Distributed Detection of Multi-hop Information Flows with Fusion Capacity Constraints

Ameya Agaskar, Ting He, *Member, IEEE*, and Lang Tong[†], *Fellow, IEEE*

***Abstract*—The problem of detecting multi-hop information flows subject to communication constraints is considered. In a distributed detection scheme, eavesdroppers are deployed near nodes in a network, each able to measure the transmission timestamps of a single node. The eavesdroppers must then compress the information and transmit it to a fusion center, which then decides whether a sequence of monitored nodes are transmitting an information flow. A performance measure is defined based on the maximum fraction of chaff packets under which flows are still detectable. The performance of a detector becomes a function of the communication constraints and the number of nodes in the sequence. Achievability results are obtained by designing a practical distributed detection scheme, including a new flow finding algorithm that has vanishing error probabilities for a limited fraction of chaff packets. Converse results are obtained by characterizing the fraction of chaff packets sufficient for an information flow to mimic the distributions of independent traffic under the proposed compression scheme.**

***Index Terms*—Intrusion detection, Traffic analysis, Network surveillance, Information-theoretic limits**

I. Introduction

CONSIDER a wireless ad hoc network as illustrated in Figure 1. We want to analyze traffic in the network to detect the presence of information flows. If every packet is reencrypted and padded at each relay node, the only information we can work with is the timing of transmissions. The problem, then, is to infer the presence of flows from correlations among the transmission timing patterns of nodes. To do this, eavesdroppers may be deployed near several nodes in the network. No decision can be made by any eavesdropper since it can only observe the signal from a single node, but each eavesdropper may communicate with a fusion center under a rate constraint. The detector at the fusion center, given the limited data arriving from each eavesdropper, can then decide whether the observed path contains an information flow[1].

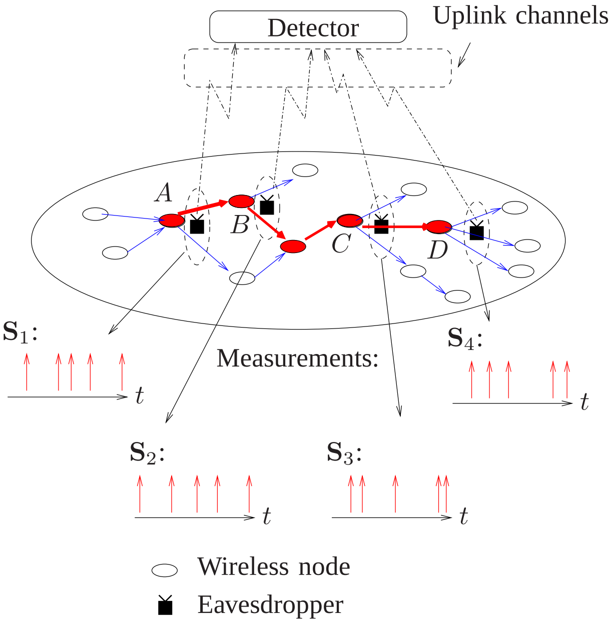


Fig. 1. In a wireless network, eavesdroppers are deployed to collect transmission activities (denoted $\mathbf{S}_i, i = 1, \ldots, N$) of N nodes. These processes are compressed and transmitted to a fusion center, which determines whether there is a flow from node A through nodes B, C, and D.

There are many challenges in such a distributed traffic analysis problem. The network can mask information flows through random timing perturbations, packet shuffling, or the insertion of extra packets that are not part of the flow. These extra packets, as well as packets not part of the flow in question, constitute *chaff noise*. The rate constraint for the communications channels poses another challenge; each eavesdropper must quantize its measurements in a manner that preserves the detectability of flows at the fusion center.

The problem considered here has applications in network surveillance and intrusion detection. With the proliferation of advanced encryption techniques, it can be useful to glean as much information about data flow in a network as possible, even if decryption of the data is out of reach. An eavesdropper may want to monitor a network using several low-cost sensors that communicate wirelessly with a fusion center; the limited power of the sensors motivate the rate constraints in this problem.

A. Related Work

Staniford and Heberlein [2] were the first to consider the problem of detecting stepping-stone attacks in networks. These are attacks wherein a malicious user routes data and commands

A. Agaskar was and L. Tong is with the School of Electrical and Computer Engineering, Cornell University, Ithaca, NY 14850 USA e-mail: {apa22, lt35}@cornell.edu.

T. He is with the IBM T.J. Watson Research Center, Hawthorne, NY 10532 e-mail: the@us.ibm.com

This work is supported in part by the National Science Foundation under Contract CCF-0635070 and by the Army Research Office MURI program under award W911NF-08-1-0238. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation thereon.

[†] Corresponding author.

[1] Part of this paper was presented in [1].

through several compromised nodes that do not belong to him before reaching the the final, targeted node. Solving this problem involved finding information flows. Staniford and Heberlein used traffic content to make a decision, as did Wang *et al.* [3]. Zhang and Paxson [4] first considered dealing with encrypted traffic using timing information instead of content. Donoho *et al.* [5] considered encrypted traffic with timing perturbations, and found that it was possible to exploit a known delay constraint to differentiate a flow from independent traffic. The authors of [6] showed that similar results could be obtained if an intruder were known to have a memory constraint. They later found the first timing-based detectors with vanishing error probabilities that work even when the chaff noise grows proportionally to the total traffic [7]. In [8], the authors showed that as the number of hops in an information flow grows large, an information flow buried in chaff noise can no longer be hidden from a detector. This paper is a followup of [9], where the problem of distributed detection of 2-hop information flows was first considered. Under a similar formulation as that in [9], we extend the results of [9] in several aspects. First, we consider general multi-hop (more than 2) information flows. Second, we propose a new algorithm. Unlike [9], our algorithm operates directly on the quantized processes, rather than first reducing the problem to an equivalent unquantized problem. The new algorithm is faster than the old except in cases where the rate constraint is loose. Third, we present experimental results of our algorithm applied to actual network traces.

B. Summary of Results and Organization

We consider the distributed detection of multi-hop information flows by timing analysis. As in [9], we characterize the performance of a detector by the largest fraction of chaff noise under which a flow is still detectable, and we break the problem down to quantization at local traffic sensors and detection at the fusion center. In [9], algorithms previously developed for the unquantized version of the problem were applied to preprocessed realizations of the quantized processes. In this paper, a new algorithm is developed that applies directly to the quantized realizations and provides some speedup when the rate constraint is not too loose. This algorithm finds the minimum possible fraction of chaff noise that could have been mixed with a delay-constrained information flow to generate the observed, quantized realizations. This value is then compared to a threshold, yielding detection if the threshold is not exceeded. Under the assumption that, in the absence of information flows, nodes transmit according to independent Poisson schedules, we provide a characterization of the threshold. We show that if the flow size is above a certain level, the miss detection and false alarm probabilities vanish as the number of packets used in the detection increases. We then provide an analytical upper bound on the fraction of chaff noise under which our detector provides a consistent detection. We compare the performance under various rate constraints and flow lengths. We then loosen the Poisson assumption and determine the performance of the distributed detection scheme when the interarrivals follow the Pareto distribution, which is said to more closely model packet transmission timestamps in networks [10]. Finally, we test our algorithm on real data, demonstrating the performance of the detection scheme on TCP traces.

The rest of the paper is organized as follows. Section II is the problem formulation. In Section III, we define a performance measure. In Section IV, we define the quantization scheme. In Section V, we define an algorithm for finding the minimum fraction of chaff noise. In Section VI, we show that the value computed by this algorithm converges almost surely (a.s.) when the sources are Poisson and we define a detector that takes advantage of this fact. We also derive an analytical upper bound on performance as well as a lower bound that is computed numerically. In Section VII, we show the results of these experiments and interpret them. In Section VIII, we consider the Pareto interarrival model and compare results to those under the Poisson model. In Section IX, we demonstrate the performance of the detection scheme on actual TCP traces. Finally, in Section X we conclude with remarks.

