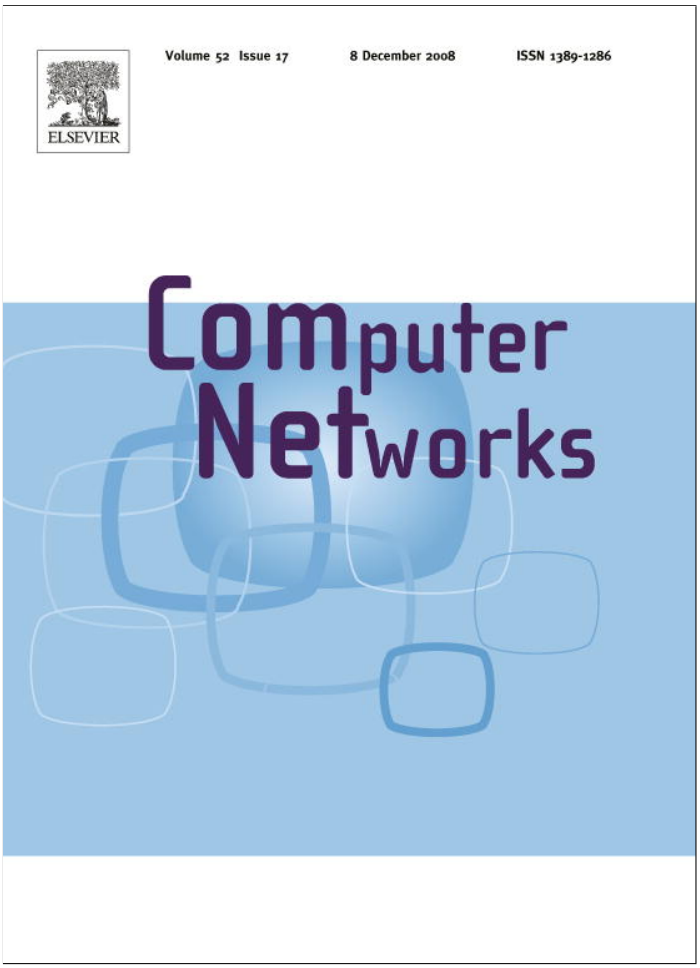


Computer Networks 52 (2008) 3205–3217

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

Classification of access network types: Ethernet, wireless LAN, ADSL, cable modem or dialup?

Wei Wei [a], Bing Wang [a,*], Chun Zhang [b], Jim Kurose [c], Don Towsley [c]

[a] Computer Science and Engineering Department, University of Connecticut, Storrs, CT 06269, United States
[b] IBM T.J. Watson Research Center, Hawthorne, NY 10532, United States
[c] Computer Science Department, University of Massachusetts, Amherst, MA 01003, United States

ARTICLE INFO

Article history:
Received 29 April 2008
Received in revised form 6 August 2008
Accepted 12 August 2008
Available online 2 September 2008

Responsible Editor: L.G. Xue

Keywords:
Access network classification
Packet pair
Network measurement

ABSTRACT

Ethernet, wireless LAN, ADSL, cable modem, and dialup are common access networks that have dramatically different characteristics. Fast and accurate classification of access networks can benefit a wide range of applications. In this paper, we propose a simple and efficient end-to-end scheme to classify access networks into three categories: Ethernet, wireless LAN and low-bandwidth connection. Our scheme is based on the intrinsic characteristics of the various access networks, and utilizes the median and entropy of packet-pair inter-arrival times. Extensive experiments show that our scheme obtains accurate classification results in a very short time (95% accuracy in 2 s, with 10 packet pairs).

1. Introduction

Access networks consist of links that connect end systems to edge routers. Ethernet, wireless LAN (WLAN), ADSL, cable modem, and dialup are commonly used access networks that have dramatically different characteristics. Fast and accurate classification of access networks is useful for a wide range of applications. It is useful for constructing peer-to-peer networks—after identifying the connection type, a peer can choose nodes with Ethernet connections to be neighbors. Similarly, it is useful for constructing application-level overlays and multicast trees [1,2]—after identifying the connection type, a super-node can choose those with Ethernet connections to be overlay nodes or place them at higher levels of a multicast tree. Connection type classification is also useful for applications with bandwidth and/or delay requirements. For instance, knowing a client's connection type helps a video-streaming server to adjust the bitrate of the video accordingly. Last, connection type classification helps to determine when to use performance-enhancing techniques for certain connection types, e.g., techniques for wireless links [3,4], and techniques for cable connections [5].

Accurate connection type classification is, however, not an easy task. It is often not possible for an end system to reliably report its connection type. This is mainly due to two reasons. First, the end system may not know its connection type. For instance, a laptop connected to a cable or ADSL modem through a wireless connection would mistakenly report WLAN (instead of cable or ADSL) as its connection type. Second, an end system may have incentives to conceal its connection type, and a compromised machine may also report its connection type inaccurately (e.g., to degrade the performance of an overlay network).

In this paper, we are interested in end-to-end approaches that require no network assistance for determining the type of an access network. We propose a simple and efficient scheme to classify access networks into three categories: Ethernet, WLAN, and low-bandwidth connection (cable, ADSL or dialup). Our algorithm uses packet pairs (a packet pair contains two back-to-back packets)

* Corresponding author. Tel.: +1 860 486 0582; fax: +1 860 486 4817.
E-mail addresses: weiwei@engr.uconn.edu (W. Wei), bing@engr.uconn.edu (B. Wang), czhang1@us.ibm.com (C. Zhang), kurose@cs.umass.edu (J. Kurose), towsley@cs.umass.edu (D. Towsley).

1389-1286/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.comnet.2008.08.018

and is based on the intrinsic characteristics of the various connection types. It works roughly as follows. If node *A* needs to determine the connection type of node *B*, *A* asks *B* to send it a sequence of packet pairs. Then node *A* determines *B*'s connection type based on the median and entropy of the inter-arrival times of the packet pairs. Extensive experiments show that our scheme obtains accurate classification results in a very short time (95% accuracy in 2 s, with 10 packet pairs).

Packet-pair or packet-train approaches have been used to determine the capacity or available bandwidth of an end-to-end path in wired networks [6,7]. Packet inter-arrival times are used in [8] to distinguish congestion losses from wireless losses under the assumption that the last-hop wireless link is the only bottleneck. Our work differs from previous studies in that our goal is to classify connections, and we make no assumption on the location of the bottleneck link. The study reported in [9] distinguishes wireless and wired connections. It assumes that wireless links have very low bandwidth and are lossy, and hence lead to a wider RTT spread than a wired connection. Our work, in contrast, is based on the intrinsic characteristics of the various access networks and therefore provides accurate classification regardless of the loss rate at the access network. Recent studies [10–12] differentiate two connection types, Ethernet and WLAN, based on packet traces collected passively at an aggregation point (e.g., the gateway router) of a local area network. Our study focuses on access network classification at a server and differentiates Ethernet, WLAN, and low-bandwidth access networks.

The rest of the paper is organized as follows. Section 2 provides some background on the access networks considered in this paper. Section 3 presents our classification scheme. Section 4 presents an analytical foundation for our approach. Section 5 describes the experimental results. Finally, Section 6 concludes the paper and describes future work.

2. Background

In this section, we provide background on all the access networks considered in this paper. Our focus is on those mechanisms within the access protocol that will allow us to distinguish one type of access network from another. We first describe IEEE 802.11 WLANs, and then describe cable networks. Last, we briefly describe Ethernet, ADSL and dialup connections.

2.1. IEEE 802.11 WLAN

We focus on two widely used types of WLANs: IEEE 802.11b and 802.11g. The MAC protocols of both 802.11b and 802.11g use CSMA/CA (carrier sense multiple access with collision avoidance) [13]. A wireless station accesses the channel using a basic access method or an optional four-way handshaking access method. The basic access method is used for small packets (smaller than a *RTS Threshold*). Since the packet pairs that we consider are small, we only describe the basic access method.

When using the basic access method, if a station has a packet to send, it is allowed to transmit when the media is free for a DIFS (distributed interframe space) amount of time. If the media is busy, a station sets a random backoff timer following a binary exponential backoff procedure. The initial value of the backoff timer is uniformly distributed in the range of 0 and CW (contention window). The contention window is set to CW_{min} for each new data transmission and doubles each time a transmission fails until it reaches the maximum contention window, CW_{max}. The backoff timer decreases by one when the media is idle for a slot time and is frozen when the channel is sensed busy. When the backoff timer reaches zero, the station sends the data frame.

