

# A Distributed Scheme for Detection of Information Flows

Ameya Agaskar and Lang Tong  
 School of Electrical and Computer Engineering  
 Cornell University  
 Ithaca, NY  
 Email: {apa22, lt35}@cornell.edu

Ting He  
 IBM T.J. Watson Research Center  
 Hawthorne, NY  
 E-mail: the@us.ibm.com

**Abstract**—Distributed detection of information flows spanning many nodes in a wireless sensor network is considered. In such a system, eavesdroppers are deployed near several nodes in the network. As data may be encrypted or padded, the eavesdroppers can only measure packet timestamps. Each eavesdropper, given a sequence of timestamps, must compress the information for transmission to a fusion center. Given the compressed data, the fusion center must decide whether the monitored nodes are part of an information flow. Information flows may be embedded with chaff noise, and packets may be perturbed by a random but bounded delay. A specific quantizer and algorithmic detection scheme are proposed. Performance is characterized by the maximum fraction of chaff that may be inserted in an information flow while still achieving vanishing error probabilities. A lower bound on the performance of the optimal system is derived. An upper bound on the performance of a system using the proposed quantizer is also found.

**Index Terms**—Intrusion detection, Traffic analysis, Network surveillance

## I. INTRODUCTION

Consider the wireless ad hoc network illustrated in Fig. 1. Eavesdroppers have been deployed at nodes  $A$ ,  $B$ ,  $C$ , and  $D$  to measure the traffic at each node. Under the assumption that each packet is reencrypted and padded at each node, each eavesdropper is only able to measure the packet transmission times at its node. With no other information, it is impossible for any single eavesdropper to determine whether an information flow exists. Therefore, each eavesdropper must compress its timestamp data and send it over a channel to a fusion center. Given the compressed realizations from each eavesdropper, the fusion center must analyze the correlation between timestamps to determine whether an information flow exists or whether the traffic is independent. Making the problem more difficult, chaff noise, or packets that are not relayed and are thus not part of the information flow, may be inserted into the data stream. In addition, at each node, a packet may be delayed by a random amount of time before being relayed, so long as the delay is bounded by a constant.

To formulate the problem, we define a sequence of point

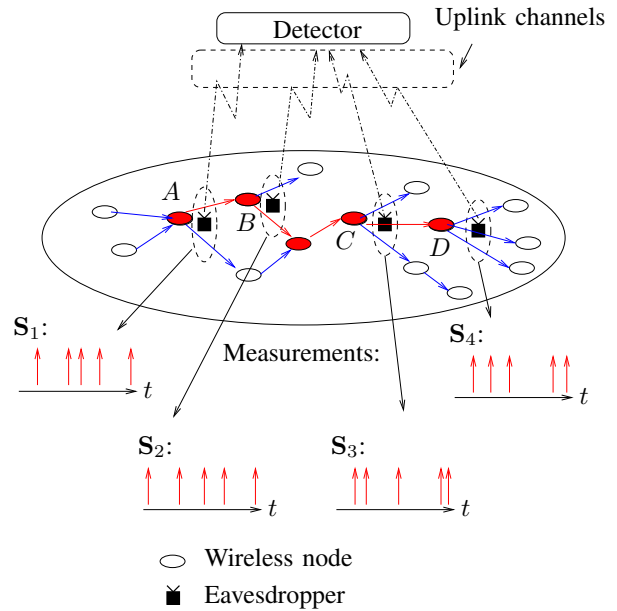


Fig. 1. In a wireless network, eavesdroppers are deployed to collect transmission activities (denoted  $\mathbf{S}_i, i = 1, \dots, N$ ) of  $N$  nodes. These processes are compressed and transmitted to a fusion center, which determines whether there is a flow from node  $A$  through nodes  $B$ ,  $C$ , and  $D$ .

processes<sup>1</sup>

$$\mathbf{S}_i = (S_i(1), S_i(2), \dots), i = 1, \dots, N, \quad (1)$$

where  $N$  is the number of nodes being measured, and  $S_i(j)$  is the  $j$ th packet measured at the  $i$ th node. We use the following definition of an information flow.