II. Problem Formulation

A. Notation

We use the following notation. Uppercase letters denote random processes; lowercase letters denote their realizations. Boldface letters represent vectors, and plain letters represent scalars. We use parentheses to indicate indexing. Script letters represent sets. Thus $\mathbf{S}$ is a point process, $\mathbf{s}$ its realization, $S(k)$ the kth epoch, $s(k)$ the realization of the kth epoch, and $\mathcal{S}$ the set of epochs in $\mathbf{s}$. As in [8] and [9], we define the *superposition operator* $\bigoplus$ to operate on the realizations of two point processes such that $(a_k)_{k=1}^{\infty} \bigoplus (b_k)_{k=1}^{\infty} = (c_k)_{k=1}^{\infty}$, where $c_1 \leq c_2 \leq \ldots$ and $\{a_k\}_{k=1}^{\infty} \bigcup \{b_k\}_{k=1}^{\infty} = \{c_k\}_{k=1}^{\infty}$.

B. Problem Statement

Let $\mathbf{F}_i, i = 1, \ldots, N$, be the point processes representing transmission activities at nodes 1 through N, respectively. We say that $\mathbf{F}_i, i = 1, \ldots, N$, is an information flow if it satisfies the following definition.

Definition 2.1: A sequence of processes $(\mathbf{F}_i)_{i=1}^{N}$ is an *information flow with delay constraint* Δ if for every realization $\mathbf{f}_i, i = 1, \ldots, N-1$, there exist bijections $g_i : \mathcal{F}_i \rightarrow \mathcal{F}_{i+1}$ such that $0 \leq g_i(s) - s \leq \Delta$ for every $s \in \mathcal{F}_i$.

The lower bound on $g_i(s) - s$ is the *causality constraint*, because it forbids a relay packet from occuring before its source packet. The upper bound is the *delay constraint*, as it sets a maximum delay between a source packet and its relay packet.

In reality, a node may have transmissions that are part of an information flow as well as other transmissions that are not part of a flow. We call these extra transmissions *chaff noise*. Then we say that a sequence of processes $(\mathbf{S}_1, \ldots, \mathbf{S}_N)$ *contains* an information flow if each process $\mathbf{S}_i$ can be decomposed into an information-carrying process $\mathbf{F}_i$ satisfying Definition 2.1 and a chaff process $\mathbf{W}_i$ so that [8]

$$\mathbf{S}_i = \mathbf{F}_i \bigoplus \mathbf{W}_i, i = 1, \ldots, N \tag{1}$$

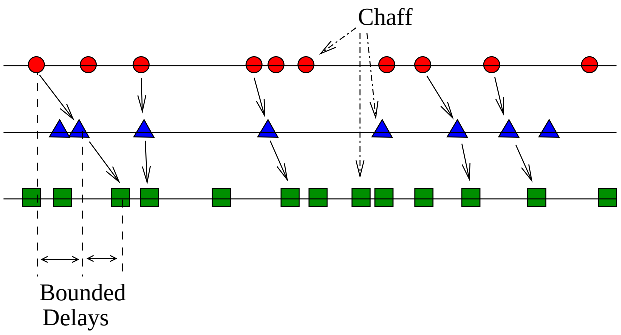


Fig. 2. An example of information flow with bounded delay superposed with chaff noise for $N = 3$ nodes. Each packet in a flow is matched to another packet in the next node with a delay less than Δ.

A realization containing an information flow is illustrated in Figure 2.

We pose the problem of flow detection as a test of the following hypotheses:

$$\begin{aligned}\mathcal{H}_0 :&\quad \mathbf{S}_1, \ldots, \mathbf{S}_N \text{ are jointly independent},\\ \mathcal{H}_1 :&\quad (\mathbf{S}_1, \ldots, \mathbf{S}_N) \text{ contains an information flow},\end{aligned}$$

Note that, although (1) appears to be the conventional ”signal plus noise” model, there is a key difference in that we allow the chaff noise to be arbitrarily correlated with the flow. Note also that our modeling hypotheses are not complementary — it is possible that along a sequence of N monitored nodes, there is a subset of $N-1$ nodes on which there is an information flow. Our detector, then, could be part of a scheme that sequentially searches for information flows on successively larger subsets of the monitored nodes. We consider lossless information flows only; if a packet is lost before reaching the destination, it will be considered chaff (though a detector of shorter information flows may detect it as such.)

In the distributed detection scheme, as illustrated in Figure 3, each eavesdropper can perfectly measure the realization of the process at a single node. It can then communicate directly with a fusion center independent of the other eavesdroppers, so long as it obeys a capacity constraint R. To achieve this, each eavesdropper applies a quantization function $h_i^{(t)}(\cdot)$ to the realization over $[0, t)$, obtaining $\mathbf{Q}_i^{(t)} = h_i^{(t)}(\mathbf{S}_i), i = 1, \ldots, N$. It then encodes the quantized process by choosing a codeword Ξ_i from a codebook of size e^{tR}, and transmits the codeword to the fusion center. The fusion center decodes the realizations to obtain estimated quantized realizations, $\hat{\mathbf{Q}}_i^{(t)}$. This can be done without error for sufficiently large t so long as $\frac{1}{t}H(\mathbf{Q}_i^{(t)}) \leq R$, where $H(\cdot)$ is the joint entropy of a collection of random variables. We assume that the sensors are synchronized to some reference timing signal; without such synchronization, it would be impossible to determine whether a hypothesized flow satisfies a delay bound.

Given these quantized realizations, the detector $\delta_t((\mathbf{q}_i^{(t)})_{i=1}^N)$ chooses a hypothesis. This is a partially non-parametric hypothesis test, as the correlations among the processes under $\mathcal{H}_1$ is not specified. We require that the marginal distribution of each node's process be the same under each hypothesis; otherwise, each eavesdropper could independently test the hypothesis and send its one bit result to the fusion center [9].

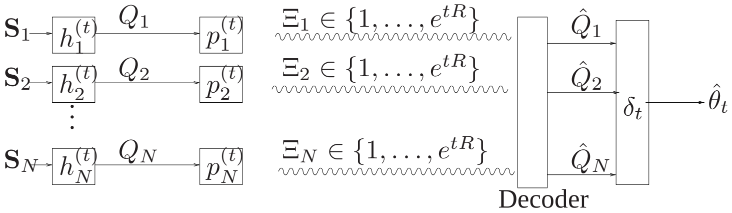


Fig. 3. A distributed detection system. The system consists of N quantizers $h_1^{(t)}, \ldots, h_N^{(t)}$ and a detector δ_t.

Given a per-node rate constraint R, the problem is to design the quantizer functions $h_i^{(t)}(\cdot)$ and the detector $\delta_t(\cdot)$ to optimize the overall detection performance.

III. Performance Measure

A measure of performance of a detector must take into account the detector's tolerance of chaff noise. To this end, we need to quantify the chaff noise contained in a process. We use the concept of the *chaff-to-traffic ratio* (CTR) defined in [8] and [9].

Definition 3.1: Given realization of an information flow $(\mathbf{f}_i)_{i=1}^N$ and chaff noise $(\mathbf{w}_i)_{i=1}^N$, the CTR is defined as

$$\begin{aligned}\text{CTR}(t) &\equiv \frac{\sum_{i=1}^{N} |\mathcal{W}_i \bigcap [0,t]|}{\sum_{i=1}^{N} |(\mathcal{W}_i \bigcup \mathcal{F}_i) \bigcap [0,t]|},\\ \text{CTR} &\equiv \limsup_{t\to\infty} \text{CTR}(t), \qquad (2)\end{aligned}$$

where the operator $|\cdot|$ gives the number of elements in its argument. So CTR(t) is the fraction of packets that are chaff before time t, and the CTR is the asymptotic fraction of chaff packets. The CTR is a function of the realizations, so it is itself a random variable.

To measure the performance of a detector, we use the following metric defined in [8], which is derived from the notion of Chernoff-consistency [11].