In 802.11, unlike in Ethernet, the destination needs to send an explicit ACK to the sender since a wireless sender cannot determine whether or not its transmission has succeeded. Furthermore, to avoid channel capture, a wireless station must wait for a random backoff time after a successful transmission, even if no other station is transmitting. This implies that random backoff will occur between two back-to-back packets. Hence, when two back-to-back packets are sent on a perfect wireless channel, the inter-departure time of the packet pair follows a uniform distribution. We will take advantage of the distribution and median of the packet-pair inter-departure time in our classification scheme.

2.2. Cable network

In a cable network, the downstream channel from the Cable Modem Termination System (CTMS) at the headend to a Cable Modem (CM) at a residential home is a broadcast channel shared by many homes. The upstream channel from the CMs to the CTMS is a random access channel. We next briefly describe contention resolution in the upstream channel specified by DOCSIS (Data Over Cable Service Interface Specification), the *de facto* standard in the cable industry.

The upstream channel is divided into *mini-slots* of length 6.25 μs. The CTMS periodically broadcasts a downstream management message, referred to as *MAP*, to all the CMs. Each MAP contains timing information regarding *request mini-slots* and *data mini-slots*. When a CM has data to send, it must first send a request message using request mini-slots to the CMTS and wait for a *data grant*. After receiving a data grant message from the CMTS, a CM uses the assigned data mini-slots to transmit data on the upstream channel. The CMs contend for the use of request mini-slots following a contention resolution method based on binary exponential backoff [14]. The CTMS assigns the size of the initial backoff window and the maximum backoff window in the MAP. When a CM needs to send a request message, it randomly chooses a backoff value in its window. This backoff value indicates the number of contention mini-slots that the CM must wait before it can transmit the request. If a collision occurs (indicated by the absence of a data grant or a data pending indication in the next MAP from the CTMS to the CM), the CM doubles its window size, until the maximum window size is reached. A request message is discarded by the CM after 16 retries.

The contention on the upstream channel provides us an opportunity to distinguish cable modem from ADSL (since

ADSL provides a dedicated collision free connection), even though they have similar bandwidth. Because of contention, the inter-departure times of a cable modem are more random than those of an ADSL connection, leading to a larger entropy in the inter-arrival time of packet pairs. This intuition is confirmed by our experimental results.

2.3. ADSL, dialup, and Ethernet

ADSL has dedicated access with low bandwidth. Dialup has dedicated access and very low bandwidth. Switched Ethernet has dedicated access and high bandwidth. Non-switched Ethernet uses shared media, which, however, only causes a negligible amount of randomness since Ethernet has high bandwidth and the capability to detect collisions. In this paper, we do not differentiate switched and non-switched Ethernet, and refer to them loosely as high-bandwidth wired Ethernet or simply wired connection.

3. Classification scheme

Roughly, our classification scheme operates as follows. When node A needs to determine the connection type of node B, A asks B to send it a sequence of packet pairs. These packets are very small and, therefore, their disturbance to the network is negligible. For each packet pair from B, A records the inter-arrival time of the two packets in the pair. Then A determines B's connection type based on the median and entropy of the inter-arrival times. The main reason for A to determine B's connection type is to prevent B from incorrectly reporting its connection type to A.[1] In the rest of the paper, we refer to B as the *sender* and A as the *receiver*. We assume that the receiver is well-connected (i.e., it has a wired connection with high bandwidth). This is reasonable in the settings where a server classifies client connection types. In a peer-to-peer or overlay network, we can assume that one or several well-connected nodes are in charge of determining the connection types of the other nodes.

The intuition behind using median and entropy of packet-pair inter-arrival times is as follows. Median is useful for differentiating low-bandwidth and high-bandwidth connections—a low-bandwidth connection tends to produce a larger median value than a high-bandwidth connection. Furthermore, median is useful for differentiating WLAN and Ethernet connections—the slower carrier sensing and explicit ACK in WLAN leads to a larger median inter-arrival time than that seen in Ethernet. Entropy tracks the amount of randomness inherent in an access network: in a WLAN, a packet pair from a wireless station is separated by a random backoff duration, even when the channel has no contention or transmission errors; in a cable network, a packet pair from a CM is also separated by a random backoff duration for resolving contention among the CMs; the other types of connections do not use random backoff or the effect of random backoff is negligible.

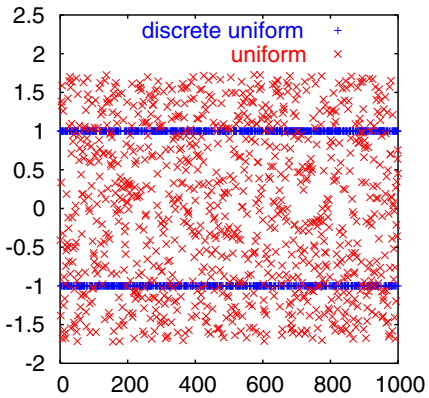


Fig. 1. Two distributions have the same mean and variance but very different entropies.

We use median instead of mean to reduce the effect of outliers in the measurements [15]. We use entropy instead of variance (or coefficient of variation) because it is a much better metric to capture the randomness of a random variable. A simple example illustrates this advantage. Suppose V_1 is a discrete random variable taking values +1 and -1 with equal probability, and V_2 is a continuous random variable that follows a uniform distribution on the interval of $[-\sqrt{3}, \sqrt{3}]$. Fig. 1 plots 1000 instances of V_1 and V_2. Both V_1 and V_2 have a variance of one. Therefore, they cannot be distinguished using variance. They, however, can be easily distinguished using entropy. More specifically, let $H_2(V_i)$ denote the entropy of V_i when using base 2 in the logarithm.[2] Then when discretizing the interval $[-\sqrt{3}, \sqrt{3}]$ into 1024 bins, we have $H_2(V_1) = 1$ bit and $H_2(V_2) = \log_2(1024) = 10$ bits, which easily distinguishes V_1 and V_2. Intuitively, entropy performs better because it captures the randomness of a random variable over the *entire* domain while variance only describes variations of a random variable around its mean. Our experimental results confirm that entropy is indeed a better statistic to classify access networks than variance (or coefficient of variation).

We next describe our classification scheme in detail. Let I denote the inter-arrival time of a packet pair from the sender to the receiver. Let $\xi_{.5}(I)$ and $H(I)$ denote the (population) median and entropy of I, respectively. In practice, we obtain estimates of $\xi_{.5}(I)$ and $H(I)$ through a sequence of samples. Let I_i denote the inter-arrival time of the two packets in the ith packet pair. Suppose the receiver receives n packet pairs. Then $\{I_i\}_{i=1}^n$ represents a sequence of packet-pair inter-arrival times. Let $\xi_{.5}^n(I)$ denote the sample median of $\{I_i\}_{i=1}^n$. To obtain the entropy, we discretize $\{I_i\}_{i=1}^n$ using a bin size of 300 μs or 900 μs, and calculate the entropy of the discretized values. For convenience,

[1] A malicious sender may manipulate the inter-sending time of a packet pair to hide its connection type. For instance, an Ethernet sender may purposely increase the inter-sending time of a packet pair so as to fake as a slow connection. On the other hand, it is difficult for a slow connection to reduce the inter-sending time of a packet to fake as a fast connection. Accurate connection type classification in the presence of malicious senders is left as future work.

[2] For a discrete random variable X taking value in $\mathcal{X}$ and having a probability mass function of $p(x) = P(X = x), x \in \mathcal{X}$, the entropy of X, $H(X)$, is defined as

$$H(X) = -\sum_{x \in \mathcal{X}} p(x) \log p(x).$$

The sender sends a sequence of packet pairs to the receiver;

The receiver obtains a sequence of packet-pair inter-arrival times, $\{I_i\}_{i=1}^{n}$;

Rule 1:

if $(\xi_{.5}^{n}(I) \leq 600\ \mu s$ **and** $H_2^{300}(I) < 2.0$ bits **and** $H_2^{900}(I) < 0.6$ bit)

then the sender uses Ethernet (high-bandwidth wired);

Rule 2:

else if $(350\ \mu s < \xi_{.5}^{n}(I) \leq 2$ ms **and** $H_2^{300}(I) \geq 0.8$ bit)

then the sender uses WLAN;

else the sender uses low-bandwidth connection (i.e., ADSL, cable or dialup).

Fig. 2. Classification scheme: the receiver classifies the connection type of the sender based on a sequence of packet pairs from the sender.

we use base 2 logarithm and denote the entropies at these two time scales as $H_2^{300}(I)$ and $H_2^{900}(I)$, respectively.

Our classification scheme is summarized in Fig. 2. First, the sender sends a sequence of packet pairs to the receiver. The receiver records the sequence of the inter-arrival times of the packet pairs, $\{I_i\}_{i=1}^{n}$, where n is the number of packet pairs received. *Rule 1* differentiates Ethernet and non-Ethernet connections. It states that if $\xi_{.5}^{n}(I) \leqslant 600$ μs, $H_2^{300}(I) < 2.0$ bits and $H_2^{900}(I) < 0.6$ bit, then the connection type is Ethernet. This rule is based mainly on our analytical results in Section 4. All three conditions in the rule (i.e., median, entropies at the time scales of 300 μs and 900 μs) are required to determine that a connection is Ethernet (see Section 5).