**Definition 1.1:** A sequence of processes  $(\mathbf{F}_i)_{i=1}^N$  is an *information flow with delay constraint*  $\Delta$  if and only if for every realization  $\mathbf{f}_i, i = 1, \dots, N$ , there exist bijections  $g_i : \mathcal{F}_i \rightarrow \mathcal{F}_{i+1}$  such that  $0 \leq g_i(s) - s \leq \Delta$  for all  $s \in \mathcal{F}_i$ . We say that a sequence  $(\mathbf{S}_1, \dots, \mathbf{S}_N)$  *contains an information flow* if each process can be decomposed into an information-

<sup>1</sup>We use the following notational convention: uppercase letters denote random processes, lowercase letters denote their realizations, boldface letters denote vectors, plain letters denote scalars, parentheses signify indexing, and script letters denote sets. So  $\mathbf{S}$  is a point process,  $s$  is its realization,  $S(k)$  is the  $k$ th epoch,  $s(k)$  is the realization of the  $k$ th epoch, and  $\mathcal{S}$  is the set of epochs in  $s$ .

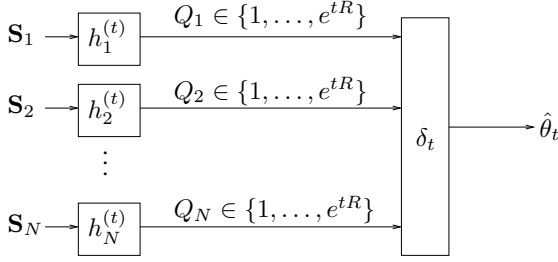


Fig. 2. A distributed detection system. The system consists of  $N$  quantizers  $h_1^{(t)}, \dots, h_N^{(t)}$  and a detector  $\delta_t$ .

carrying process  $\mathbf{F}_i$  and a chaff process  $\mathbf{W}_i$  such that<sup>2</sup>  $\mathbf{S}_i = \mathbf{F}_i \oplus \mathbf{W}_i$ ,  $i = 1, \dots, N$ , and  $(\mathbf{F}_i)_{i=1}^N$  satisfies the definition of an information flow.

We seek a distributed scheme of a form illustrated in Fig. 2 to test the following two hypotheses:

- $\mathcal{H}_0$ :  $\mathbf{S}_1, \dots, \mathbf{S}_N$  are jointly independent,
- $\mathcal{H}_1$ :  $(\mathbf{S}_1, \dots, \mathbf{S}_N)$  contains an information flow.

This is a partially non-parametric hypothesis test. Note that  $H_0$  and  $H_1$  are not complementary. This hypothesis test, though, can be used as part of an overall scheme that starts by looking for 2-node flows, then tests whether the 2-node flows are simply parts of 3-node flows, and so forth. Framing the problem as we have above greatly simplifies it. We also assume that sources are Poisson processes.

In this scheme, each eavesdropper has a quantization function  $h_i^{(t)}(\cdot)$  that maps the realization  $s_i$  over the time period  $[0, t)$  to a set of size  $\|h_i^{(t)}\|$  and transmits the results to the fusion center. The fusion center now has a sequence of random processes  $(\mathbf{Q}_i^{(t)})_{i=1}^N$  where each realization  $\mathbf{q}_i^{(t)} = h_i^{(t)}(s_i)$ ,  $i = 1, \dots, N$ . Each eavesdropper must obey a rate constraint  $R$  when transmitting data to the fusion center, so that<sup>3</sup>  $\limsup_{t \rightarrow \infty} \frac{1}{t} \log \|h_i^{(t)}\| \leq R$ ,  $i = 1, \dots, N$ .

#### A. Related Work

The problem discussed here is an example of timing analysis, which is a form of traffic analysis [1]. The related problem of stepping-stone attack detection was first considered by Staniford and Heberlein [2]. Their detection scheme involved using the actual traffic content, as did the scheme developed by Wang *et al.* [3]. These schemes were unable to deal with encrypted traffic, so Zhang and Paxson [4] considered using timing information instead of content. Donoho *et al.* allowed for timing perturbations and showed that detection could be achieved if a delay constraint were assumed [5]. He and Tong showed that a similar result could be achieved if a memory constraint were assumed [6]. Later, they showed that timing-based detectors can be designed that achieve vanishing error probabilities even when there is chaff noise proportional to

<sup>2</sup>We define the *superposition operator*  $\oplus$  to operate on the realizations of two point processes such that  $(a_k)_{k=1}^\infty \oplus (b_k)_{k=1}^\infty = (c_k)_{k=1}^\infty$  if and only if  $c_1 \leq c_2 \leq \dots$  and  $\mathcal{A} \cup \mathcal{B} = \mathcal{C}$ .

<sup>3</sup>In this paper, all logarithms are base  $e$ .

the number of packets transmitted [7]. Later, they showed that an information flow with a very large number of hops is nearly impossible to hide in any amount of chaff [8]. They first considered the problem of distributed detection of 2-hop information flows in [9]. Here, we extend that problem to an arbitrary number of nodes in a flow.

