solution takes into account the historic variability in the packet success rates due to jamming mobility. In the following section, we build on this example, providing a set of parameters to be estimated by network nodes and methods for the sources to aggregate this information and characterize the available paths on the basis of expected throughput. B. Estimating Local Packet Success Rates We let xij (t) denote the packet success rate over link(i, j) ! E at time t, noting that xij (t) can be computed analytically as a function of the transmitted signal power of node i, the signal power of the jammers, their relative distances from node j, and the path loss behavior of the wireless medium. In reality, however, the locations of mobile jammers are often unknown, and, hence, the use of such an analytical model is not applicable. Due to the uncertainty in the jamming impact, we model the packet success rate xij (t) as a random process and allow the network nodes to collect empirical data in order to characterize the process. We suppose that each node j maintains an estimate µij (t) of the packet success rate xij (t) as well as a variance parameter !2ij (t) to characterize the estimate uncertainty and process variability4. We propose the use of a recursive update mechanism allowing each node j to periodically update the estimate µij(t) as a function of time. As illustrated in Figure 3, we suppose that each node j updates the estimate µij(t) after each update period of T seconds and relays the estimate to each relevant source node s after each update relay period of Ts # T seconds. The shorter update period of T seconds allows each node j to characterize the variation in xij (t) over the update relay period of Ts seconds, a key factor in !2ij (t).

We propose the use of the observed packet delivery ratio (PDR) to compute the estimate µij(t). While the PDR incorporates additional factors such as congestion, it has been shown by extensive experimentation that such factors 4At a time instant t, the estimate µij (t) and estimation variance !2 ij (t) define a random variable describing the current view of the packet success rate. This random variable can be appropriately modeled as a beta random variable, though the results of this article do not require such an assumption. do not affect the PDR in a similar manner. Furthermore, we propose to average the empirical PDR values over time to smooth out the relatively short-term variations due to noise or fading. During the update period represented by the time interval [t − T, t], each node j can record the number rij ([t − T, t]) of packets received over link (i, j) and the number vij ([t−T, t]) % rij ([t−T, t]) of valid packets which pass an error detection check5. The PDR over link (i, j) for the update period [t−T, t], denoted PDRij ([t−T, t]), is thus equal to the ratio PDRij ([t − T, t]) = vij ([t − T, t]) rij ([t − T, t]). (1)

This PDR can be used to update the estimate µij (t) at the end of the update period. In order to prevent significant variation in the estimate µij (t) and to include memory of the jamming attack history, we suggest using an exponential weighted moving average (EWMA) to update the estimate µij(t) as a function of the previous estimate
µij (t − T) as µij (t) = "µij (t − T) + (1 − ")PDRij ([t − T, t]), (2)

where " ! [0, 1] is a constant weight indicating the relative preference between current and historic samples. We use a similar EWMA process to update the variance !2 ij (t) at the end of each update relay period of Ts seconds.
Since this variance is intended to capture the variation in the packet success rate over the last Ts seconds, we consider the sample variance Vij ([t − Ts, t]) of the set of packet delivery ratios computed using (1) during the interval [t − Ts, t] as
Vij ([t − Ts, t]) =V ar {PDRij ([t − kT, t − kT + T]) :

k = 0, . . . , &Ts/T' − 1} . (3)
The estimation variance !2ij (t) is thus defined as a function of the previous variance !2 ij(t − Ts) as !2

ij (t) = #!2
ij (t − Ts) + (1 − #)Vij ([t − Ts, t]), (4)