Rule 2 states that if a non-Ethernet connection has $\xi_{.5}^{n}(I)$ lying between 350 μs and 2 ms, and $H_2^{300}(I) \geqslant 0.8$ bit, then it is a WLAN connection. These conditions are based mainly on empirical results and partially supported by the analysis in Section 4. Connections not satisfying Rules 1 and 2 are labeled low-bandwidth connections, i.e., cable, ADSL or dialup. Our empirical results show that it is difficult to obtain a clear-cut distinction among these three low-bandwidth connections using median and entropy. Although the entropy of a cable connection can be much larger than that of ADSL due to contention in cable networks, it is difficult to differentiate cable and ADSL completely. This is because the upstream cable and ADSL connections have similar bandwidths. Moreover, the entropies of cable and ADSL connections are similar when there is little sharing or contention among the cable modems. Dialup may exhibit a much larger median value than cable and ADSL due to its low bandwidth. Roughly, we determine the connection to be dialup when $\xi_{.5}^{n}(I) \geqslant 10$ ms. However, we also observe very low values of $\xi_{.5}^{n}(I)$ for some dialup connections, which is likely due to traffic shaping, as observed in [16].

4. An analytical basis for classification

In this section, we present analytical models that provide the foundation for our classification scheme. Even though the models are idealized, they provide insight into defining appropriate classification rules that work extremely well in practice. In the following, we first describe the assumptions for the analysis, and then state the median and entropy results for an Ethernet connection. Last, we present results for non-Ethernet connections.

4.1. Assumptions

Consider a sender sending packet pairs to a receiver. We assume at most two low-bandwidth links from the sender to the receiver (since the backbone network is usually well provisioned). The remaining links have high bandwidths and negligible effect on packet-pair inter-arrival times at the receiver. We therefore ignore the high-bandwidth links and consider two settings, as illustrated in Fig. 3. In these two settings, the sender is connected to the receiver by one C Mbps link (Setting (a)) or by two C Mbps links (Setting (b)). We refer to the first link as L_1, and the second link as L_2 (in Setting (b)). Assume that packet arrivals to L_k are independent and follow a Poisson process. We model L_k as an $M/D/1$ queue, and let ρ_k denote the utilization of L_k, $0 \leqslant \rho_k \leqslant 1$, $k = 1,2$. Let Δ_k denote the inter-departure time of a packet pair at L_k, $k = 1,2$. For convenience, let Δ_0 denote the inter-departure time of a packet pair after the access link. For ease of analysis, we assume that all packets consist of S bytes.

To make the discussion concrete, we assume $C = 10$ Mbps. Measurement studies show that the average packet

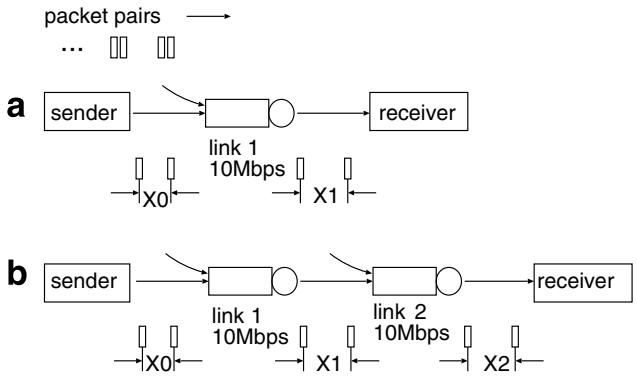


Fig. 3. Settings for the analysis: one and two 10 Mbps links connect the sender and the receiver in Settings (a) and (b), respectively, $X_k = \lceil \Delta_k / (300\ \mu s) \rceil$, $k = 0,1,2$.

size in the Internet is between 300 and 400 bytes [17,18]. We use $S = 375$ bytes for ease of computation. Let μ denote the bandwidth of L_k in packets per second, i.e., $1/\mu$ denotes the transmission time at L_k, $k = 1,2$. For a 375 byte packet and a bandwidth of 10 Mbps, $1/\mu = 300$ μs. For convenience, we discretize the inter-departure time of a packet pair, Δ_k, using a time unit of 300 μs, and denote the discretized value as X_k, that is, $X_k = \lceil \Delta_k \mu \rceil = \lceil \Delta_k / (300\ \mu s) \rceil$, $k = 0,1,2$.

Recall that I denotes packet-pair inter-arrival time at the receiver. We are interested in the median and entropy of I. In Setting (a), since the inter-arrival time of a packet pair at the receiver is the same as the inter-departure time of the packet pair at link L_1, we have $I = \Delta_1$. Similarly, in Setting (b), we have $I = \Delta_2$. We calculate the entropy of I using base 2 logarithm for two time scales, 300 μs and 900 μs, denoted, respectively, as $H_2^{300}(I)$ and $H_2^{900}(I)$. We next present analytical results for Ethernet and non-Ethernet connections.

4.2. Ethernet connections

We now state two theorems concerning the median and entropy of packet-pair inter-arrival times for a 100 Mbps Ethernet connection. These two theorems form the foundation of *Rule 1* in our classification scheme.

Theorem 1 (Median for Ethernet connection). *In Settings (a) and (b), the median inter-arrival time of 100 to 500 packet pairs at the receiver is below 600 μs with probability close to 1, i.e.,* $P(\xi_{.5}^n(I) \leqslant 600\ \mu s) \approx 1$, $n \in [100,500]$.

The above theorem provides an upper bound on the median packet pair inter-arrival time and is used in *Rule 1* of our classification scheme. Its proof is found in Appendix A.

Theorem 2 (Entropy for Ethernet connection). *In Setting (a),* $H_2^{300}(I) \leqslant 0.49$ *bit,* $H_2^{900}(I) \leqslant 0.07$ *bit. In Setting (b), when* $\rho_1 = 1$ *and* $\rho_2 = 1$, $H_2^{300}(I) = 1.99$ *bits,* $H_2^{900}(I) = 0.57$ *bit.*

The proof of Theorem 2 is found in Appendix B. Theorem 2 provides an upper bound of the entropy for Setting (a). This upper bound is achieved when $\rho_1 = 1$ since a high utilization at link L_1 makes it more likely for other packets to be inserted between a packet pair, leading to more randomness and hence a higher entropy. Theorem 2 also provides the entropies at the time scales of 300 and 900 μs when $\rho_1 = 1$ and $\rho_2 = 1$ in Setting (b). They are derived from $H(X_2)$. Numerical results indicate that $H(X_2|\rho_2 = 1)$ is an increasing function of ρ_1 and hence $H(X_2|\rho_2 = 1)$ obtains its maximum value when $\rho_1 = 1$ [19]. The intuition is that, when $\rho_2 = 1$, a larger value of ρ_1 can lead to greater randomness and hence larger entropies. We speculate that $H(X_2)$ is an increasing function of both ρ_1 and ρ_2, which is confirmed by our simulation results [19]. Then the entropies in Theorem 2 are upper bounds of $H_2(I)$, which are used in *Rule 1* of our classification scheme.

4.3. Non-Ethernet connections

We now analyze the case where the access network is not Ethernet. We first present a theorem when the access link uses IEEE 802.11b; its proof is found in Appendix C.

Theorem 3 (Median and entropy for 802.11b). *When using 11 Mbps 802.11b, under ideal conditions (with no contention, no retransmissions and perfect channel conditions), the median inter-departure time at the sender is greater than 800* μs, *and* $H_2^{300}(\Delta_0) > 1$ *bit.*

Although the above results are for packet-pair inter-departure time at the sender under idealized conditions, they provide important insights: we expect larger median and entropy values when the conditions are not ideal; we expect similar results for packet-pair inter-arrival times at the receiver when the intermediate links from the sender to the receiver are lightly utilized.

Theorems 1 and 3, and our empirical observations (in Section 5) form the foundation for *Rule 2* of our classification scheme. In *Rule 2*, the median threshold of 350 μs is based on Theorem 3 and empirical observations of 802.11g connections; the median threshold of 2 ms is based on empirical observations of cable, ADSL and dialup connections; the entropy threshold of 0.8 bit at the time-scale of 300 μs comes from Theorem 3 (relaxed to allow for sampling errors) and empirical results on 802.11g connections.

Last, we use several examples to illustrate that the entropy of a non-Ethernet connection can be much larger than that of an Ethernet connection. Table 1 lists several entropy values when $\rho_1 = 1$ and $\rho_2 = 1$ (see the proof of Theorem 2 in Appendix B for the calculation). In the table, the inter-departure time of a packet-pair at the sender, Δ_0, is 900 or 1200 μs, corresponding approximately to the range in an 802.11b WLAN. For low-bandwidth connections (i.e., cable, ADSL and dialup), Δ_0 can be much larger, and hence the corresponding entropy values can be much larger.