#### B. Summary of Results and Organization

We consider the distributed detection of multi-hop ( $N \geq 2$ ) information flows by timing analysis. We define the slot quantization scheme, which transmits the number of packets in successive time slots, and characterize its information rate. We define a performance measure based on the maximum fraction of chaff noise under which a detector achieves vanishing error probabilities. We introduce QBDR, an algorithm that, given slot-quantized realizations from each node, finds the minimum possible ratio of chaff packets to total packets in the flow. We define a detector that applies QBDR to the quantized realizations and compares the estimated minimum chaff-to-traffic ratio to a threshold. We characterize the detector's performance as a function of the rate constraint and the number of hops. Finally, we find upper and lower bounds for the performance of any detection scheme using slot quantizers.

The rest of the paper is organized as follows. Section II defines a performance measure. Section III defines the slotted quantization scheme. Section IV introduces a chaff assignment algorithm that will be used to quantify the amount of chaff in a set of measurements. Section V defines a detection scheme that uses the algorithm, and shows analytical bounds on the performance of detection schemes using slotted quantizers. Section VI and the results of numerical simulations of the detector. Section VII concludes the paper with remarks.

## II. PERFORMANCE MEASURES

Any detector must be able to find information flows embedded in some amount of chaff. We can say that a detector is "better" if it can tolerate more chaff noise. First, we must quantify the chaff noise in a process using the concept of the *chaff-to-traffic-ratio* (CTR) defined in [8] and [9].

*Definition 2.1:* Given a realization of an information flow  $(\mathbf{f}_i)_{i=1}^N$  and chaff noise  $(\mathbf{w}_i)_{i=1}^N$ , the CTR is

$$\text{CTR}(t) \equiv \frac{\sum_{i=1}^N |\mathcal{W}_i \cap [0, t]|}{\sum_{i=1}^N |(\mathcal{W}_i \cup \mathcal{F}_i) \cap [0, t]|}$$

$$\text{CTR} \equiv \limsup_{t \rightarrow \infty} \text{CTR}(t) \quad (2)$$

In words,  $\text{CTR}(t)$  is the fraction of total traffic packets in  $[0, t]$  that are chaff. CTR is the asymptotic fraction of chaff packets.

We can use this definition and the notion of Chernoff consistency to characterize the performance of a detector. This measure is stated in the following definition:

*Definition 2.2:* A detector  $\delta_t$  is called  $r$ -consistent ( $r \in [0, 1]$ ) if the false alarm probability  $P_F(\delta_t)$  and the miss probability  $P_M(\delta_t)$  satisfy

- 1)  $\lim_{t \rightarrow \infty} P_F(\delta_t) = 0$  for any  $(\mathbf{S}_i)_{i=1}^N$  under  $\mathcal{H}_0$ ;

2)  $\sup_{(\mathbf{S}_i)_{i=1}^N \in \mathcal{P}} \lim_{t \rightarrow \infty} P_M(\delta_t) = 0$ , where

$$\mathcal{P} = \{(\mathbf{S}_i)_{i=1}^N : (\mathbf{S}_i)_{i=1}^N \text{ contains an information flow and CTR} \leq r \text{ a.s.}\}$$

The *consistency* of a detector  $\delta_t$  is the supremum of  $r$  such that  $\delta_t$  is  $r$ -consistent.

The consistency of a distributed detection scheme will depend on, among other factors, the rate constraint the scheme obeys and the number of monitored nodes. If there exists a scheme that achieves the per-node capacity constraint  $R$  and has consistency  $r$ , we say that the pair  $(R, r)$  is achievable. If we find the supremum of achievable consistencies for each per-node rate constraint, we have a function that partitions the consistency-rate space into achievable and unachievable regions. This is called the *consistency-rate function*, as defined below:

*Definition 2.3:* Given a per-node capacity constraint  $R$ , the *consistency-rate function* is defined as

$$\begin{aligned} \alpha(R) &\equiv \sup\{r \in [0, 1] : \exists h_i^{(t)}(\cdot), i = 1, \dots, N, \delta_t(\cdot) \text{ s.t.} \\ &\quad \delta_t \text{ is } r\text{-consistent and} \\ &\quad \limsup_{t \rightarrow \infty} \frac{1}{t} I(\mathbf{S}_i^{(t)}; \mathbf{Q}_i^{(t)}) \leq R, i = 1, \dots, N\} \quad (3) \end{aligned}$$

The ultimate goal of this work is to find the consistency-rate function and design quantizers and detectors to achieve it. For now, though, we use some simplifying assumptions. In general, the quantizer functions may be randomized, which is why the rate is computed as  $\frac{1}{t} I(\mathbf{S}_i^{(t)}; \mathbf{Q}_i^{(t)})$ . To simplify the analysis, though, we will consider only deterministic quantizers. Under this assumption, the rate is computed as  $\frac{1}{t} H(\mathbf{Q}_i^{(t)})$ .