where # ! [0, 1] is a constant weight similar to " in (2). The EWMA method is widely used in sequential estimation processes, including estimation of the round-trip time (RTT) in TCP [17]. We note that the parameters " in (2) and # in (4) allow for design of the degree of historical content included in the parameter estimate updates, and these parameters can themselves be functions "(t) and #(t) of time. For example, decreasing the parameter " allows the mean µij (t) to change more rapidly with the PDR due to jammer mobility, and decreasing the parameter # allows the variance !2 ij (t) to give more preference to variation in the most recent update relay period over historical variations. We further note that the update period T and update relay period Ts between subsequent updates of the parameter estimates have significant influence on the quality of the estimate. In particular, if the update period Ts is too large, the relayed estimates µij(t) and !ij (t) will be outdated before the subsequent update at time t + Ts. In the case of jamming attacks which prevent the receiving node j from detecting transmissions by node i, additional header information can be periodically exchanged between nodes i and j to achieve the convey the total number of transmissions, yielding the same overall effect. Furthermore, if the update period T at each node is too large, the dynamics of the jamming attack may be averaged out over the large number of samples rij ([t−T, t]). The update periods T and Ts must thus be short enough to capture the dynamics of the jamming attack. However, decreasing the update period Ts between successive updates to the source node necessarily increases the communication overhead of the network. Hence, there exists a trade-off between performance and overhead in the choice of the update period Ts. We note that the design of the update relay period Ts depends on assumed path-loss and jammer mobility models. The application-specific tuning of the update relay period Ts is not further herein. Using the above formulation, each time a new routing path is requested or an existing routing path is updated, the nodes along the path will include the estimates µij (t) and !2ij (t) as part of the reply message. In what follows, we show how the source node s uses these estimates to compute the end-to-end packet success rates over each path. C. Estimating End-to-End Packet Success Rates Given the packet success rate estimates µij(t) and !2ij (t) for the links (i, j) in a routing path ps!, the source s needs to estimate the effective end-to-end packet success rate to determine the optimal traffic allocation. Assuming the total time required to transport packets from each source s to the corresponding destination ds is negligible compared to the update relay period Ts, we drop the time index and address the end-to-end packet success rates in terms of the estimates µij and !2ij. The end-to-end packet success rate ys! for path ps! can be expressed as the product ys! =$(i,j)!ps!xij , (5) which is itself a random variable6 due to the randomness in each xij. We let $s! denote the expected value of ys! and %s!m denote the covariance of ys! and ysm for paths ps!, psm ! Ps. Due to the computational burden associated with in-network inference of correlation between estimated random variables, we let the source node s assume the packet success rates xij as mutually independent, even though they are likely correlated. We maintain this independence assumption throughout this work, yielding a feasible approximation to the complex reality of correlated random variables, and the case of in-network inference of the relevant correlation is left as future work. Under this independence assumption, the mean $s! of ys! given in (5) is equal to the product of estimates µij as $s! =$(i,j)!ps!µij , (6) and the covariance %s!m = E[ys!ysm] − E[ys!]E[ysm] is similarly given by %s!m =$(i,j)!ps!"psmµi$(i,j)!ps!#psm%!2ij + µ2ij&− $s!$sm.(7)

If the xij are modeled as beta random variables, the product ys! is well approximated by a eta random variable .In (7), (denotes the exclusive-OR set operator such that an element is in A (B if it is in either A or B but not both. The covariance formula in (7) reflects the fact that the endto- end packet success rates ys! and ysm of paths ps! and psm with shared links are correlated even when the rates xij are independent. We note that the variance %2 s! of the end-to-end rate ys! can be computed using (7) with & = m. Let !s denote the Ls × 1 vector of estimated end-to-end packet success rates $s! computed using (6), and let !sdenote the Ls ×Ls covariance matrix with (&,m) entry %s!m computed using (7). The estimate pair (!s,!s) provides the sufficient statistical characterization of the end-to-end packet success rates for source s to allocate traffic to the paths in Ps. Furthermore, the off-diagonal elements in !s denote the extent of mutual overlap between the paths in Ps.