5. Experimental results

We have carried out extensive experiments over the Internet. These experiments serve two purposes. First, they validate our analytical results (in Section 4). Second, they provide several empirical results for classifying connection types. We designed and executed two sets of experiments. The first set includes *small-scale controlled experiments* performed by us. It involves 14 machines in 4 continents. The second set includes *large-scale uncontrolled experiments* performed by volunteers in 10 countries.

Most of the experiments were carried out from March to April 2004 when 802.11g WLAN was not widely deployed; additional experiments were carried out from September to November 2006 using 802.11g WLAN senders. We next describe the experimental results in detail, and summarize the key insights at the end.

Table 1
A few examples of entropy values, $\rho_1 = 1$, $\rho_2 = 1$

Δ_0 (μs)	Setting (a)		Setting (b)	
	$H_2^{300}(I)$ (bit)	$H_2^{900}(I)$ (bit)	$H_2^{300}(I)$ (bit)	$H_2^{900}(I)$ (bit)
900	2.79	1.05	3.33	1.61
1200	3.01	1.39	3.54	1.87

5.1. Small-scale controlled experiments

This set of experiments involves 14 Linux machines, using Ethernet, WLAN, cable or ADSL connection. Five machines are at University of Massachusetts, Amherst (UMass), using Ethernet or WLAN connections. In particular, two machines, UMass-1 and UMass-2, use 10 Mbps and 100 Mbps Ethernet connections, respectively. Three machines, UMass-w1, UMass-w2 and UMass-w3, use WLAN: UMass-w1 uses 11 Mbps 802.11b, UMass-w2 uses 22 Mbps 802.11b, and UMass-w3 uses 54 Mbps 802.11g. Two machines, Home-1 and Home-2, are at a residential home in Amherst, MA, where both cable and ADSL connections are installed. Home-1 connects to the Internet through a router at the residence. Home-2 has a 22 Mbps 802.11b WLAN card, and connects to the Internet through a wireless access point at the residence. The rest of the machines are located at university sites. One machine in the east coast (University of Connecticut (UConn)) uses 54 Mbps 802.11g; all the other machines use Ethernet connections, and are located in the east coast (University of Pennsylvania (UPenn)), middle west (University of Minnesota (UMN)), west coast (University of Southern California (USC)) of the US, Brazil, Taiwan and Italy.

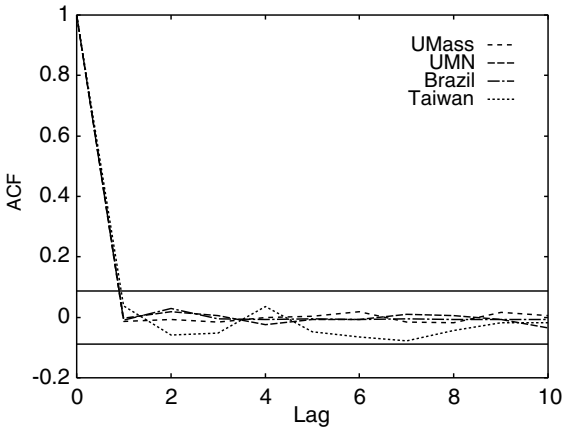


Fig. 4. Autocorrelation function of the sequence of packet pair inter-arrival times. The receiver is at USC; the senders are at UMass, UMN, Brazil and Taiwan.

Machines with Ethernet connections act as receivers. In each experiment, the sender sends a packet pair every 20 ms or 40 ms with a total of 500 packet pairs. Therefore, each experiment lasts for 10 or 20 s. The receiver records the arrival time of each packet using *tcpdump* [20] and calculates the inter-arrival time of the two packets in each packet pair. Two Linux machines in Brazil and UPenn were too old to capture timestamps accurately. We therefore use 6 machines (UMass-1, UMass-2, USC, UMN, Taiwan and Italy) as receivers.

We ran 106 experiments in total. For each experiment, we validate that packet-pair inter-arrival times at the receiver can be regarded as independent random variables. As an example, Fig. 4 plots the autocorrelation function of the inter-arrival time sequence for a receiver at USC and senders at UMass, UMN, Brazil and Taiwan. We observe that the autocorrelation function falls into the confidence interval at the various lags, indicating that the packet-pair inter-arrival times are not correlated.

We next visually examine packet-pair inter-arrival times when the sender uses different connection types. Fig. 5(a) and (b) plot packet-pair inter-arrival times when the senders use Ethernet and WLAN, respectively. For the Ethernet connection, the inter-arrival times are much lower and more regular than those for the WLAN connection, indicating a lower median inter-arrival time and entropy. Fig. 6(a) and (b) plot the inter-arrival times when the senders use cable and ADSL, respectively. For the cable connection, the inter-arrival times range from 0 to 70 ms, while for the ADSL connection, the inter-arrival times mostly lie in [4,6] ms. This indicates that the entropy of the cable connection is larger than that of the ADSL connection.

Fig. 7(a) and (b) plot the classification results for these 106 experiments at the timescales of 300 and 900 μs, respectively. The solid lines represent the bounds for Ethernet connections in our scheme: the solid horizontal line corresponds to median of 600 μs; the two vertical solid lines correspond to 2.0 bits and 0.6 bit at 300 and 900 μs timescales, respectively. We observe that all the Ethernet connections satisfy the three criteria in our classification scheme, and no other connection type satisfies these three criteria simultaneously. The dashed lines represent the bounds for WLAN connections in our scheme:

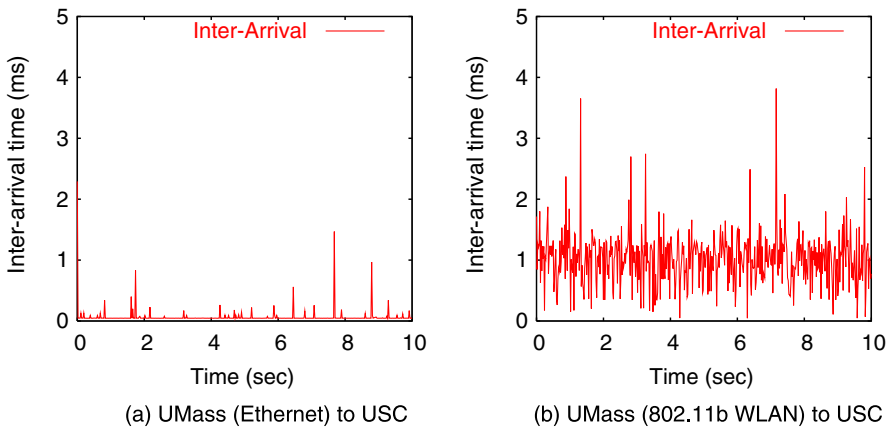


Fig. 5. Packet-pair inter-arrival time: Ethernet and WLAN connections.

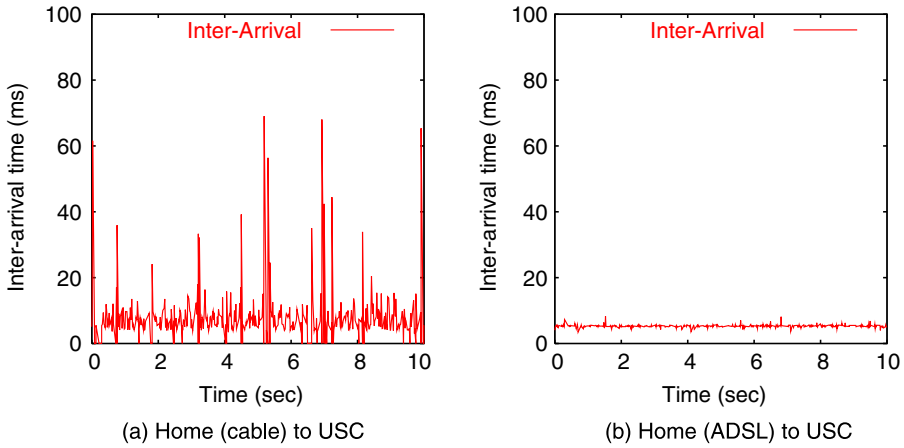
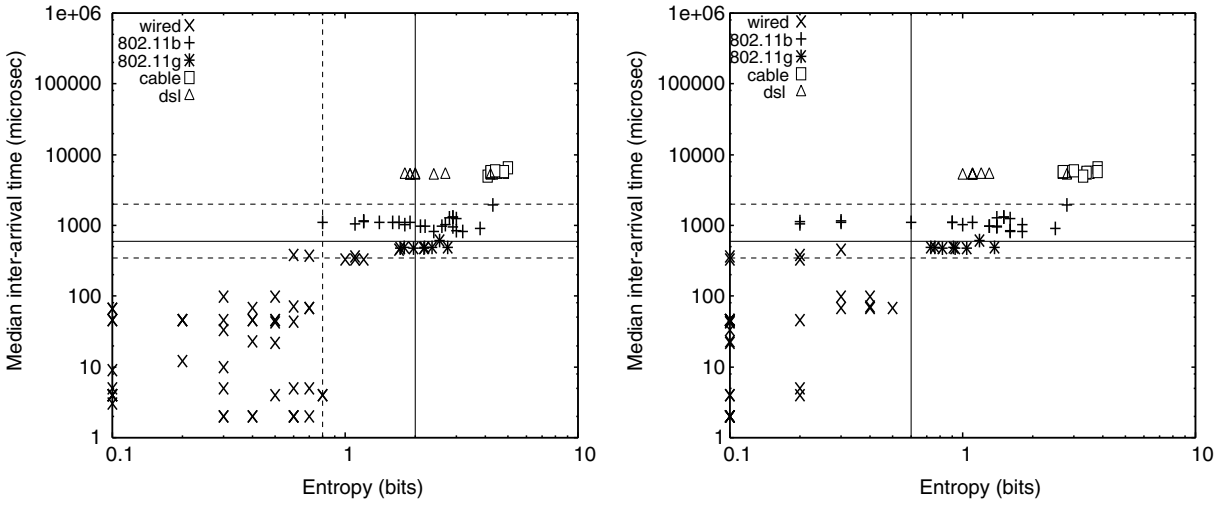