### III. A QUANTIZER

To further simplify the analysis, we focus our attention on a particular quantization scheme. If we find the consistency achievable by a detector using this scheme, we will have a lower bound on the actual rate-consistency function. The particular scheme we will use is the *slotted counter* scheme used in [9] as part of a distributed detection scheme for two-hop information flows and originally used in [10] to compress Poisson processes. Under this scheme, a slot length  $T$  is chosen, and the quantizer's output is a sequence representing the number of packets in each slot.

*Definition 3.1:* Given a point process  $\mathbf{S}$ , a slotted counter with slot length  $T$  is defined as  $\gamma(\mathbf{S}) \equiv (Z_1, Z_2, \dots)$ , where  $Z_j = |\mathcal{S} \cap [(j-1)T, jT)|$ .

If the source is a Poisson process of rate  $\lambda$ , then  $\{Z_n\}$  is an i.i.d. sequence of Poisson random variables with mean  $\lambda T$ . This sequence can be transmitted at rate

$$R(T) = \frac{1}{T} H_p(\lambda T) \quad (4)$$

where  $H_p(\cdot)$  is the entropy of a Poisson random variable with a given mean.

### IV. A CHAFF ASSIGNMENT ALGORITHM

The detection schemes in [8] and [9] used algorithms that estimated the CTR of a sequence of processes, then compared the estimated CTR to a threshold to choose a hypothesis. One of these algorithms is Bounded Greedy Match, defined in [11], which can be used when there are two monitored nodes. He and Tong generalized it to an arbitrary number of nodes, as the MBDR algorithm. In [9], a point process is constructed from slot-quantized realizations, and BGM is applied to these processes. These schemes have been shown to find the minimum possible CTR of a given sequence of processes.

We could apply the scheme from [9] to construct point processes from the output of the slot quantizers, then apply MBDR to find the minimum possible CTR. Instead, we develop a faster algorithm that applies directly to the slot-quantized realizations. We call this new algorithm Quantized Bounded Delay Relay (QBDR.) The algorithm is presented in Table I and is demonstrated on a particular realization in Table II.

QBDR begins by initializing slot pointers to point to the first slot in each process. At each step, we find the most possible flows along the current pointer path and subtract this value from each slot in the path (lines 9-12). Next, we update the pointers to find the next possible slot. First, we increment every pointer to a slot with zero packets (lines 14-16) until none meet the criterion, since no flow can pass through a slot with zero packets. Next, we increment every pointer that is before its predecessor, as these pointers violate the causality constraint (lines 18-20). Finally, we increment every pointer that is not within the delay bound of its successor, as these pointers violate the delay constraint (lines 22-24). We continue until one of the pointers is incremented past the end of the realizations. At this point no more flows can be found. Any remaining packets must be chaff.

Like MBDR and BGM, it can be shown that QBDR finds the minimum possible CTR for a given realization, as stated in the following proposition.

*Proposition 4.1:* For any realization  $(\mathbf{q}_1^n, \dots, \mathbf{q}_N^n)$ , QBDR finds the minimum CTR of an information flow with bounded delay  $\Delta$ .

The algorithm has worst case complexity  $O(N^2n)$ .

### V. A DETECTOR

The output of algorithm QBDR is  $\text{CTR}(t)$ . We use this value as a statistic for our detector:

$$\delta_t(\mathbf{q}_1^n, \dots, \mathbf{q}_N^n; \tau_n) = \begin{cases} 1 & \text{if } \widehat{\text{CTR}}(t) \leq \tau_n \\ 0 & \text{otherwise.} \end{cases} \quad (5)$$

Ideally, we should ignore packets that are early or late enough to be part of information flows with packets before time 0 or time  $t$ , but in the asymptotic case we consider, this is negligible.