3.3 Optimal Jamming-Aware Traffic Allocation

In this section, we present an optimization framework for jamming-aware traffic allocation to multiple routing paths in Ps for each source node s ! S. We develop a set of constraints imposed on traffic allocation solutions and then formulate a utility function for optimal traffic allocation by mapping the problem to that of portfolio selection in finance. Letting 's! denote the traffic rate allocated to path ps! by the source nodes, the problem of interest is thus for each source s to determine the optimal Ls×1 rate allocation vector "s subject to network flow capacity constraints using the available statistics !s and !s of the end-to-end packet success rates under jamming. A. Traffic Allocation Constraints In order to define a set of constraints for the multiple-path traffic allocation problem, we must consider the source data rate constraints, the link capacity constraints, and the reduction of

traffic flow due to jamming at intermediate nodes. The traffic rate allocation vector "s is trivially constrained to the nonnegative orthant, i.e. "s * 0, as traffic rates are non-negative. Assuming data generation at source s is limited to a maximum data rate Rs, the rate allocation vector is also constrained as 1T"s % Rs. These constraints define the convex space "sof feasible allocation vectors "s characterizing rate allocation solutions for source s. Due to jamming at nodes along the path, the traffic rate is potentially reduced at each receiving node as packets are lost. Hence, while the initial rate of 's! Is allocated to the path, the residual traffic rate forwarded by node i along the path ps! may be less than 's!. Letting p(i)s! denote the sub-path of ps! from source s to the intermediate node i, the residual traffic rate forwarded by node i is given by y(i) s! 's!, where y(i) s! is computed using (5) with ps! replaced by the sub-path p(i) s! .The capacity constraint on the total traffic traversing a link(i, j) thus imposes the stochastic constraint' s!S'!:(i,j)!ps!'s!y(i)s! % cij (8) on the feasible allocation vectors "s. To compensate for the randomness in the capacity constraint in (8), we replace the residual packet success rate y(i) s! with a function of its expected value and variance. The mean $(i) s! and variance (%(i)s!)2 of y(i) s! can be computed using (6) and (7), respectively, with ps! replaced by the sub-path p(i)s! . We thus replace y(i) s! in (8) with the statistic $(i)s! +(%(i)s! , where (* 0 is a constant which can be tuned based on tolerance to delay resulting from capacity violations7. We let Ws denote the |E|×Ls weighted link-path incidence matrix for source s with rows indexed by links (i, j) and columns indexed by paths ps!. The element w((i, j), ps!) in row (i, j) and column ps! of Ws is thus given by w ((i, j), ps!) =(min)1,$(i)s! + (%(i)s!*, if (i, j) ! ps!0, otherwise.(9) Letting c denote the |E| × 1 vector of link capacities cij for (i, j) ! E, the link capacity constraint in (8) including expected packet loss due to jamming can be expressed by the vector inequality 's!S Ws"s % c, (10) which is a linear constraint in the variable "s. We note that this statistical constraint formulation generalizes the standard network flow capacity constraint corresponding to the case of xij = 1 for all (i, j) ! E in which the incidence matrix Ws is deterministic and binary. B. Optimal Traffic Allocation Using Portfolio Selection Theory In order to determine the optimal allocation of traffic to the paths in Ps, each source s chooses a utility function Us("s) that evaluates the total data rate, or throughput, successfully delivered to the destination node ds. In defining our utility function Us("s), we present an analogy between traffic allocation to routing paths and allocation of funds to correlated assets in finance. In Markowitz's portfolio selection theory an investor is interested in allocating funds to a set of financial assets that have uncertain future performance. The expected performance of each investment at the time of the initial allocation is expressed in terms of return and risk. The return on the asset corresponds to the value of the asset and measures the growth of the investment. The risk of the asset corresponds to the variance in the value of the asset and measures the degree of variation or uncertainty in the investment's growth. We describe the desired analogy by mapping this allocation of funds to financial assets to the allocation of traffic to routing paths. We relate the expected investment return on the financial portfolio to the estimated end-to-end success rates !s and the investment risk of the portfolio to the estimated success rate covariance matrix !s. We note that the correlation between related assets in the financial portfolio corresponds to the correlation between non-disjoint routing paths. The analogy between financial portfolio selection and the allocation of traffic to routing paths is summarized below. 7The case of " = 0 corresponds to the average-case constraint and will lead to increased queueing delay whenever y(i) s! > #(i) s! . Increasing the value of " improves the robustness to variations around the mean but decreases the amount of traffic which can be allocated to the corresponding path.