Definition 3.2: A sequence of detectors $(\delta_t)_{t>0}$ 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) $\lim_{t\to\infty} P_F(\delta_t) = 0$ for any $(\mathbf{S}_i)_{i=1}^N$ under $\mathcal{H}_0$;
2) $\sup_{(\mathbf{S}_i)_{i=1}^N \in \mathcal{P}} \lim_{t\to\infty} P_M(\delta_t) = 0$, 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 *measure of consistency* of a sequence of detectors $(\delta_t)_{t>0}$ is the supremum of r such that δ_t is r-consistent.

Given a per-node capacity constraint R and a distributed detection scheme with consistency r that obeys the constraint, we have an achievable pair (R, r). We can take the supremum of achievable consistencies as a function of the rate constraint to partition the consistency-rate space into achievable and unachievable pairs. This gives us the following definition.

Definition 3.3: Given a per-node capacity constraint R, the *consistency-rate function* (extended from the consistency-rate function in [9]) is defined as

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

In this paper we will consider only deterministic quantizers. By the source-channel separation theorem [12], the constraint on the entropy of the quantized process in Definition 3.3 is sufficient for lossless transmission over a channel with capacity R. So for CTR below $\alpha(R)$, there must be a detection scheme that satisfies the capacity constraint R and will detect all flows. In general, as R increases and the rate constraint is loosened, α should increase monotonically and approach the non-quantized case ($R = \infty$). The ultimate goal of this work is to find the consistency-rate function and design quantizers and detectors to achieve it.

IV. Data Compression Scheme

We now find a lower bound on the consistency-rate function for various numbers of nodes by defining an actual detection scheme. In particular, we employ a slotted quantizer as in [9] and originally used to compress Poisson point processes in [13].

Definition 4.1: Given a point process $\mathbf{S}$, a slotted counter with slot length T is defined as $\gamma(\mathbf{S}) \equiv (Z_1, Z_2, \ldots)$, where $Z_j = |\mathcal{S} \bigcap [(j-1)T, jT)|$ is the number of packets in slot j.

The slotted quantizer so defined gives us a sequence of random variables representing the number of epochs in successive bins of width T. If $\mathbf{S}$ is a Poisson process with intensity λ, then the Z_k are i.i.d. Poisson random variables with mean λT. For large block length, we can use a Lempel-Ziv encoder to transmit the Z_k with no errors at a rate [12]

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

where $H_p(\cdot)$ is the entropy of a Poisson random variable with a given mean. As we have already required that the sensors be synchronized to a timing reference, we also require that their quantization slots be synchronized. This aids the development of an algorithm to operate on the quantized vlaues only.

V. The Quantized Bounded Delay Relay Algorithm

Given the unquantized observations at the traffic monitors, algorithms developed in [8], [14] can be used to obtain an optimal partition (cf. [8, Section V-B]) of the observed point processes into flow with delay constraints and chaff processes. Specifically, the Multihop Bounded Delay Relay (MBDR) algorithm proposed in [8] gives the minimum $\text{CTR}(t)$ of a process generating a given realization with duration t. The resulting CTR is then used as the test statistic for flow detection.

We now consider the case when observations at traffic sensors must be quantized using, for example, slotted quantizers discussed in section IV. As an extension to the detection algorithm for unquantized observations, the key idea is to obtain an optimal partition algorithm that finds the minimum CTR that could have generated the quantized observations. The first such algorithm was presented in [9], where the problem was first reduced to an equivalent unquantized case by performing a worst-case reconstruction of the original point processes. We present here an alternative algorithm that operates directly on the slot-quantized values.

This new algorithm is referred to as Quantized Bounded Delay Relay (QBDR). It operates on the quantized sequences $\mathbf{Q}_i^n$, where $n = \lceil \frac{t}{T} \rceil$. Specifically, it decomposes the sequences into traffic $h(\mathbf{F}_i)$ and chaff $h(\mathbf{W}_i)$ sequences such that $\mathbf{S}_i = \mathbf{F}_i \bigoplus \mathbf{W}_i$, $h(\mathbf{S}_i) = \mathbf{Q}_i$, placing as few packets as possible into the chaff process while satisfying the delay and causality constraints. The algorithm requires the slot width of each quantizer to be equal, which is why we operate under equal rate constraints. In practice, one could analyze a system by assuming that each node has a rate constraint equal to the tightest one present.

The algorithm starts at the beginning of the sequences and progresses through candidate flow paths, eliminating as much traffic as possible. Whatever remains is classified as chaff. The candidate flow path is represented by a set of pointers, one per node, to a slot in that node's sequence. These pointers are initialized to the beginning of each sequence. Four steps are performed repeatedly until the end of any process is reached. First, any possible flows along the candidate path are eliminated by subtracting the minimum value of any slot in the path from all slots in the path. Second, any pointer to a slot with no packets remaining is incremented. Third, the causality constraint is enforced by incrementing any pointer to a slot earlier than its source node's pointer. Fourth, the delay constraint is enforced by incrementing any pointer that is not within $D = \lceil \frac{\Delta}{T} \rceil$ slots of its destination node's pointer. The algorithm is detailed in Table I with a specific example given in Table II.

Like its predecessor MBDR in the non-quantized case, QBDR is the optimal flow-finding algorithm in the quantized case, as stated in the following proposition.

Proposition 5.1: For any realization $(\mathbf{q}_1^n, \ldots, \mathbf{q}_N^n)$, QBDR finds the minimum CTR of an information flow with bounded delay Δ.