It can be shown that as  $t \rightarrow \infty$ , the estimated CTR returned by QBDR converges to a constant, as stated in the following proposition.

```

Quantized-Bounded-Delay-Relay( $\mathbf{q}_1^n, \dots, \mathbf{q}_N^n, \Delta, T$ )
1:  $P \leftarrow \sum_{k=1}^N \sum_{i=1}^n q_k(i)$  // Count total number of packets
2:
3:  $D \leftarrow \lceil \frac{\Delta}{T} \rceil$ 
4: for  $k = 1 : N$  do
5:    $x_k \leftarrow 1$  // Initialize bin pointers
6: end for
7:
8: while each  $x_i < n$  do // Iteratively eliminate flows
9:    $m \leftarrow \min_k q_k(x_k)$ 
10:  for  $k = 1 : N$  do
11:     $q_k(x_k) \leftarrow q_k(x_k) - m$ 
12:  end for
13:
14:  while  $\exists k$  such that  $q_k(x_k) = 0$  do
15:    increment all  $x_k$  such that  $q_k(x_k) = 0$ 
16:  end while
17:
18:  while  $\exists k$  such that  $x_k < x_{k-1}$  do
19:    increment all  $x_k$  such that  $x_k < x_{k-1}$ 
20:  end while
21:
22:  while  $\exists k$  such that  $x_k < x_{k+1} - D$  do
23:    increment all  $x_k$  such that  $x_k < x_{k+1} - D$ 
24:  end while
25: end while
26:  $C \leftarrow \sum_{k=1}^N \sum_{i=1}^n q_k(i)$  // Count remaining packets
27:  $\widehat{\text{CTR}} \leftarrow \frac{C}{P}$ 

```

TABLE I  
THE QBDR ALGORITHM

*Proposition 5.1:* The minimum CTR of a process that produces the quantized realization  $\mathbf{q}_i^n, i = 1, \dots, N$  under  $\mathcal{H}_0$  with Poisson sources converges a.s. to a constant.

This fact allows us to find the consistency of the detector given in Equation 5. Suppose that under  $\mathcal{H}_0$ , QBDR converges to a constant  $c$  a.s. Now, choose any  $\tau_n$  less than  $c$ . This detector will achieve vanishing false alarm probability. In addition, the miss detection rate will vanish for any flow with CTR less than  $\tau_n$ . Therefore, the consistency of the detector will be  $c$ .

This also gives us a procedure for finding a lower bound on the rate consistency function that we will call  $\alpha^*(R)$ .

- 1) Invert Equation 4 to find  $T(R)$ .
- 2) Generate epochs of  $N$  independent Poisson processes with parameter  $\lambda$  over a period of time  $t$ , where  $t$  is long so the estimated CTR is near the convergent value.
- 3) For each  $R$ , pass each process through a slot quantizer with slot length  $T(R)$ .
- 4) Use QBDR with delay bound  $\Delta$  to find the minimum possible CTR to generate the observations.
- 5)  $\alpha^*(R) = \widehat{\text{CTR}}$

In [8], a lower bound on the asymptotic CTR estimated by the MBDR algorithm under  $\mathcal{H}_0$  was found. We can leverage this lower bound to find one for the asymptotic CTR estimated by the QBDR algorithm:

*Theorem 5.2:* If  $\mathbf{S}_i, i = 1, \dots, N$  are independent Poisson processes of maximum rate  $\lambda$ , and  $\mathbf{Q}_i, i = 1, \dots, N$  are the

QBDR operates on  $N$  sequences of random variables, representing the output of  $N$  slot quantizers. The following example demonstrates the operation of the algorithm.

**Initialization** (lines 4-6):

All slot pointers (represented by underlines) are initialized to slot 1. For this example, suppose  $D = 1$ .

|          |          |   |   |   |
|----------|----------|---|---|---|
| <u>2</u> | <u>2</u> | 5 | 3 | 0 |
| <u>4</u> | 1        | 0 | 5 | 4 |
| <u>2</u> | 3        | 4 | 1 | 0 |

**Eliminate flows** (lines 9-12):

The minimum value pointed to is 2. Subtract 2 from each value.

|          |          |   |   |   |
|----------|----------|---|---|---|
| <u>0</u> | <u>2</u> | 5 | 3 | 0 |
| <u>2</u> | 1        | 0 | 5 | 4 |
| <u>0</u> | 3        | 4 | 1 | 0 |

**Find non-empty slots** (lines 14-16):

Increment every pointer whose slot contains 0.

|          |          |   |   |   |
|----------|----------|---|---|---|
| 0        | <u>2</u> | 5 | 3 | 0 |
| <u>2</u> | 1        | 0 | 5 | 4 |
| 0        | <u>3</u> | 4 | 1 | 0 |