As in Markowitz's theory, we define a constant risk-aversion factor ks * 0 for source s ! S to indicate the eference for source s to allocate resources to less risky paths with lower throughput variance. This risk- aversion constant weighs the trade-off between expected throughput and estimation variance. We note that each source s can choose a different risk-aversion factor, and a source may vary the risk-aversion factor ks with time or for different types of data. For a given traffic rate allocation vector "s, the expected total throughput for source s is equal to the vector inner product !Ts "s. The corresponding variance in the throughput for source s due to the uncertainty in the estimate !s is equal to the quadratic term "Ts !s"s. Based on the above analogy making use of portfolio selection theory, we define the utility function Us("s) at source s as the weighted sum Us("s) = !Ts "s − ks"Ts!s"s. (11) Setting the risk-aversion factor ks to zero indicates that the source s is willing to put up with any amount of uncertainty in the estimate !s of the end-to-end success rates to maximize the expected throughput. The role of the risk-aversion factor is thus to impose a penalty on the objective function proportional to the uncertainty in the estimation process, potentially narrowing the gap between expected throughput and achieved throughput. The cases of ks = 0 and ks > 0 are compared in detail in Section V. Combining the utility function in (11) with the set of constraints defined in Section IV-A yields the following jamming aware traffic allocation optimization problem which aims to find the globally optimal traffic allocation over the set S of sources.

Optimal Jamming-Aware Traffic Allocation

$$\phi^* = \arg\max_{\{\phi_s\}} \sum_{s\in\mathcal{S}} \gamma_s^T \phi_s - k_s \phi_s^T \Omega_s \phi_s$$

$$\text{s.t.} \quad \sum_{s\in\mathcal{S}} W_s \phi_s \leq c$$

$$1^T \phi_s \leq R_s \text{ for all } s \in \mathcal{S},$$

$$0 \leq \phi_s \text{ for all } s \in \mathcal{S}.$$

Since the use of centralized protocols for source routing may be undesirable due to excessive communication overhead in large-scale wireless networks, we seek a distributed formulation for the optimal traffic allocation problem in (12).C. Optimal Distributed Traffic Allocation using NUM In the distributed formulation of the algorithm, each sources determines its own traffic allocation "s, ideally with minimal message passing between sources. By inspection, we see that the optimal jamming-aware flow allocation problem in (12) is similar to the network utility maximization (NUM) formulation of the basic maximum network flow problem.

We thus develop a distributed traffic allocation algorithm using Lagrangian dual decomposition techniques [14]

for NUM. The dual decomposition technique is derived by decoupling the capacity constraint in (10) and introducing the link prices) ij corresponding to each link (i, j). Letting # denote the |E| × 1 vector of link prices)ij, the Lagrangian L(", #) of the optimization problem in (12) is given by L(", #) ='s!S!Ts "s − ks"Ts!s"s + #T!c −'s!SWs"s#.(13)

The distributed optimization problem is solved iteratively using the Lagrangian dual method as follows. For a given set of link prices #n at iteration n, each source s solves the local optimization problem "$s,n = argmax!s!!s%!Ts − #TnWs&"s − ks"Ts!s"s. (14)

The link prices #n+1 are then updated using a gradient descent iteration as #n+1 =!#n − a!c −'sSWs"$s,n##+, (15)

where a > 0 is a constant step size and (v)+ = max(0, v) is the element-wise projection into the non-negative orthant. In order to perform the local update in (15), sources must exchange information about the result of the local optimization step. Since updating the link prices # depends only on the expected link usage, sources must only exchange the |E| × 1 link usage vectors us, n = Ws"$ s, n to ensure that the link prices are consistently updated across all sources. The iterative optimization step can be repeated until the allocation vectors "s converge8 for all sources s ! S, i.e. when +"$ s, n − "$ s,n−1+ % * for all s with a given * > 0. The above approach yields the following distributed algorithm for optimal jamming-aware flow allocation. Distributed Jamming-Aware Traffic Allocation Initialize n = 1 with initial link prices #1.