Fig. 6. Packet-pair inter-arrival time: cable and ADSL connections.



(a) Timescale of 300 μs. (b) Timescale of 900 μs.

Fig. 7. Small-scale controlled experiments: classification results.

the two dashed horizontal lines correspond to medians of 350 μs and 2 ms; the dashed vertical line corresponds to entropy of 0.8 bit.

All 802.11b WLAN connections' median inter-arrival times are between 600 μs and 2 ms. Their entropies at 300 μs timescale are larger than 1 bit except one case (value of 0.8 bit, where the sender and the receiver are in the same domain). All 802.11g WLAN connections' median inter-arrival times are between 400 μs and 2 ms, and their entropies at 300 μs timescale are larger than 1 bit. We observe that three 802.11g experiments have medians less than 600 μs and entropies less than 2 bits at 300 μs timescale. However, their entropies at 900 μs timescale are larger than 0.6 bit, and hence they are correctly classified as non-Ethernet connections. All the cable and ADSL connections have medians larger than 2 ms. The cable connections exhibit large entropies than ADSL connections, indicating a larger amount of randomness due to contention in cable networks.

To gain additional insights, we list several results when the sender and the receiver are at UMass in Table 2. The sender in the first experiment uses Ethernet connection; the senders in the remaining experiments use WLAN connections (802.11b or 802.11g). It is interesting to note that, even when the sender and receiver are in the same domain, the median inter-arrival time for a WLAN connection is larger than 400 μs, and the 300 μs timescale entropy is generally larger than 1 bit, except for one case (where the value is 0.8 bit).

5.2. Large-scale uncontrolled experiments

To increase the scale of the experiments, we created a Windows program that sends UDP packet pairs to two receivers at UMass (i.e., UMass-1 and UMass-2). The program sends 500 packet pairs, one every 200 ms (to accommodate low-bandwidth dialup connections). The receivers run *tcpdump* [20] to collect the arrival time of each packet and calculate the inter-arrival time of each packet pair. We distributed the sender program to friends at different locations and asked them to run it on their local machines. The senders are located in 28 states in the US (illustrated by the

Table 2
Small-scale controlled experiments: both the sender and the receiver are at UMass

Sender	Receiver	Sample size (μs)	Min (μs)	Max (μs)	Median (μs)	$H_2^{300}(I)$ (bit)	$H_2^{900}(I)$ (bit)
UMass-2	UMass-1	500	1	190	16	0.0	0.0
UMass-w1	UMass-2	500	79	8246	1099	0.8	0.3
UMass-w1	UMass-2	500	873	3959	1138	1.2	0.2
UMass-w1	UMass-1	500	693	8644	1117	1.4	0.6
UMass-w1	UMass-1	500	845	5700	1115	1.7	0.9
UMass-w2	UMass-1	500	193	7567	1096	1.9	1.1
UMass-w3	UMass-1	500	177	7788	480	1.9	0.9

shaded regions in Fig. 8) and 9 other countries (Brazil, Canada, China, Germany, Israel, Japan, Korea, Norway, United Kingdom). We collected 473 traces in total. Table 3 lists the breakdown of the traces. We next describe the classification results, and then explore the sensitivity of our scheme to the number of packet pairs.

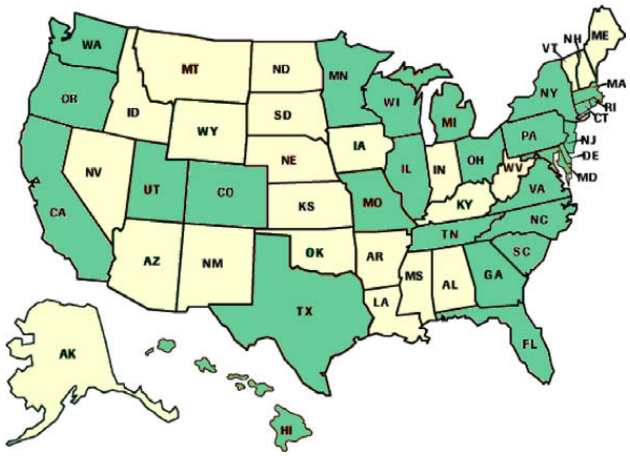


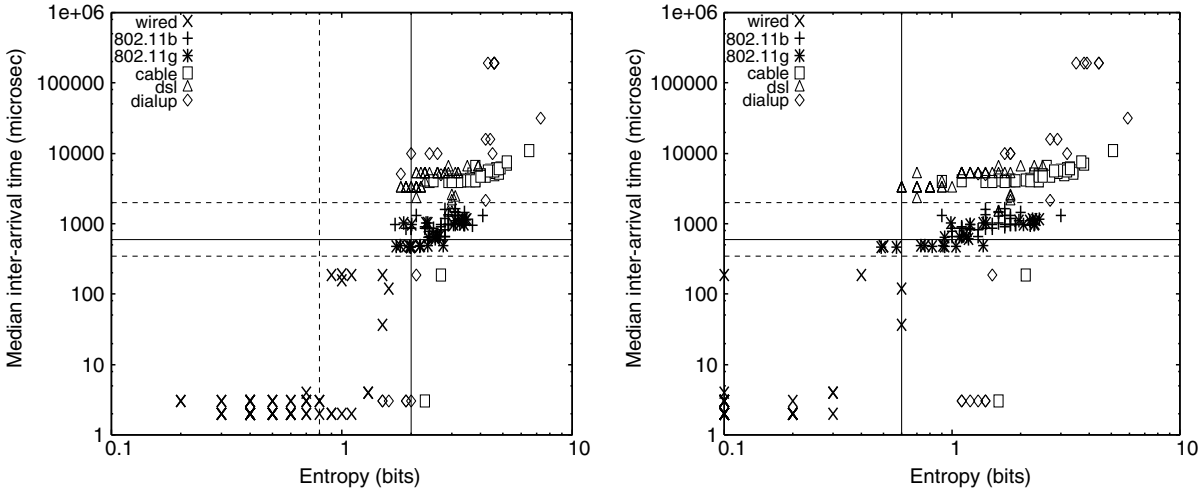
Fig. 8. Large-scale uncontrolled experiments: trace coverage (illustrated by the shaded regions) in the US.

5.2.1. Classification results

Fig. 9 plots the median and entropy of the inter-arrival times for all the experiments where UMass-1 is the receiver. We observe that all Ethernet connections satisfy the three criteria in our classification scheme. Furthermore, approximately half of the Ethernet connections satisfy the entropy upper bounds when a single bottleneck link is between the sender and receiver (i.e., bounds of 0.5 and 0.1 bit at the timescales of 300 μs and 900 μs, respectively, from Theorem 2). For 802.11b WLAN connections, the medians are in the range of 600 μs and 2 ms, and the entropies at 300 μs timescale are larger than 1 bit, satisfying the criteria for WLAN connection in our classification scheme. For 802.11g WLAN connections, the medians are in the range of 350 μs and 2 ms, and the entropies at 300 μs timescale are larger than 0.8 bit. All the traces from

Table 3
Large-scale uncontrolled experiments: breakdown of the traces

Receiver	Ethernet	802.11b	802.11g	Cable	ADSL	Dialup
UMass-1	74	37	26	39	46	22
UMass-2	70	33	26	35	46	19



(a) Timescale of 300 μs. (b) Timescale of 900 μs.