Proof: It was shown in [8] that an algorithm that greedily finds the earliest possible *relay sequences*, or sequences containing one epoch from each point process, will find the most possible relay sequences. The proof, which is omitted here, shows that there exists a largest set of order-preserving relay sequences (since packets can be shuffled without breaking the constraints to ensure that relay sequences don't cross). Next, if this largest set is not the one found through the greedy algorithm, then there exists a way to find a relay sequence not in the largest set. As this is a contradiction, the assertion is proved. Here, it remains to be shown that QBDR finds the earliest possible relay sequence at each step. We use an inductive argument. First note that the maximum delay bound in number of bins is $D = \lceil \frac{\Delta}{T} \rceil$, defined on line 3 of Table I. We define the "path" at each step as the

```
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 such $x_k$    // Search for traffic
16:     end while
17:
18:     while $\exists k$ such that $x_k < x_{k-1}$ do
19:         increment all such $x_k$    // Enforce causality
20:     end while
21:
22:     while $\exists k$ such that $x_k < x_{k+1} - D$ do
23:         increment all such $x_k$    // Enforce delay constraint
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

QBDR operates on N sequences of random variables, representing the output of N slot quantizers. The following example demonstrates the operation of the algorithm when $N = 3$. In the first step, each row represents the output of a node. Each column is a time slot. The values are stored and modified in the steps that follow.

Initialization (lines 4-6):
All slot pointers (represented by underlines) are initialized to slot 1. For this example, suppose $D = 1$.

$\underline{2}$	2	5	3	0
$\underline{4}$	1	0	5	4
$\underline{2}$	3	4	1	0

Eliminate flows (lines 9-12):
The minimum value pointed to is 2. Subtract 2 from each value.

$\underline{0}$	2	5	3	0
$\underline{2}$	1	0	5	4
$\underline{0}$	3	4	1	0

Find non-empty slots (lines 14-16):
Increment every pointer whose slot contains 0.

0	$\underline{2}$	5	3	0
$\underline{2}$	1	0	5	4
0	$\underline{3}$	4	1	0

Enforce causality (lines 18-20):
Increment every pointer that is earlier than its predecessor's pointer.

0	$\underline{2}$	5	3	0
2	$\underline{1}$	0	5	4
0	$\underline{3}$	4	1	0

Enforce delay (lines 22-24):
No action necessary.

0	$\underline{2}$	5	3	0
2	$\underline{1}$	0	5	4
0	$\underline{3}$	4	1	0

Eliminate flows (lines 9-12):
The minimum value pointed to is 1. Subtract 1 from each value.

0	$\underline{1}$	5	3	0
2	$\underline{0}$	0	5	4
0	$\underline{2}$	4	1	0

Find non-empty slots (lines 14-16):
Pointer 2 is incremented twice to get to the first nonzero value.

0	$\underline{1}$	5	3	0
2	0	0	$\underline{5}$	4
0	$\underline{2}$	4	1	0

Enforce causality (lines 18-20):
Pointer 3 is incremented until it no longer occurs before pointer 2.

0	$\underline{1}$	5	3	0
2	0	0	$\underline{5}$	4
0	2	4	$\underline{1}$	0

Enforce delay (lines 22-24):
Pointer 1 is incremented until it is within D of pointer 2.

0	1	$\underline{5}$	3	0
2	0	0	$\underline{5}$	4
0	2	4	$\underline{1}$	0

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

TABLE II
QBDR DEMONSTRATION

sequence of quantized values specified by the slot pointers x_k, or $(q_1(x_1), \dots, q_N(x_N))$. The slot pointers are represented by underlines in Table II. Each row in the demo represents a node in the flow, and has a single slot pointer.

1) On lines 4-6, we initialize our slot pointers to 1. Clearly the earliest possible relay sequence is along this path, computed on line 9. The maximum number of relay sequences is the minimum count along the path. We subtract this number from each slot in the path (lines 10-12) so that there are no more possible relay sequences.
2) At step l, suppose that the relay sequences have been removed from the path so that we must update it to find the earliest possible relay sequences. Clearly an active path cannot contain a slot with count 0, so we increase any pointer that points to a slot with count 0 (lines 14-16). Then we enforce the causality constraint by increasing each pointer that is earlier than its predecessor until all satisfy the constraint (lines 18-20). Note that each pointer can only be affected by its predecessors. Next we enforce the delay bound by increasing each pointer that is not within D of its successor until all satisfy the bound (lines 22-24). Note that each pointer can only be affected by its successors. This means that enforcing the delay bound can in no way undo the enforcement of the causality constraint. So we have the $(l+1)$th path that contains the earliest possible relay sequence.

The end condition is clear. When our path takes us outside the domain of any of the quantizer sequences, we can find no more flows. Since QBDR finds the earliest possible relay sequence at each step, it finds the most possible relay sequences, so it finds the minimum possible chaff. ■

QBDR's advantage over the previous method (constructing a worst-case point process and running MBDR) is that it is able to eliminate multiple flows in one step. On the other hand, if the average number of packets per slot is low, QBDR wastes time cycling through empty slots before finding an occupied one to search for a flow. Empirical results using MATLAB implementations show that QBDR performs better when more than one in four slots on average have at least one packet. This corresponds, for intensities of $\lambda = 1$, to a rate constraint less than 2-3 nats. For low rates, the improvement can be several orders of magnitude.

VI. A Threshold Detector and its Performance

Algorithm QBDR returns $\widehat{\mathrm{CTR}}(t)$, which we can use as a statistic for our detector, which we define as follows.

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

To determine the detector threshold τ_n we use the following proposition:

Proposition 6.1: If the $\widehat{\mathrm{CTR}}(t)$ obtained from QBDR converges almost surely to c under $\mathcal{H}_0$ as $t \to \infty$, then the measure of consistency (c.f. Definition 3.2) of the detector given in (5) is c.

Proof: Suppose that for large enough t, the estimated CTR from QBDR under $\mathcal{H}_0$ converges to some value c with probability one. Then we can fix any $\tau_n = \tau < c$ for the threshold detector and be sure that the false alarm probability will go to zero. Meanwhile, under $\mathcal{H}_1$, $\widehat{\mathrm{CTR}}(t) < \mathrm{CTR}$, where CTR is the true CTR of the process (this follows from Proposition 5.1.) So long as $\mathrm{CTR} < \tau$, $\widehat{\mathrm{CTR}} < \tau$, and we will have detection with probability 1. So as long as $\tau < c$, our detector is τ-consistent, so the measure of consistency for this detector is c. ■