1. Each source s independently computes!! s,
n = argmax!s"!s""Ts − #TnWs#!s − ks!Ts !s!s.
2. Sources exchange the link usage vectors
us,n = Ws!!s,n.
3. Each source locally updates link prices as
#n+1 =$ #n – a $c −! s"S us,n %%+.

4. If $!!s,n − !!s,n−1$ > " for any s, increment n and go to step 1. Given the centralized optimization problem in (12) and the above distributed formulation for jamming-aware traffic allocation, a set of sources with estimated parameters !s and !s can proactively compensate for he presence of jamming on network traffic flow. 8In order to prevent premature termination at a local minimum, sources could additionally exchange a flag fs indicating whether or not local convergence has been attained such that all sources continue to iterate until all convergence flags have been set. Computational Complexity We note that both the centralized optimization problem in (12) and the local optimization step in the distributed algorithm are quadratic programming optimization problems with linear constraints. The computational time required for solving these problems using numerical methods for quadratic programming is a polynomial function of the number of optimization variables and the number of constraints. In the centralized problem, there are + s!S |Ps| optimization variables corresponding to the number of paths available to each of the sources. The number of constraints in the centralized problem is equal to the total number of links, s!S Es|, corresponding to the number of link capacity constraints. In the distributed algorithm, each source iteratively solves a local optimization problem, leading to |S| decoupled optimization problems. Each of these problems has |Ps| optimization variables and |Es| constraints. Hence, as the number of sources in the network increases, the distributed algorithm may be advantageous in

terms of total computation time. In what follows, we provide a detailed performance evaluation of the methods proposed in this article. The wireless network of interest can be represented by a directed graph. The vertex set represents the network nodes, and an ordered pair of nodes is in the edge set if and only if node can receive packets directly from node. I assume that all communication directed edges, i.e., each packet transmitted by node is intended for a unique node with. The maximum achievable data rate, or capacity, of each uni cast link in the absence of jamming is denoted by the predetermined constant rate in units of packets per second. In this paper, I assume that the source nodes in have no prior knowledge about the jamming attack being performed.

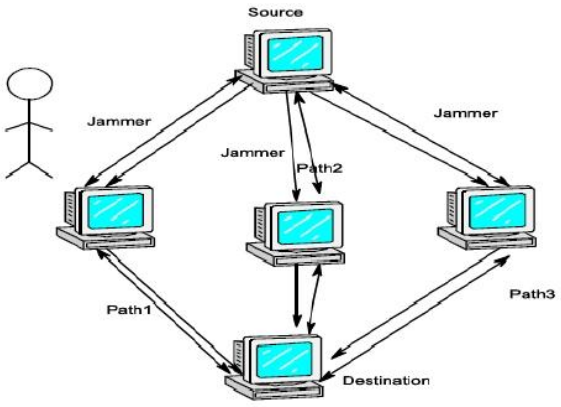
System Architecture

Instead of relying on direct knowledge of the jammers, We suppose That the Network nodes characterize the jamming Impact in terms of the Empirical Packet Delivery Rate. Network Nodes can then relay the relevant information to the source nodes in order to assist in optimal traffic allocation. Each time a new routing path is requested or an existing routing path is updated , the responding nodes along the path will relay the necessary parameters to the source node as part of the reply message for routing path. Using the information from the routing reply, Each source node is thus provided with additional information about jamming impact on the individual nodes.

IV. Indentations And Equations

In this section, we simulate various aspects of the proposed techniques for estimation of jamming impact and jamming aware traffic allocation. We first describe the simulation setup, including descriptions of the assumed models for routing path construction, jammer mobility, packet success rates, and estimate updates. We then simulate the process of computing the estimation statistics µij (t) and !2 ij (t) for a single link (i, j). Next, we illustrate the effects of the estimation process on the throughput optimization, both in terms of optimization objective functions and the resulting simulated throughput. Finally, we simulate a small-scale network similar to that in Figure 2 while varying network and protocol parameters in order to observe performance trends. A. Simulation Setup The simulation results presented herein are obtained using the following simulation setup. A network of nodes is deployed randomly over an area, and links are formed between pairs of nodes within a fixed communication range. The set S of source nodes is chosen randomly, and the destination node ds corresponding to each source s ! S is randomly chosen from within the connected component containing s. Each routing path in the set Ps is chosen using a randomized geometric routing algorithm which chooses the next hop toward the destination ds from the set of neighboring nodes that are closer to ds in terms of either distance or hop-count. Nodes transmit using fixed power Pt. We simulate the case of continuous jamming at a fixed power Pj using omnidirectional antennas. The mobility of each jammer j consists of repeatedly choosing a random direction +j ! [0, 2,) and a random speed vj ! [0, Vmax] and moving for a random amount of time -j > 0 at the chosen direction and speed. At each instant in time, the packet error rate is a function of the transmission powers