Fig. 9. Large-scale uncontrolled experiments: classification results for all connection types (UMass-1 as the receiver).

cable, ADSL and dialup connections have a median outside the range of 600 μs to 2 ms except for two traces from an ADSL connection, which are misclassified as WLAN.

We now examine cable, ADSL and dialup connections more closely. Fig. 10 plots the median and entropy of the inter-arrival times for these three types of connections. The solid horizontal lines represent medians of 600 μs and 2 ms. We observe that it is difficult to obtain a clear-cut differentiation among the three low-bandwidth connections using median and entropy. The medians of ADSL connection traces are scattered in a narrower range than those of cable and dialup connection traces. The majority of the traces have medians larger than 2 ms. However, two cable connection traces and five dialup connection traces have medians as low as a few microseconds (These traces are nonetheless correctly classified as non-Ethernet connections since their entropies at 900 μs timescale exceed 0.6 bit.). We speculate that these low medians are caused by traffic shaping, as observed in [16]. Fig. 11 shows an example of a dialup connection with small median inter-arrival time. We observe that although majority of the inter-arrival times are less than 10 μs, a significant fraction of the inter-arrival times are in a wide range of 1000 to 6000 μs. The widespread of inter-arrival times results in a large entropy, which differentiates it from an Ethernet connection.

To summarize, for all the 473 experiments, our classification scheme distinguishes between Ethernet and non-Ethernet connections accurately in all but one experiment. In the misclassified experiment, the sender uses an 802.11g connection, the median is 369 μs, the entropies are 1.9 bits and 0.4 bit at 300 μs and 900 μs timescales, respectively. Of the 122 WLAN traces, our classification scheme identifies 121 traces correctly (one 802.11g trace was misclassified as Ethernet). However, 5 of the 92 traces from ADSL connections are misclassified as WLAN connections. All of these misclassified traces are from Calgary, Canada.

5.2.2. Sensitivity to number of packet pairs

So far, we use all the packet pairs (up to 500 pairs) for connection type classification in each experiment. We now explore the sensitivity of our scheme to the number of pack-

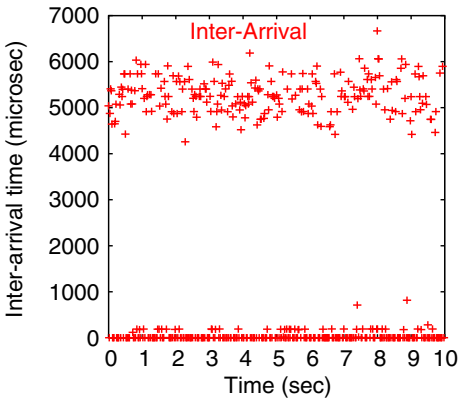


Fig. 11. An example of packet-pair inter-arrival times of a dialup connection: the small inter-arrival times might be due to traffic shaping.

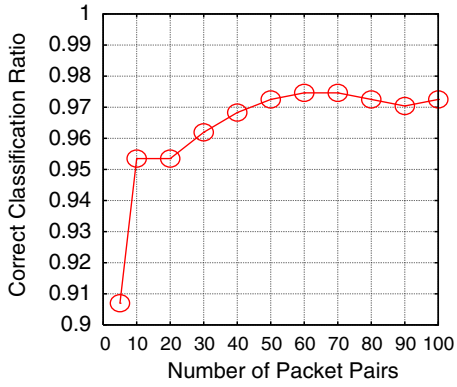
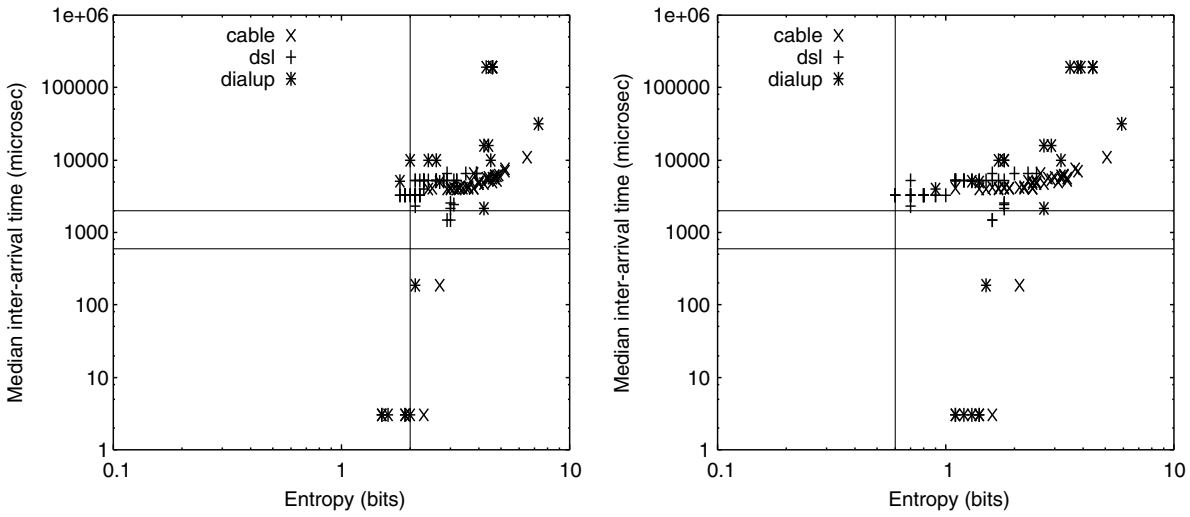


Fig. 12. Correct classification ratio versus the number of packet pairs.



(a) Timescale of 300 μs. (b) Timescale of 900 μs.

Fig. 10. Large-scale uncontrolled experiments: classification results for cable, ADSL and dialup connections (UMass-1 as the receiver): (a) timescale of 300 μs and (b) timescale of 900 μs.

et pairs. More specifically, we use the first n packet pairs of each experiment for connection type classification, where n is varied from 5 to 100. Fig. 12 plots the correct classification ratio versus the number of packet pairs over all the 473 experiments. We observe that, even for 5 packet pairs, the correct classification ratio is above 90%. The correct classification ratio exceeds 95% with 10 packet pairs (i.e., in 2 s) and exceeds 97% with 50 packet pairs (i.e., in 10 s). The above results indicate that our scheme makes accurate decisions with very few packet pairs in a very short time.

6. Conclusion and future work

Ethernet, wireless LAN, ADSL, cable modem and dialup are common connection types that have dramatically different characteristics. Fast and accurate classification of connection types can improve the performance of network protocol and applications dramatically. In this paper, we proposed a simple and efficient end-to-end scheme to classify access links into three categories: Ethernet, wireless LAN and low-bandwidth wired connection. Our scheme is based on the intrinsic characteristics of the various connection types and utilized both the median and the entropy of packet-pair inter-arrival times. Extensive experiments showed that our scheme obtains accurate classification results in a very short time.

As future work, we are pursuing in the following directions: (i) investigating more complicated probing techniques (e.g., probes of different sizes and non-periodic probing) that might potentially reveal more clearly the characteristics of an access network; (ii) differentiating cable and ADSL connections based on more specific characteristics of these two types of connections, (iii) investigating connection type classification when the sender can be malicious (e.g., it may manipulate the inter-sending time of a packet pair), and (iv) classifying connection types when considering other emerging access networks, such as cellular networks and FiOS (Fiber Optic Service).

Acknowledgements

A preliminary version of this paper appeared in Infocom 2005 [21]. This research was supported in part by the National Science Foundation under NSF Grants ANI-0240487, ANI-9980552, ANI-0085848, EEC-0313747 and EIA-0080119. We thank L. Golubchik, R. Guerin, C. Papadopulous, F. L. Presti, E. de Souza e Silva, R.-H. Hwang and Z.-L. Zhang for providing us accounts for the Internet experiments. We would also like to thank our friends and colleagues for helping us with the experiments. This work would not have been possible without their help.

Appendix A. Proof of Theorem 1

We first present two lemmas on the distribution of packet-pair inter-departure times at an $M/D/1$ queue. These two lemmas are used to prove Theorems 1 and 2.

Lemma 1. *Consider an $M/D/1$ queue with processing rate μ and utilization ρ. Let Δ_a denote the inter-arrival time of a packet pair at the queue. Let Δ_d denote the inter-departure time of the packet pair after the queue. Furthermore, let $X_a = \Delta_a\mu$ and $X_d = \Delta_d\mu$. When $0 \leqslant x_a \leqslant 1$, we have*

$$P(X_d = x_d | X_a = x_a, \rho) = \frac{e^{-x_a\rho}(x_a\rho)^{x_d-1}}{(x_d-1)!}$$

where $x_d = 1,2,\ldots$.