**Enforce causality** (lines 18-20):

Increment every pointer that is earlier than its predecessor's pointer.

|   |          |   |   |   |
|---|----------|---|---|---|
| 0 | <u>2</u> | 5 | 3 | 0 |
| 2 | <u>1</u> | 0 | 5 | 4 |
| 0 | <u>3</u> | 4 | 1 | 0 |

**Enforce delay** (lines 22-24):

No action necessary.

|   |          |   |   |   |
|---|----------|---|---|---|
| 0 | <u>2</u> | 5 | 3 | 0 |
| 2 | <u>1</u> | 0 | 5 | 4 |
| 0 | <u>3</u> | 4 | 1 | 0 |

**Eliminate flows** (lines 9-12):

The minimum value pointed to is 1. Subtract 1 from each value.

|   |          |   |   |   |
|---|----------|---|---|---|
| 0 | <u>1</u> | 5 | 3 | 0 |
| 2 | <u>0</u> | 0 | 5 | 4 |
| 0 | <u>2</u> | 4 | 1 | 0 |

**Find non-empty slots** (lines 14-16):

Pointer 2 is incremented twice to get to the first nonzero value.

|   |          |   |          |   |
|---|----------|---|----------|---|
| 0 | <u>1</u> | 5 | 3        | 0 |
| 2 | 0        | 0 | <u>5</u> | 4 |
| 0 | <u>2</u> | 4 | 1        | 0 |

**Enforce causality** (lines 18-20):

Pointer 3 is incremented until it no longer occurs before pointer 2.

|   |          |   |          |   |
|---|----------|---|----------|---|
| 0 | <u>1</u> | 5 | 3        | 0 |
| 2 | 0        | 0 | <u>5</u> | 4 |
| 0 | 2        | 4 | <u>1</u> | 0 |

**Enforce delay** (lines 22-24):

Pointer 1 is incremented until it is within  $D$  of pointer 2.

|   |   |          |          |   |
|---|---|----------|----------|---|
| 0 | 1 | <u>5</u> | 3        | 0 |
| 2 | 0 | 0        | <u>5</u> | 4 |
| 0 | 2 | 4        | <u>1</u> | 0 |

This process continues until all flows are found. All leftover packets are deemed chaff.

TABLE II  
QBDR DEMONSTRATION

slot-quantized versions of the  $\mathbf{S}_i$  with slot length  $T$ , then

$$\lim_{n \rightarrow \infty} \text{CTR}_{\text{QBDR}}(n) \geq 1 - \kappa_N \text{ a.s.} \quad (6)$$

where

$$\kappa_N = \min \left( (\lambda \Delta')^{N-2} (1 - e^{-\lambda \Delta'}), \prod_{i=1}^{n-1} (1 - e^{-i \lambda \Delta'}) \right),$$

$$\Delta' = T \left( \lceil \frac{\Delta}{T} \rceil + 1 \right).$$

If  $\lambda \Delta' < 1$ , then the CTR of QBDR goes to one exponentially as  $N$  increases. As  $T \rightarrow 0$ ,  $\Delta' \rightarrow \Delta$ , giving us the same lower bound as in the unquantized case, which is to be expected.

Suppose we have a way to embed chaff in an information flow in such a way that the marginal distributions of the unquantized processes and the joint distribution of the quantizer outputs are the same as under independent traffic. There could then be no detector that distinguish between  $\mathcal{H}_0$  and  $\mathcal{H}_1$ . The

CTR of the information flow generated in this scheme is then an upper bound on the consistency achievable by any detector using the quantizer.

For the slot quantizer, we have the following scheme:

- 1) Generate sequences  $X_k(j), k = 1, \dots, N; j = 1, \dots$  of i.i.d. Poisson random variables with mean  $\lambda T$ .
- 2) In each slot  $j$ , find  $X_{\min}(j) = \min_{k=1, \dots, N} X_k(j)$ .
- 3) Generate the point process  $\mathbf{F}_1$  by choosing  $X_{\min}(j)$  epochs uniformly over  $[(j-1)T, jT)$  for all  $j$ , and let  $\mathbf{F}_i = \mathbf{F}_1, i = 2, \dots, N$ .
- 4) Generate the point processes  $\mathbf{W}_k, k = 1, \dots, N$  by choosing  $(X_k(j) - X_{\min}(j))$  epochs uniformly over  $[(j-1)T, jT)$  for all  $k \in 1, \dots, N$  and  $j$ .
- 5) Generate the final point processes  $\mathbf{S}_k = \mathbf{F}_k \oplus \mathbf{W}_k$ .