Pt and TABLE I SUMMARY OF SIMULATION PARAMETERS.



Pj, the distance dtr from the transmitter to the receiver, and the distances djr from each jammer to the receiver. The packet error rate is set equal to e−"s where s is the signal to interference and noise ratio (SINR) s = S/(I + N). The SINR is computed as a function of the received signal power S = .Ptd−# tr from the transmitter,

the received interference power I = .+j Pjd−# jr from the jammers, and the noise N at the receiver. The constant / >
0 determines the relationship between the SINR and the packet error rate, and the constants . > 0 and 0 * 2 characterize the path-loss of the wireless medium. In our simulation study, we choose parameters based on IEEE 802.15.4 and the CC2420 transceiver, and these parameters are summarized in Table I. We are interested in comparing the performance of several methods of traffic allocation using the given network and jamming models. We define the following cases of interest. Case I - Ignoring jamming: Each source s chooses the allocation vector "s using the standard maximum-flow formulation corresponding to µij = 1 and !2 ij = 0 for all links (i, j). This case is included in order to observe the improvement that can be obtained by incorporating the jamming statistics. Case II - Maximum throughput: The allocation vectors "s are chosen using the jamming-aware optimization problem in (12) with risk-aversion constant ks = 0. This case incorporates the estimates µij, updated every Ts seconds, in the allocation.
Case III - Minimum risk-return: Similar to Case II with ks > 0. This case incorporates the estimates µij and uncertainty parameters !2 ij to balance the mean throughput with the estimation variance. Case IV - Oracle model: Each source s continuously optimizes the allocation vector "s using the true values of the packet success rates xij. This impractical case is included in order to illustrate the effect of the estimation process. Our simulations are performed using a packet simulator which generates and allocates packets to paths in a fixed network according to the current value of the allocation vector "s. Each trial of the simulation compares several of the above

V. Conclusion

We studied the problem of traffic allocation in multiple-path routing algorithms in the presence of jammers whose effect can only be characterized statistically. Traffic allocation in multiple-path routing algorithms in the presence of jammers whose effect can only be characterized statistically. We have presented methods for each network node to probabilistically characterize the local impact of a dynamic jamming attack and for data sources to I presented simulation results to illustrate the impact of jamming dynamics and mobility on network throughput and to demonstrate the efficacy of our traffic allocation algorithm(NUM) and Porfolio selection. We have thus shown that multiple-path source routing algorithms can optimize the throughput. Tracing attackers' traffic through stepping stones is a challenging problem, especially when the attack traffic is encrypted, and its timing is manipulated (perturbed) to interfere with traffic analysis. The random timing perturbation by the adversary can greatly reduce the effectiveness of passive, timing-based correlation techniques. We presented a novel active timing-based correlation approach to deal with random timing perturbations. By embedding a unique watermark into the inter-packet timing, with sufficient redundancy, we can make the correlation of encrypted flows substantially more robust against random timing perturbations.