Proof. Without loss of generality, suppose the two packets in the packet pair arrive at the queue at time 0 and Δ_a respectively. Let X denote the number of packet arrivals between time 0 and Δ_a. From the Poisson arrival assumption, X follows a Poisson distribution with parameter of $x_a\rho$ given $X_a = x_a$. Since $x_a \leqslant 1$, the first packet of the packet pair has not departed from the queue when the second packet arrives. Therefore, $X_d = X + 1$ and we have the desired result. □

Lemma 2. *Under the conditions of Lemma 1, if $x_a > 1$ and $\rho = 1$, then*

$$P(X_d = x_d | X_a = x_a, \rho = 1) = \frac{e^{-x_a}x_a^{x_d-1}}{(x_d-1)!}$$

where $x_d = 1,2,\ldots$.

Proof. Without loss of generality, suppose the two packets in the packet pair arrive at the queue at time 0 and Δ_a, respectively. Let X denote the number of packet arrivals between time 0 and Δ_a. We prove this lemma by considering the following two cases:

- *Case 1:* When the second packet arrives at the queue, the first packet has not departed from the queue. This is the same as the situation in Lemma 1. Similar to the proof of Lemma 1, we have $X_d = X + 1$.
- *Case 2:* When the second packet arrives at the queue, the first packet has departed from the queue. Let q_1 and q_2 be the queue length seen by the first and second packet, respectively. The departure time of the first and the second packet is $(q_1 + 1)/\mu$ and $\Delta_a + (q_2 + 1)/\mu$, respectively. Hence, $\Delta_d = \Delta_a + (q_2 - q_1)/\mu$, which implies

  $$X_d = X_a + q_2 - q_1. \tag{A.1}$$

  Now let us consider the relationship between q_1 and q_2. At time 0, there are $q_1 + 1$ packets in the queue. Since $\Delta_a = x_a/\mu$ and $\rho = 1$, there are x_a packet departure events between the arrival of the two packets in the packet pair. Under the assumption that $\rho = 1$, the probability that the queue is empty is 0. Hence, $q_2 = q_1 + 1 + X - X_a$. This implies

  $$q_2 - q_1 = 1 + X - X_a. \tag{A.2}$$

  Combining Eqs. (A.1) and (A.2), we have $X_d = X + 1$.

From Cases 1 and 2, we have $X_d = X + 1$ and X follows a Poisson distribution with parameter of $x_a\rho = x_a$ when $X_a = x_a$ and $\rho = 1$. We therefore have the desired result. □

Proof of Theorem 1:

Proof. Under the assumption that the bandwidth of the sender is 100 Mbps and the first low-bandwidth interme-

diate link is 10 Mbps, we have $\Delta_0 = 30\ \mu s = 0.1/\mu$. That is, $X_0 = 0.1$. By Lemma 1,

$$P(X_1 \leqslant 2|X_0 = 0.1, \rho_1) = e^{-0.1\rho_1} + 0.1\rho_1 e^{-0.1\rho_1} \geqslant e^{-0.1} + 0.1e^{-0.1} = 0.995$$

We first prove the theorem in Setting (a). Since $I = \Delta_1$ in Setting (a), we only need to show that $P(\xi^n_{.5}(\Delta_1) \leqslant 600\ \mu s) \approx 1$. Let $p = P(X_1 \leqslant 2|\rho_1) = P(\Delta_1 \leqslant 600\mu s)$. Then

$$P(\xi^n_{.5}(\Delta_1) \leqslant 600\ \mu s) = \sum_{i=n/2}^{n} \binom{n}{i} p^i (1-p)^{n-i}. \quad (3)$$

When n lies between 400 and 500, we prove that $P(\xi^n_{.5}(\Delta_1) \leqslant 600\ \mu s)$ is an increasing function of p. This is accomplished by showing that the derivative of (3) with respect to p is positive. When $p = 0.995$, $\sum_{i=n/2}^{n} \binom{n}{i} p^i (1-p)^{n-i} \approx 1$. Since $p \geqslant 0.995$ and $P(\xi^n_{.5}(\Delta_1) \leqslant 600\ \mu s)$ is an increasing function of p, we have $P(\xi^n_{.5}(\Delta_1) \leqslant 600\ \mu s) \approx 1$.

We next prove the above theorem in Setting (b). Since $I = \Delta_2$ in Setting (b), we only need to show that $P(\xi^n_{.5}(\Delta_2) \leqslant 600\ \mu s) \approx 1$. We next only show that $P(\Delta_2 \leqslant 600\ \mu s) < 0.657$; the rest of the proof is similar to that in Setting (a),

$$\begin{aligned} P(\Delta_2 \leqslant 600\ \mu s) &= \sum_{x_1=1}^{\infty} P(X_1 = x_1|X_0 = 0.1) P(\Delta_2 \\ &\leqslant 600\ \mu s|X_1 = x_1) > P(X_1 = 1|X_0 = 0.1) P(\Delta_2 \\ &\leqslant 600\ \mu s|X_1 = 1) = e^{-0.1\rho_1} P(X_2 \leqslant 2|X_1 = 1) \\ &\geqslant e^{-0.1}[P(X_2 = 1|X_1 = 1) + P(X_2 = 2|X_1 = 1)] \\ &= 0.905 \times (e^{-\rho_1} + e^{-\rho_1}\rho_1) \geqslant 0.905 \times 2e^{-1} = 0.657 \end{aligned}$$

The inequality $e^{-\rho_1} + e^{-\rho_1}\rho_1 \geqslant 2e^{-1}$ holds because $e^{-\rho_1} + e^{-\rho_1}\rho_1$ is a decreasing function of ρ_1. □

Appendix B. Proof of Theorem 2

We first prove a Lemma that is to be used to prove Theorem 2.

Lemma 3. *Let Y be a random variable, $Y \sim$ Poisson(α), $0 \leqslant \alpha \leqslant 1$. Let $H(Y|\alpha)$ denote the entropy of Y given α. Then $H(Y|\alpha)$ is an increasing function of α, and $H_2(Y|\alpha = 0.1) = 0.49$ bit.*

Proof

$$\begin{aligned} H(Y|\alpha) &= \sum_{i=0}^{\infty} \frac{e^{-\alpha}\alpha^i}{i!}[\log(e^{-\alpha}\alpha^i) - \log(i!)] \\ &= \alpha - \alpha\log\alpha + \sum_{i=0}^{\infty} \frac{e^{-\alpha}\alpha^i \log(i!)}{i!} \end{aligned}$$

We now obtain the derivative of $H(Y|\alpha)$ with respect to α

$$\frac{dH(Y|\alpha)}{d\alpha} = -\log\alpha + e^{-\alpha}\sum_{i=0}^{\infty} \frac{\alpha^i \log(i+1)}{i!} > -\log\alpha \geqslant 0$$

Therefore, $H(Y|\alpha)$ is an increasing function of α. One can prove that the infinite sum converges. We obtain $H_2(Y|\alpha = 0.1) = 0.49$ through direct calculation. □

Proof of Theorem 2:

Proof. We first prove the theorem in Setting (a). Since $I = \Delta_1$ in Setting (a), we only need to prove that $H_2^{300}(\Delta_1) \leqslant 0.49$ bit and $H_2^{900}(\Delta_1) \leqslant 0.07$ bit. By Lemma 1,

$$P(X_1 = x_1|X_0 = 0.1, \rho_1) = \frac{e^{-0.1\rho_1}(0.1\rho_1)^{x_1-1}}{(x_1-1)!}$$

Let Z be a random variable and $Z \sim$ Poisson($0.1\rho_1$). It is easy to show that $H_2(X_1) = H_2(Z)$. Since $0 < \rho_1 \leqslant 1$, by Lemma 3, we have $H_2(Z) \leqslant H_2(Z|\rho_1 = 1) = 0.49$ bit. Therefore, $H_2^{300\ \mu s}(\Delta_1) = H_2(X_1) \leqslant 0.49$ bit. The result for the time scale of 900 μs is obtained by change of variables.