We can show that under independent Poisson processes, the estimated CTR from QBDR does converge, as stated in the following proposition.

Proposition 6.2: The minimum CTR of a process that produces the quantized realizations $\mathbf{q}_i^n, i = 1,\ldots,N$ under $\mathcal{H}_0$ with Poisson marginals converges a.s. to a constant C.

Proof: First consider $T \ge \Delta$ for simplicity; the argument can be extended to the other cases. As QBDR's slot pointers move along the process, they modify the value in each slot by selectively removing packets corresponding to flows. Define $C_k^d(j)$ as the number of packets remaining in slot j of process k when the pointer to the first process x_1 first transitions from $j-d$ to $j-d+1$ (before the packet removal for that stage.)

Because of causality, $C_k^0(j) = C_k(j)$, the final number of chaff packets remaining in slot j of process k. Once the process 1 pointer has passed a particular slot, there can be no more flow packets in that slot for any process. Because of the delay constraint, a flow can only move forward $k-1$ slots from process 1 to process k. So $C_k^k(j) = Q_k(j)$.

Now consider the following collection of random variables, illustrated in Figure 4:

$$\mathbf{G}(j) \equiv \left(\left(C_k^{l-j+1}(l) \right)_{k=l-j+1}^{N} \right)_{l=j}^{j+N-1} \quad (6)$$

In other words, when the pointer x_1 jumps from $j-1$ to j, $\mathbf{G}(j)$ has the number of remaining packets in processes 1 through N in slot j, processes 2 through N in slot $j+1$, and processes k through N in slot $j+k-1$ for k up to N.

For the $N=2$ case, $\mathbf{G}(j)$ contains $C_1^1(j) = Q_1(j)$, $C_2^1(j)$, and $C_2^2(j+1) = Q_2(j+1)$. Given these variables

$$\begin{aligned} &C_2^1(j+1) = \\ &\max(\min(Q_2(j+1) - Q_1(j) + C_2^1(j), Q_2(j+1)), 0). \end{aligned}$$

In addition, $C_1^1(j+1) = Q_1(j+1)$, and $C_2^2(j+2) = Q_2(j+1)$, and we have a complete formula for $\mathbf{G}(j+1)$ given $\mathbf{G}(j)$.

$C_1^1(j)$	$\ldots$	$\ldots$	$\ldots$	$\ldots$
$C_2^1(j)$	$C_2^2(j+1)$	$\ldots$	$\ldots$	$\ldots$
$C_3^1(j)$	$C_3^2(j+1)$	$C_3^3(j+2)$	$\ldots$	$\ldots$
$C_4^1(j)$	$C_4^2(j+1)$	$C_4^3(j+2)$	$C_4^4(j+3)$	$\ldots$

Fig. 4. The Markov chain $\mathbf{G}(j)$ consists of the values that will be used by QBDR after the slot pointer x_1 jumps from $j-1$ to j. Shown here are the locations and random variables corresponding to the values in the Markov chain for $N=4$.

In general, $\mathbf{G}(j)$ contains every possible slot that a flow starting in slot j of process 1 can pass through. So obtaining $\mathbf{G}(j+1)$ involves running QBDR on $\mathbf{G}(j)$ until x_1 jumps from j to $j+1$, then copying over the top diagonal from $(Q_1(j+1), Q_2(j+2), \ldots)$. So $\mathbf{G}$ is a Markov process.

Process $\mathbf{G}$ is aperiodic and irreducible. The realization of $\mathbf{G}$ is bounded by the number of packets in each slot, which is a collection of independent Poisson random variables. The return time for the 0-state of this collection (where each element is zero) is finite, so the return time for the 0-state of $\mathbf{G}$ is finite. So $\mathbf{G}$ is positive recurrent. This property, combined with its aperiodicity and irreducibility, means that $\mathbf{G}$ is ergodic. So $\mathbf{G}$ converges to a limiting distribution. The amount of chaff in slot j, $C(j)$, is the number of packets remaining in slot j after QBDR is applied to $\mathbf{G}(j)$. Because $\mathbf{G}$ converges to a limiting distribution, $\frac{1}{n}\sum_{j=1}^{n} C(j)$ converges a.s. to a constant (that we call C) by ergodicity. The CTR, then, converges a.s. to $\frac{C}{N\lambda T}$. ■

Although we do not have an analytical expression for C, the proof of Proposition 6.1 suggests a numerical procedure: Given λ and R,

1) Invert (4) to find $T(R)$.
2) Generate epochs of N independent Poisson processes with parameter λ 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 Δ to find the minimum possible CTR to generate the observations.
5) $\alpha^*(R) = \widehat{\mathrm{CTR}}$

This gives us an achievable consistency value for each rate.

In [8], it was shown that under certain conditions, a lower bound on the estimated CTR of independent Poisson traffic converges to one as N increases. We can use this to obtain a similar result.

Theorem 6.3: If $\mathbf{S}_i, i = 1,\ldots,N$ are independent Poisson processes of maximum rate λ, and $\mathbf{Q}_i, i = 1,\ldots,N$ are the slot-quantized versions of the $\mathbf{S}_i$ with slot length T, then

$$\lim_{t\to\infty} \widehat{\mathrm{CTR}}_{\mathrm{QBDR}}(t) \ge 1 - \kappa \,\mathrm{a.\,s.} \quad (7)$$

where

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

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

Proof: First consider the meaning of Δ'. In QBDR, $D = \lceil \frac{\Delta}{T} \rceil$ is the maximum difference in slots for two packets to be matched. Each packet can be matched to another packet in one of $D+1$ different slots. For a packet at the very beginning of a slot, the maximum delay to the next packet is thus $T(D+1) = \Delta'$.

Since all the processes have the same arrival rate, and a flow takes one packet from each process, the estimated CTR tends to $1 - P_N$, where P_N is the probability that a packet in the $\mathbf{S}_1$ is matched to packets in $\mathbf{S}_2$ through $\mathbf{S}_N$, which each successive packet satisfying the delay bound. We therefore wish to upper bound P_N. Since the first packet in $\mathbf{S}_1$ claims packets in the remaining processes that become unavailable to future flows, P_N is upper bounded by the probability that the very first packet in $\mathbf{S}_1$ (call its arrival time t) is part of a flow. This is upper bounded by the probability that each $\mathbf{S}_i, i = 2, \ldots, N$ has at least one packet somewhere in $t + i\Delta'$ (a necessary but not sufficient condition for a flow.) So, since arrivals are Poisson, $P_N \leq \prod_{i=1}^{N-1} (1 - e^{-i\lambda\Delta'})$. Using inductive arguments as in [8], we can also show that $P_N \leq \lambda\Delta' P_{N-1}$ for $N \geq 3$. Substituting our expression for P_2, we therefore have also that $P_N \leq (\lambda\Delta')^{N-2}(1 - e^{-i\lambda\Delta'})$. So P_N is upper bounded by the minimum of the two expressions. Thus we have a lower bound on the CTR. ■