Now each  $\mathbf{S}_k$  is a Poisson process with rate  $\lambda$ , and the quantized sequences are still  $\mathbf{X}_k, k = 1, \dots, N$ , which are i.i.d. Poisson random variables with mean  $\lambda T$ . The average number of flow packets per node will be  $E\left(\min_{k=1, \dots, N} Y_k\right)$ , where  $Y_k, k = 1, \dots, N$  are i.i.d. Poisson random variables. From this we can compute the CTR and state that under slot quantization,

$$\alpha(R) \leq 1 - \frac{E\left(\min_{k=1, \dots, N} Y_k\right)}{\lambda T}. \quad (7)$$

## VI. SIMULATIONS

Experimental traces were generated in MATLAB, and  $\alpha^*(R)$  was computed using the procedure defined above. For each  $R$ , the value of  $T(R)$  was computed, traces of length  $t = 10^4 \times T(R)$  were generated. Results appeared consistent across multiple trial simulations, giving us confidence that the convergent values were being reached.

Consistency-rate lower bounds were found by taking a fixed number of hops and sweeping over many values of  $R$ . The results are shown in Figure 3.

The relationship between number of nodes and consistency was found by fixing the rate and sweeping over many values of  $N$ . The results are shown in Figure 4.

Meanwhile, the upper bounds were computed by generating many length  $N$  sequences of Poisson random variables and finding the empirical average of the minimum values.

These plots yield several observations. For very low rates, the consistency does not grow much with the number of hops. In some cases where the condition for the lower bound is not met, the consistency still approaches one as  $N \rightarrow \infty$ , but in other such cases it appears to approach a smaller asymptote. The performance of our detection scheme deteriorates as the packet arrival rate increases, because the increase in  $\lambda$  causes an increase in information entropy and a higher density of packets leads to more false matches between packets in successive nodes. As  $\lambda$  increases, the achievable region shrinks. Figure 5 shows that except at large rates our detector may be unable to achieve vanishing error probabilities even if there is a distinction between the measurable distributions under  $\mathcal{H}_0$  and  $\mathcal{H}_1$ .

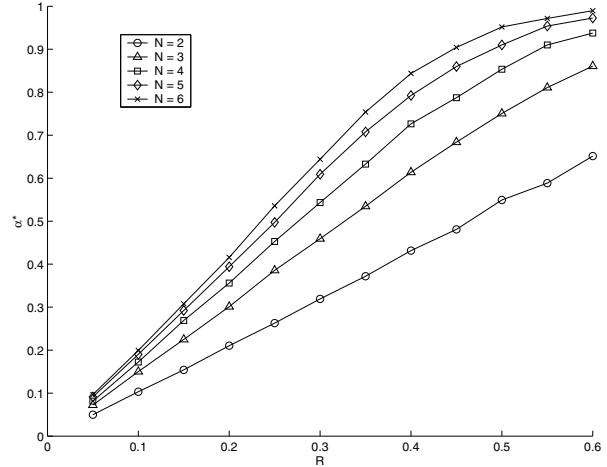


Fig. 3. An achievable consistency-rate function for detecting flows of various lengths using slot quantization. In this case,  $\lambda = 0.25$ ,  $\Delta = 1$ , and  $10^4$  slots are used for each experiment to ensure convergence.

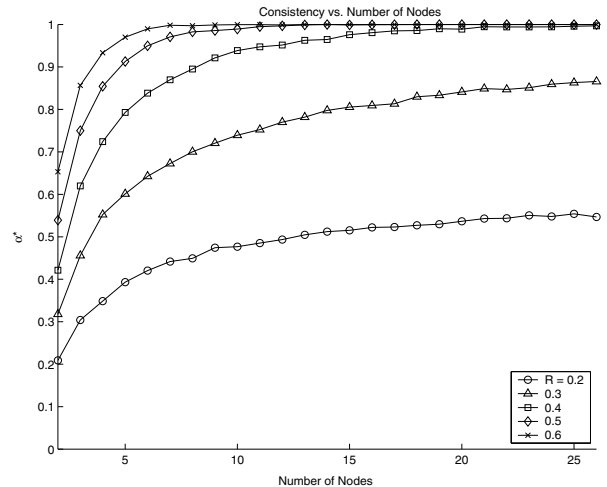


Fig. 4. The QBDR CTR  $\alpha^*(R)$  vs.  $N$ , the number of nodes, for various per-node rate constraints. In this case,  $\lambda = .25$ ,  $\Delta = 1$ , and  $10^4$  slots are used for each experiment to ensure convergence. Only in the  $R = 0.5$  and  $R = 0.6$  cases does the lower bound from Equation 6 go to 1 as  $N \rightarrow \infty$ .