References

[1] I. F. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: A survey," Computer Networks, vol. 47, no.4, pp. 445–487, Mar. 2005.
[2] E. M. Sozer, M. Stojanovic, and J. G. Proakis, "Underwater acoustic networks," IEEE Journal of Oceanic Engineering, vol. 25, no. 1, pp. 72–83, Jan. 2000.
[3] R. Anderson, Security Engineering: A Guide to Building Dependable Distributed Systems. JohnWiley&Sons, Inc.,2001.
[4] J. Bellardo and S. Savage, "802.11 denial-of-service attacks: Real vulnerabilities and practical solutions," in Proc. USENIX Security Symposium,Washington, DC, Aug. 2003, pp. 15–28.
[5] D. J. Thuente and M. Acharya, "Intelligent jamming in wireless networks with applications to 802.11b and other networks," in Proc. 25th IEEE Communications Society Military Communications Conference (MILCOM'06), Washington, DC, Oct. 2006, pp. 1–7.
[6] A. D. Wood and J. A. Stankovic, "Denial of service in sensor networks," IEEE Computer, vol. 35, no. 10, pp. 54–62, Oct. 2002.
[7] G. Lin and G. Noubir, "On link layer denial of service in data wireless LANs," Wireless Communications and Mobile Computing, vol. 5, no. 3,pp. 273–284, May 2005.
[8] W. Xu, K. Ma, W. Trappe, and Y. Zhang, "Jamming sensor networks:Attack and defense strategies," IEEE Network, vol. 20, no. 3, pp. 41–47,May/Jun. 2006.
[9] D. B. Johnson, D. A. Maltz, and J. Broch, DSR: The Dynamic Source Routing Protocol for Multihop Wireless Ad Hoc Networks. Addison-Wesley, 2001, ch. 5, pp. 139–172.
[10] E. M. Royer and C. E. Perkins, "Ad hoc on-demand distance vector routing," in Proc. 2nd IEEE Workshop on mobile Computing Systems and Applications (WMCSA'99), New Orleans, LA, USA, Feb. 1999, pp.90–100.
[11] L. Zhang, A. G. Persaud, A. Johnson and Y. Guan, "Detectionof Stepping Stone Attack under Delay and Chaff Perturbations", In Proceedings of the 25th IEEE International Performance Computing and Communications Conference (IPCCC 2006), **(2006)** April.
[12] X. Wang, S. Chen and S. Jajodia, "Network Flow Watermarking Attack on Low-Latency Anonymous Communication Systems", In Proceedings of the 2007 IEEE Symposium on Security & Privacy (S&P 2007), **(2007)** May.
[13] B. D. Song and S. Venkataraman, "Detection of Interactive Stepping Stones: Algorithms and Confidence Bounds", In Proceedings of the 7th International Symposium on Recent Advances in Intrusion Detection (RAID 2004), Springer, **(2004)** October.
[14] P. Danzig and S. Jamin, "Tcplib: A Library of TCP Internetwork Traffic Characteristics", Technical Report USC-CS-91-495, University of Southern California, **(1991)**.
[15] P. Danzig, S. Jamin, R. Cacerest, D. Mitzel and E. Estrin, "An Empirical Workload Model for Driving Wide-Aea TCP/IP Network Simulations", Journal of Internetworking, vol. 3, no. 1, **(1992)** March, pp. 1–26.
[16] D. Donoho, et al., "Multiscale Stepping Stone Detection: Detecting Pairs of Jittered Interactive Streams by Exploiting Maximum

Tolerable DeLay", In Proceedings of the 5th International Symposium on Recent Advances in Intrusion Detection (RAID 2002): LNCS vol. 2516, Springer, **(2002)** October, pp. 17–35.
[17] M. T. Goodrich, "Efficient packet marking for large-scale ip traceback", In Proceedings of the 9th ACM Conference on Computer and Communications Security (CCS 2002), ACM, **(2002)** October, pp. 117–126.
[18] T. He and L. Tong, "Detecting Encrypted Stepping-Stone Connections", In IEEE Transactions on Signal rocessing, vol. 55, no. 5, **(2006)**, pp. 1612-1623.
[19] G. Kramer, "Generator of Self-Similar Network Traffic", http://wwwcsif.cs.ucdavis.edu/ Kramer/code/ trf gen2.html.
[20] P. Moulin, "Information-Hiding Games", In Proceedings of International Workshop on Digital Watermarking (IWDW 2003), LNCS vol. 2613, **(2003)** May.