It is easy to see, then, that if $\lambda\Delta' < 1$, then the CTR of QBDR goes to one exponentially with N. It is also clear that as $T \to 0$, $\Delta' \to \Delta$, meaning that the perfect information case agrees with the lower bound computed in [8].

Given a quantizing scheme and a null hypothesis, it may be possible to construct an information flow with chaff embedded in such a way that it is statistically indistinguishable from independent traffic. It can be shown [8] in the non-distributed version of this problem, with a Poisson null hypothesis, that the fraction of chaff required to do this is no greater than the consistency of a CTR threshold detector. This is not the case here because the optimal quantizer is unknown, but we can use it to find an upper bound on the consistency of our detector:

Theorem 6.4: Under the slot quantization scheme, the consistency obeys

$$\alpha(R) \leq 1 - \frac{\mathrm{E}\left(\min\limits_{k=1,\ldots,N} X_k\right)}{\lambda T}, \tag{8}$$

where $X_k \overset{\text{i.i.d.}}{\sim} \text{Poisson}(\lambda T)$.

Proof: We use the fact that in a Poisson point process, conditioned on the number of epochs Q in a slot, the location of the epochs has the same distribution as Q i.i.d. uniform random variables over that slot. We consider the following construction:

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

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

$$\alpha(R) \leq 1 - \frac{\mathrm{E}\left(\min\limits_{k=1,\ldots,N} X_k\right)}{\lambda T}. \tag{9}$$

■

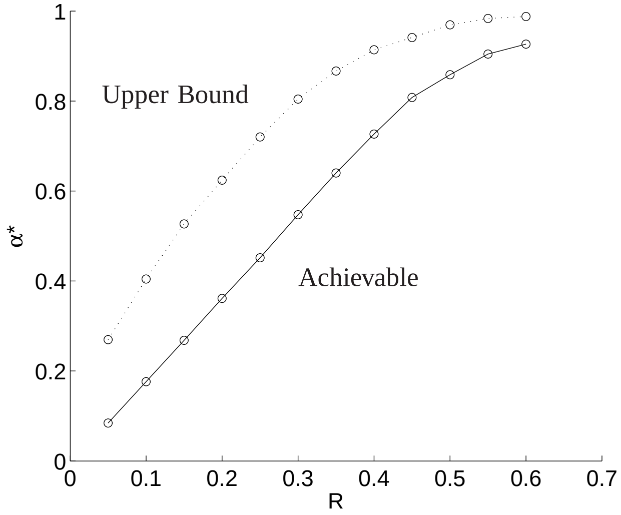


Fig. 7. An upper bound on the consistency-rate function of a scheme using slot quantization compared to the achievable consistency-rate function found by simulation. In this case, $\lambda = 0.25$, $\Delta = 1$, and $N = 4$.

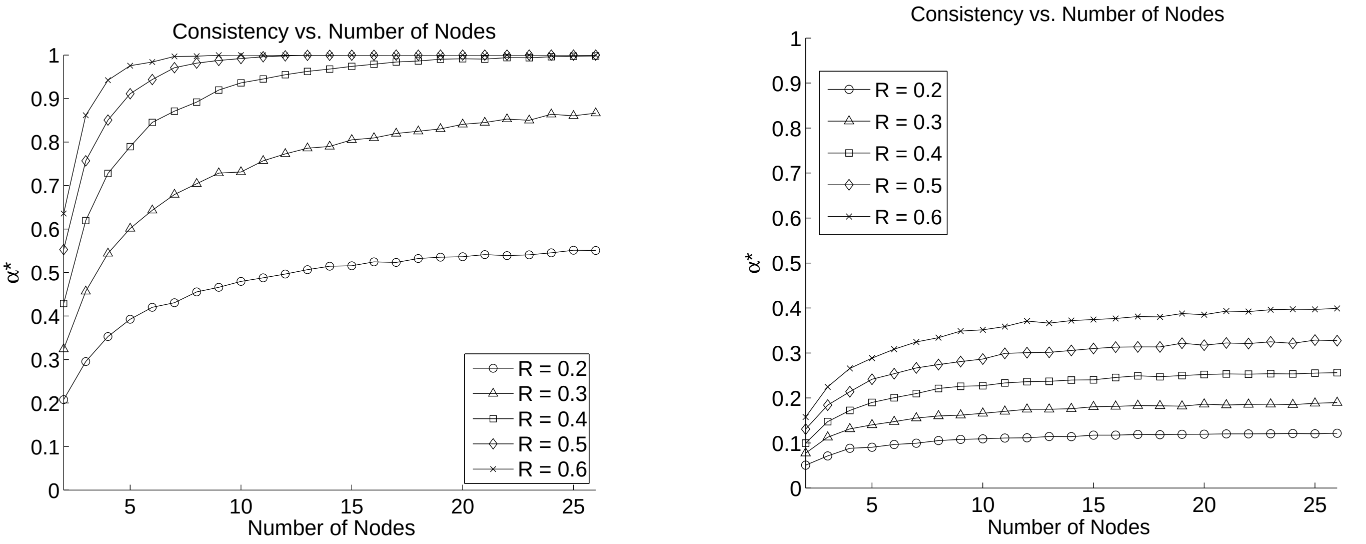
This upper bound holds for all detectors that use the slot quantization scheme. For large N, the upper bound is close to 1, and is thus not very instructive. For $N = 2$, it is equivalent to the upper bound derived in [9].

VII. Simulations

Experimental traces were generated in MATLAB, and the function $\alpha^*(R)$ was computed according to the procedure outlined above. We generated new processes for each R, and set $t = 10^4 \cdot T(R)$. This was long enough to produce consistent behavior between trials, which led us to believe that the convergent value was being found by the QBDR algorithm.

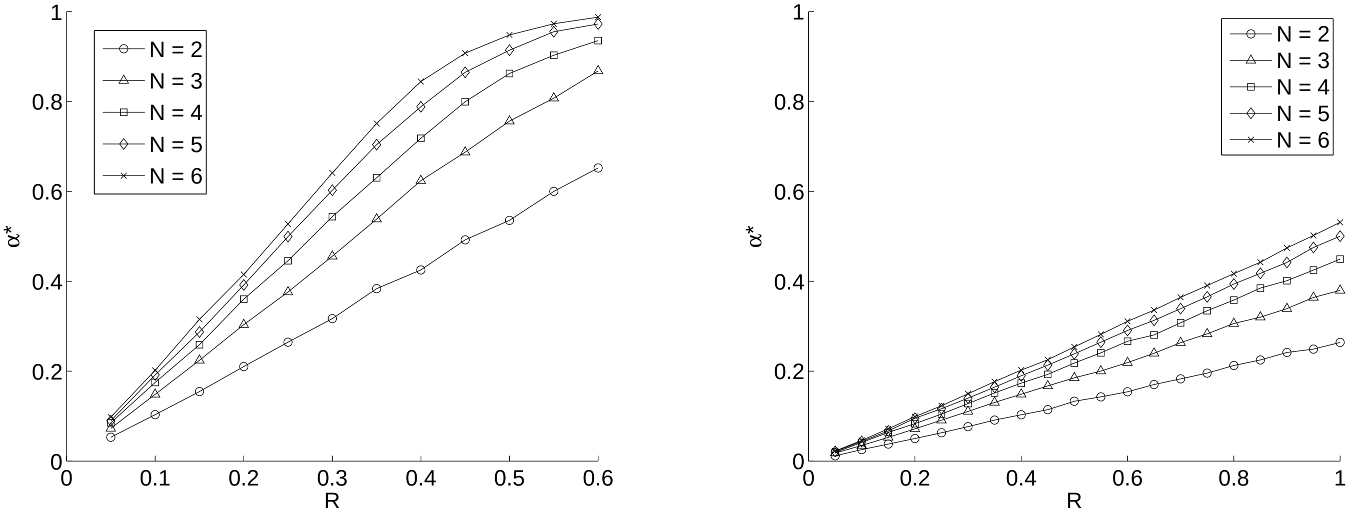
To assess the relationship between the number of nodes in the hypothesized flow and the consistency of the detection scheme, we let the number of nodes vary from 2 to 26, and for five rates plotted the consistency versus the number of nodes. The results are shown in Figure 5.

We also found the consistency-rate curves for 2 through 6 nodes, and plotted the curves on the domain $R \leq 1$. The results are shown in Figure 6.



(a) (b)

Fig. 5. The QBDR CTR $\alpha^*(R)$ vs. N, the number of nodes, for various per-node rate constraints. In this case, (a) $\lambda = .25$ and (b) $\lambda = 1$. $\Delta = 1$, and 10^4 slots are used for each experiment to ensure convergence. Slot lengths corresponding to rates are shown. Only in the $\lambda = .25$, $R = 0.5$ and $R = 0.6$ cases does the lower bound from (7) go to 1 as $N \to \infty$.



(a) (b)

Fig. 6. An achievable consistency-rate function for detecting flows of various lengths using slot quantization. In this case, (a) $\lambda = 0.25$ and (b) $\lambda = 1$. $\Delta = 1$, and 10^4 slots are used for each experiment to ensure convergence.

The upper bounds from (8) were computed by generating length N sequences of Poisson random variables and finding the average of the minimum value. The upper bound for a particular case is plotted and compared to the achieved consistency in Figure 7.

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 $N \to \infty$, but in other such cases it appears to approach a smaller asymptote. This indicates that the slotted quantization introduces a fundamental limitation that is not present in the perfect information case [8], in which the consistency tends to one as N grows for any arrival rate and delay bound. At high rates, the consistency should approach that of the unquantized case. At low rates, it is now clear, the consistency grows linearly with the rate. In this region, R is proportional to $\frac{\log T}{T}$. Intuitively, this can be understood as transmitting a random variable with entropy proportional to $\log T$ being transmitted every period of length T. Since T is large in this region, $\log T$ grows much more slowly than T. Changing the rate, then, is equivalent to changing the rate at which measurements are sent. So it is reasonable to expect that doubling this rate doubles the chaff we can tolerate, as the same chaff is now split over twice the measurements. The performance of our detection scheme deteriorates as the packet arrival rate increases, because the increase in λ causes an increase in information entropy and a higher density of packets leads to more false matches between packets in successive nodes. As λ increases, the achievable region shrinks. Figure 7 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 $\mathcal{H}_1$.

VIII. Other Interarrival Models

For tractability, we have thus far clung to the Poisson assumption—that interarrivals fit an exponential distribution. In actual networks, this assumption is not valid at the packet level. For example, TELNET packets have been shown [10] to be better modeled as i.i.d. Pareto distributed[2] with the shape parameter $\beta \approx 1$.

In [15], we showed that it is more difficult to hide an information flow in traffic under Pareto interarrival times with shape parameter within a certain range (which includes, in particular, the shape parameter used in [10]) than traffic with exponential interarrival times. An equivalent statement is that it is easier to detect an information flow under such conditions. The intuition is that such traffic is "burstier" than Poisson traffic, and so under the independent hypothesis fewer packets would be falsely labeled as flow packets.

Because the distribution of the slot quantized values under Pareto interarrival times is not known analytically, we resort to simulation to test the performance of our detection scheme under this hypothesis. In addition, we will not be able to accurately determine the entropy of the quantized process, and so will need an upper bound.