## VII. CONCLUSIONS

In this paper, we developed a practical scheme for distributed detection of information flows. A new algorithm was developed to find the minimum fraction of chaff noise in a set of slot-quantized realizations. This algorithm gives similar results to a previous algorithm, but is more efficient and applies directly to the quantized realizations without reconstructing point processes. Under certain circumstances, the asymptotic performance of this scheme is almost perfect (detecting information flows dominated by chaff) when the number of monitored nodes is large.

## ACKNOWLEDGMENT

This work is supported in part by the U.S. Army Research Laboratory under the Collaborative Technology Alliance Pro-

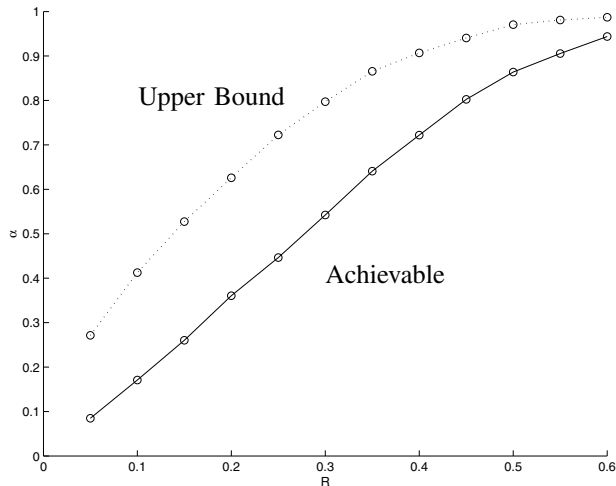


Fig. 5. An upper bound on the consistency-rate function of a scheme using slot quantization compared to the achievable consistency-rate function found by simulation. In this case,  $\lambda = 0.25$ ,  $\Delta = 1$ , and  $N = 4$ .

gram, Cooperative Agreement DAAD19-01-2-0011 and the National Science Foundation under Contract CCF-0635070. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation thereon.

#### REFERENCES

- [1] N. Ferguson and B. Schneier, *Practical Cryptography*. Indianapolis, IN: John Wiley & Sons, Inc., 2003.
- [2] S. Staniford-Chen and L. Heberlein, "Holding intruders accountable on the internet," in *Proc. the 1995 IEEE Symposium on Security and Privacy*, (Oakland, CA), pp. 39–49, May 1995.
- [3] X. Wang, D. Reeves, S. Wu, and J. Yuill, "Sleepy watermark tracing: An active network-based intrusion response framework," in *Proc. of the 16th International Information Security Conference*, pp. 369–384, 2001.
- [4] Y. Zhang and V. Paxson, "Detecting stepping stones," in *Proc. the 9th USENIX Security Symposium*, pp. 171–184, August 2000.
- [5] D. Donoho, A. Flesia, U. Shankar, V. Paxson, J. Coit, and S. Staniford, "Multiscale stepping-stone detection: Detecting pairs of jittered interactive streams by exploiting maximum tolerable delay," in *5th International Symposium on Recent Advances in Intrusion Detection, Lecture Notes in Computer Science 2516*, 2002.
- [6] T. He and L. Tong, "A Signal Processing Perspective to Stepping-stone Detection," in *Proc. 2006 Conference on Information Sciences and Systems*, (Princeton, NJ), March 2006.
- [7] T. He and L. Tong, "Detecting Information Flows: Improving Chaff Tolerance by Joint Detection," in *Proc. 2007 Conference on Information Sciences and Systems*, (Baltimore, MD), March 2007.
- [8] T. He and L. Tong, "Detection of Information Flows." submitted to *IEEE Trans. on Information Theory*, 2007.
- [9] T. He and L. Tong, "Distributed Detection of Information Flows." submitted to *IEEE Trans. on Information Forensics and Security*, 2007.
- [10] I. Rubin, "Information Rates and Data-Compression Schemes for Poisson Processes," *IEEE Transactions on Information Theory*, vol. 20, pp. 200–210, March 1974.
- [11] A. Blum, D. Song, and S. Venkataraman, "Detection of Interactive Stepping Stones: Algorithms and Confidence Bounds," in *Conference of Recent Advance in Intrusion Detection (RAID)*, (Sophia Antipolis, French Riviera, France), September 2004.