The distribution of the slot-quantized sequences will have mean λT, where λ is the average rate of the original process and T is the slot length, and will be over all nonnegative integers. It can be shown that of all discrete probability distributions with mean λT and nonnegative integer support, the one with the maximum entropy is the geometric distribution. The entropy rate under this distribution is $R = \frac{1}{T}\left((1+\lambda T)\log(1+\lambda T) - \lambda T \log(\lambda T)\right)$. We will use this formula to compute the desired slot length, knowing that the actual entropy rate of the sequence is smaller than the maximum allowed rate.

A. Consistency Experiment

This experiment was similar to the one described in Section VII. We simulated two independent renewal sources over a long period of time with interarrivals obeying Pareto distributions, performed slot quantization with slot length determined by the geometric entropy rate, and ran the QBDR algorithm to find the minimum fraction of chaff. Because the interarrival process is not memoryless, we cannot assume that the slot quantized sequences are i.i.d. or even Markov, so we cannot analytically show convergence of QBDR. Instead, we ran the experiment several times with the same parameters and found that the results do appear to converge to a limiting CTR. We then compared these results to the results under exponential interarrivals (the Poisson process.) The results are shown in Figure 8.

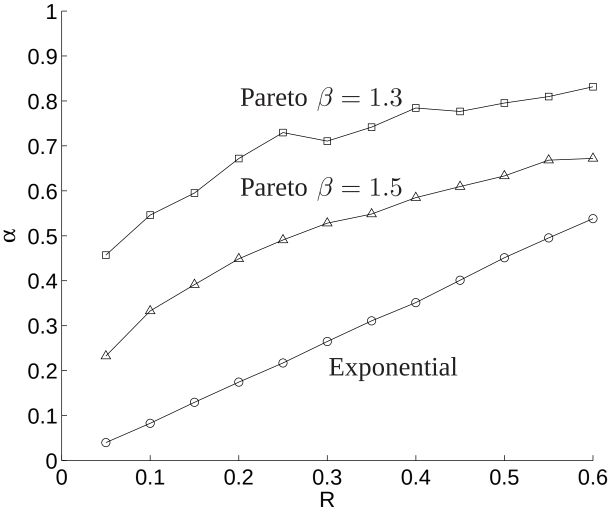


Fig. 8. Minimum CTR under independent traffic computed by QBDR after slot quantization under Pareto and exponential interarrival distributions. An upper bound on the rate is used for the Pareto distributions, so the estimated CTR is actually a lower bound for a particular rate. Pareto distributions have shape parameters $\beta = 1.5$ and $\beta = 2$. The results are closer to the Poisson case for larger shape parameters. In this experiment, $N = 4$, $\Delta = 1$, $\lambda = 0.5$ (where λ is the mean arrival rate), 10^4 slots are used to ensure convergence under Poisson interarrivals, and five trials with 2×10^5 slots are used in the Pareto interarrivals case.

From the figure, it is clear that at a given arrival rate, the Pareto interarrival distribution gives a higher CTR under $\mathcal{H}_0$ than the Poisson process, with the CTR getting smaller as the shape parameter β increases. This means that a detector designed under the Poisson assumption would also be Chernoff consistent if the arrival rate is the same but the interarrivals are Pareto distributed. This fits the intuition of a process with Pareto interarrivals being more "bursty" than those with Poisson interarrivals. Since, in the absence of an information flow, the processes for different nodes are independent, the arrivals of bursts are independent as well. The gaps between the bursts must be relatively long, since we are holding the mean arrival rate fixed. This makes it less likely that the packets in a burst can be matched up to packets in all the remaining processes. Thus, the estimated CTR is higher under the Pareto interarrivals than the Poisson.

B. ROC Experiment

In this experiment, we compared the receiver operating characteristics of the QBDR threshold detector under a Poisson process source with those under a Pareto interarrival source. For each threshold value between 0 and 1, the false alarm rate was found by generating N independent processes with mean arrival rate λ over 45 seconds (much smaller than the processes used in the consistency experiments,) quantizing them with a slot quantizer of slot length T (the same length

[2] We use for our definition of the Pareto cumulative distribution function $F_X(x) = 1 - (\frac{x}{x_m})^{\beta}$, where x_m is the *scale parameter* and β is the *shape parameter*.

was used for both kinds of processes,) and running the QBDR algorithm. If the estimated CTR was below the threshold, the trial was counted as a false alarm. 500 trials were used for each threshold value. To find the probability of detection of flows with a particular CTR γ, an information flow in chaff had to be generated. A process with mean arrival rate $(1-\gamma)\lambda$ was generated, and each packet was delayed by a uniform random value between 0 and $\Delta = 1$ to generate N processes. Then, each process was contaminated by superposing independent noise processes of rate $\gamma\lambda$ (so that the total rate was λ.) QBDR was run, and a trial was ruled a detection if the estimated CTR was below the threshold. Again, two hundred trials were used. The shape parameter for the Pareto distribution was 1.2. Figure 9 shows the results. In all tests, the ROC curves under the Pareto interarrival distribution were slightly better than those under the Poisson process distribution. Since, under $\mathcal{H}_0$, the estimated CTR is higher with Pareto interarrivals than Poisson, the false alarm rate is lower at a given threshold. The detection rate, on the other hand, will be lower under Pareto interarrivals because less of the chaff packets will be incorrectly considered as parts of a flow. However, it is clear from the graphs that the improvement in false alarm probability outweighs the loss in detection rate at a given threshold. It is also clear that the detector performance improves with decreasing CTR (which is to be expected) and with decreasing packet arrival rate.

IX. Performance on Network Traces

The final step from models to reality involved considering actual network data. We applied our quantization scheme followed by the QBDR algorithm on the LBL-PKT-4 trace generated by Paxson and first used in [10]. This trace is a record of packet timing and routing information for all wide-area traffic in and out of the Lawrence Berkeley Laboratory. It was also used in [8] for an experiment similar to ours. We extracted 41 traces that appeared upon inspection to be independent. For each trial, we randomly picked a pair of traces. We applied the slot quantizer to the pair and performed QBDR to find the estimated minimum CTR. Then we measured the average packet rate for the two traces and generated realizations of Poisson processes of those rates. We applied the slot quantizer to the Poisson pair and perform QBDR to find the estimated minimum CTR. The results are shown in Figure 10. Each point corresponds to a single trial; the Y-coordinate is the CTR of the traces, and the X-coordinate is the CTR of the corresponding Poisson process pair. In the experiment shown, 93% of the traces have a higher CTR than the Poisson processes of the same rate. This indicates that the false alarm probabilities will be lower for actual network traffic than our Poisson model, which gives us increased confidence in the performance of a system designed under the Poisson assumption.

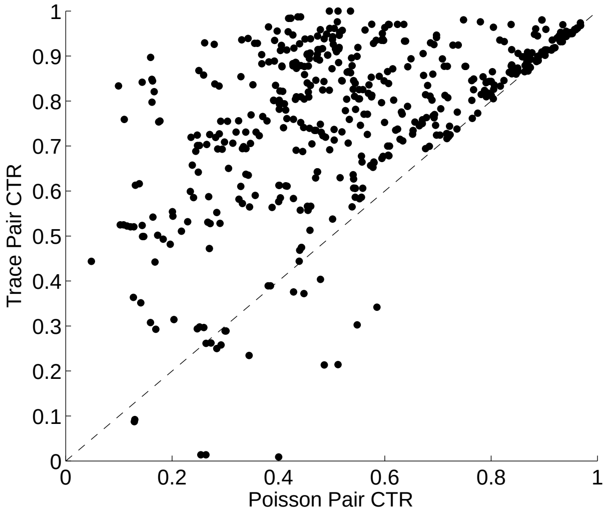


Fig. 10. CTR of actual traces vs. CTR of Poisson processes of the same rate, found using the QBDR algorithm after quantization. The rate constraint $R = 1$, and the delay bound $\Delta = 1$. Forty-one network traces are considered; for each trial two are selected at random to be processed. The plot shows the results for 500 trials. In 463 of them, the CTR for the traces is higher than the CTR for the Poisson processes.

X. Conclusions

In this paper we developed a practical distributed detection scheme for arbitrary-length information flows. As in the corresponding problem with perfect information, under certain conditions the performance of the scheme is near perfect for large N. Simulations show that if these conditions are not met, it is possible for flows embedded in chaff to remain undetected even for a large number of hops. Thus, given a rate constraint, traffic rate, and delay bound, it is possible to determine whether perfect detection of large information flows is guaranteed or unlikely. Preliminary results indicate that even if the detection scheme is designed under the assumption of exponential interarrivals, actual network traffic will perform at least as well as predicted under the faulty interarrival assumption.

References

[1] A. Agaskar, T. He, and L. Tong, “A Distributed Scheme for Detection of Information Flows in Chaff,” in *Proc. Conf. Info. Sci. and Signals 2008*.

[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,” to appear in IEEE Trans. on Information Theory.

[9] T. He and L. Tong, “Distributed Detection of Information Flows,” *IEEE Trans. on Information Forensics and Security*, vol. 3, pp. 390–403, September 2008.

[10] V. Paxson and S. Floyd, "Wide-Area Traffic: The Failure of Poisson Modeling," *IEEE/ACM Transactions on Networking*, vol. 3, pp. 226–244, June 1995.

[11] J. Shao, *Mathematical Statistics*. Springer, 1999.

[12] T. Cover and J. Thomas, *Elements of Information Theory*. John Wiley & Sons, Inc., 1991.

[13] I. Rubin, "Information Rates and Data-Compression Schemes for Poisson Processes," *IEEE Transactions on Information Theory*, vol. 20, pp. 200–210, March 1974.

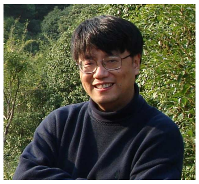
[14] A. Blum, D. Song, and S. Venkataraman, "Detection of Interactive Stepping Stones: Algorithms and Confidence Bounds," in *Conference of Recent Advance in Intrusion Detection (RAID)*, (Sophia Antipolis, French Riviera, France), September 2004.

[15] T. He, A. Agaskar, and L. Tong, "On Security-Aware Transmission Scheduling," in *Proc. ICASSP*, April 2008.

Ameya Agaskar received his B.S. in Engineering Physics in 2007 and his M.Eng. in Electrical and Computer Engineering in 2008 from Cornell University, where he was a Graduate Research Assistant in Prof. Lang Tong's Adaptive Communications & Signal Processing group (ACSP.) Since 2008, he has been with MIT Lincoln Laboratory's Advanced Sensor Techniques group, where he has studied signal processing techniques for radar and communication systems.

Ting He (S04-M07) joined IBM T. J. Watson Research Center in 2007, where she is now a Research Staff Member in the Wireless Networking Research Group. She received the Ph.D. degree in Electrical and Computer Engineering from Cornell University in 2007 and the B.S. degree in Computer Science from Peking University in 2003. At IBM, Ting is a primary researcher and task lead in the International Technology Alliance (ITA) program funded by US ARL and UK MoD, and a technical lead and primary researcher in several other research programs funded by ARL and NIST. Previously at Cornell (2003-2007), Ting was a Graduate Research Assistant in the Adaptive Communications & Signal Processing Group (ACSP) under the supervision of Prof. Lang Tong. Before joining Cornell, she worked as an Undergraduate Research Assistant in the Micro Processor Research & Development Center of Peking University from 2001 to 2003.

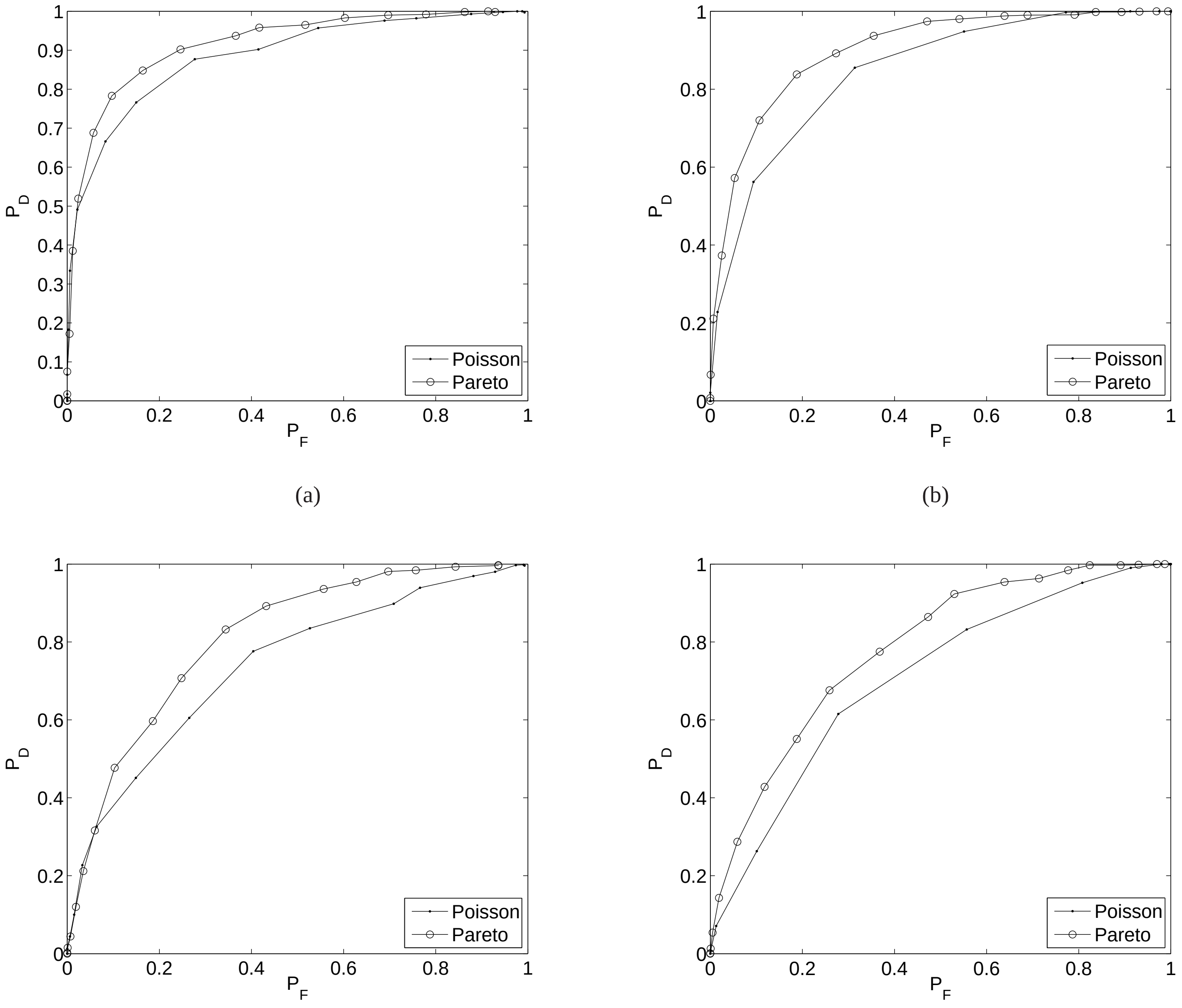
Ting is a member of IEEE. She received the Best Student Paper Award at the 2005 International Conference on Acoustic, Speech and Signal Processing (ICASSP). She was an Outstanding College Graduate of Beijing Area and an Outstanding Graduate of Peking University in 2003. She received the Excellent Student Award and the Canon, Sony, and Yang-Wang Academician scholarships for academic excellence from Peking University during 1999-2002.



Lang Tong (S'87,M'91,SM'01,F'05) is the Irwin and Joan Jacobs Professor in Engineering at Cornell University Ithaca, New York. He received the B.E. degree from Tsinghua University, Beijing, China, in 1985, and M.S. and Ph.D. degrees in electrical engineering in 1987 and 1991, respectively, from the University of Notre Dame, Notre Dame, Indiana. He was a Postdoctoral Research Affiliate at the Information Systems Laboratory, Stanford University in 1991. He was the 2001 Cor Wit Visiting Professor at the Delft University of Technology and had held visiting positions at Stanford University, and U.C. Berkeley.

Lang Tong is a Fellow of IEEE. He received the 1993 Outstanding Young Author Award from the IEEE Circuits and Systems Society, the 2004 best paper award (with Min Dong) from IEEE Signal Processing Society, and the 2004 Leonard G. Abraham Prize Paper Award from the IEEE Communications Society (with Parvathinathan Venkitasubramaniam and Srihari Adireddy). He is also a coauthor of six student paper awards. He received Young Investigator Award from the Office of Naval Research.

Lang Tong's research is in the general area of statistical signal processing, wireless communications and networking, and information theory. He has served as an Associate Editor for the IEEE Transactions on Signal Processing, the IEEE Transactions on Information Theory, and IEEE Signal Processing Letters.



(a)

(b)

(c)

(d)

Fig. 9. ROC curves for the QBDR threshold detector. In all cases, $\Delta = 1$, $T = 4$, the length of each source measurement was 45 s, $N = 4$, the Pareto shape parameter was 1.2, and 500 trials were used to find each P_F and P_D. In (a), CTR = 0.5 and $\lambda = 0.25$. In (b), CTR = 0.5 and $\lambda = 1.0$. In (c), CTR = 0.7 and $\lambda = 0.25$. In (d), CTR = 0.7 and $\lambda = 1.0$. ROC curves are shown for Poisson process and Pareto-interarrival renewal process sources.