We now prove the theorem in Setting (b). Since $I = \Delta_2$ in Setting (b), we only need to prove that $H_2^{300}(\Delta_2) \leqslant 1.99$ bit and $H_2^{900}(\Delta_2) \leqslant 0.57$ bit. When $\rho_2 = 1$, we calculate the entropy of X_2 as

$$H(X_2|\rho_2 = 1) = -\sum_{x_2=1}^{\infty} P(X_2 = x_2|\rho_2 = 1)\log(P(X_2 = x_2|\rho_2 = 1))$$

We obtain $P(X_2 = x_2|\rho_2 = 1)$ as follows.

$$P(X_2 = x_2|\rho_2 = 1) = \sum_{x_1=1}^{\infty} P(X_1 = x_1) P(X_2 = x_2|X_1 = x_1,$$

$$\rho_2 = 1) = \sum_{x_1=1}^{\infty} \frac{e^{-0.1\rho_1}(0.1\rho_1)^{x_1-1}}{(x_1-1)!} \frac{e^{-x_1}(x_1)^{x_2-1}}{(x_2-1)!}$$

where $P(X_1 = x_1)$ and $P(X_2 = x_2|X_1 = x_1, \rho_2 = 1)$ are from Lemmas 1 and 2, respectively. Note that we require $\rho_2 = 1$ to use Lemma 2. When $\rho_1 = \rho_2 = 1$, we obtain $H_2^{300\ \mu s}(\Delta_2)$ by direct calculation from the above. The result for the time scale of 900 μs is obtained by change of variables. □

Appendix C. Proof of Theorem 3

Proof. The proof utilizes the transmission overhead per frame when using 11 Mbps 802.11b WLAN, as listed in Table C.1 [22]. The random backoff follows a uniform distribution in the range of 0 to 31 slots with the slot time of 20 μs. In other words, the random backoff is in the range of 0 to 620 μs, with the mean and median of 310 μs. Two consecutive packets from the same wireless station are separated by a random backoff, even under ideal condi-

Table C.1
Breakdown of transmission overhead per frame in 11 Mbps 802.11b [22]

Overhead type	Time (μs)	Comments
PHY	192	Includes PLCP header and the physical layer preamble
MAC	24.7	Time to transmit 34 bytes of MAC header at 11 Mbps
IP & UDP	20.4	Transmission time for 28 bytes of IP and UDP headers
ACK	202.2	ACK transmission time including associated PHY overhead
SIFS	10	After frame is received but before ACK is sent
DIFS	50	Minimum idle time to be observed before backoff starts
Backoff	310	Average backoff value
Total	809.3	

tions. Therefore, from Table C.1, the average (as well as the median) of the inter-departure times of a packet pair at the wireless station is above 800 μs under ideal conditions.

Since the range of the inter-departure times of a packet pair at the sender is 620 μs, this range is divided into three bins under the time scale of 300 μs. The entropy obtains the minimum value of 1.2 bits in the following case: the probability of falling into two of the three bins is 300/620 and the probability of falling into the third bin is 20/620. Therefore, $H_2^{300\ \mu s}(\Delta_0) > 1$. □.

References

[1] D.G. Andersen, H. Balakrishnan, M.F. Kaashoek, R. Morris, Resilient overlay networks, in: Proceedings of the ACM SOSP, October 2001.
[2] S. Banerjee, B. Bhattacharjee, C. Kommareddy, Scalable application layer multicast, in: Proceedings of the ACM SIGCOMM, August 2002.
[3] M. Gerla, R. Bagrodia, L. Zhang, K. Tang, L. Wang, TCP over wireless multihop protocols: Simulation and experiments, in: Proceedings of the IEEE ICC, June 1999.
[4] B.S. Bakshi, P. Krishna, N.H. Vaidya, D.K. Pradhan, Improving performance of TCP over wireless networks, in: International Conference on Distributed Computing Systems (ICDCS), Baltimore, MD, USA, May 1997.
[5] R. Cohen, S. Ramanathan, TCP for high performance in hybrid fiber coaxial broad-band access networks, IEEE/ACM Transactions on Networking 6 (1) (1998) 15–29.
[6] R. Carter, M. Crovella, Measuring bottleneck link speed in packet-switched networks, Performance Evaluation 27 (8) (1996) 297–318.
[7] C. Dovrolis, P. Ramanathan, D. Moore, What do packet dispersion techniques measure? in: Proceedings of the IEEE INFOCOM, Anchorage, AL, 2001.
[8] S. Biaz, N. Vaidya, Discriminating congestion losses from wireless losses using inter-arrival times at the receiver, in: IEEE Symposium on Application-Specific Systems and Software Engineering Technology (ASSET), March 1999.
[9] L. Cheng, I. Marsic, Fuzzy reasoning for wireless awareness, International Journal of Wireless Information Networks 8 (1) (2001).
[10] W. Wei, S. Jaiswal, J. Kurose, D. Towsley, Identifying 802.11 traffic from passive measurements using iterative Bayesian inference, in: Proceedings of IEEE INFOCOM, April 2006.
[11] V. Baiamonte, K. Papagiannaki, G. Iannaccone, Detecting 802.11 wireless hosts from remote passive observations, in: Proceedings of IFIP/TC6 Networking, Atlanta, GE, May 2007.
[12] W. Wei, K. Suh, B. Wang, Y. Gu, J. Kurose, D. Towsley, Passive online rogue access point detection using sequential hypothesis testing with TCP ACK-pairs, in: ACM SIGCOMM Conference on Internet Measurement (IMC), October 2007.
[13] LAN/MAN standards of the IEEE Computer Society. Wireless LAN medium access control (MAC) and physical layer (PHY) specification. IEEE standard 802.11, 1997.
[14] CableLabs, Data-over-cable service interface specifications – radio frequency interface specification, in: MCNS Consortium, 2000, SP-RFI v1.1-106-001215.
[15] K. Papagiannaki, S. Moon, C. Fraleigh, P. Thiran, C. Diot, Measurement and analysis of single-hop delay on an IP backbone network, IEEE JSAC Special Issue on Internet and WWW Measurement Mapping and Modeling 21 (6) (2003).
[16] K. Lakshminarayanan, V.N. Padmanabhan, Some findings on the network performance of broadband hosts, in: ACM SIGCOMM Conference on Internet Measurement (IMC), Miami Beach, FL, October 2003.
[17] K. Thompson, G. Miller, R. Wilder, Wide-area Internet traffic patterns and characteristics, IEEE Network 11 (1997) 10–23.
[18] Packet trace analysis. <http://ipmon.sprintlabs.com/packstat/packetoverview.php>.
[19] W. Wei, B. Wang, C. Zhang, J. Kurose, D. Towsley, Classification of access network types: Ethernet, wireless LAN, ADSL, cable or dialup? Technical Report 04-46, Department of Computer Science, University of Massachusetts, Amherst, 2004.
[20] tcpdump,' <http://www.tcpdump.org/>.
[21] W. Wei, B. Wang, C. Zhang, J. Kurose, D. Towsley, Classification of access network types: Ethernet, wireless LAN, ADSL, cable or dialup?, in: Proceedings of the IEEE INFOCOM, Miami, FL, March 2005.
[22] S. Garg, M. Kappes, A.S. Krishnakumar, On the effect of contention-window sizes in IEEE 802.11b networks, Technical Report ALR-2002-024, Avaya Labs Research, 2002.

Wei Wei received his B.S. degree in Applied Mathematics from Beijing University, China in 1992, and M.S. degree in Statistics from Texas A&M University in 2000. He then received M.S. degrees in Computer Science and Applied Mathematics, and a Ph.D. in Computer Science from the University of Massachusetts, Amherst in 2004, 2004, and 2006, respectively. He is currently a visiting assistant professor of the Computer Science and Engineering Department at the University of Connecticut. His research interests are in the areas of computer networks, distributed embedded systems and performance modeling. He is a member of ACM and IEEE.

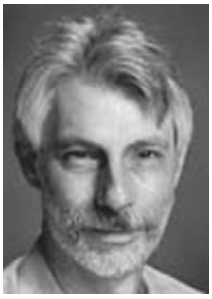
Bing Wang received her B.S. degree in Computer Science from Nanjing University of Science and Technology, China in 1994, and M.S. degree in Computer Engineering from Institute of Computing Technology, Chinese Academy of Sciences in 1997. She then received M.S. degrees in Computer Science and Applied Mathematics, and a Ph.D. in Computer Science from the University of Massachusetts, Amherst in 2000, 2004, and 2005, respectively. Afterwards, she joined the Computer Science & Engineering Department at the University of Connecticut as an assistant professor. Her research interests are in Computer Networks, Multimedia, and Distributed Systems. More specifically, she is interested in topics on Internet technologies and applications, wireless and sensor networks, overlay networks, content distribution, network management and measurement, network modeling and performance evaluation. She is a member of ACM, ACM SIGCOMM, IEEE, IEEE Computer Society, and IEEE Communications Society. She received NSF CAREER award in 2008.

Chun Zhang received his B.S. and M.S. degree in computer science from University of Science and Technology of China, Hefei in 1996 and 1999, and his Ph.D. degree in computer science from University of Massachusetts at Amherst in 2006, respectively. He is currently a research staff member at IBM T.J. Watson Research Center, Hawthorne, NY. His research interest areas include wired and wireless networking, performance modeling and optimization, services computing and distributed computing.

Jim Kurose received his Ph.D. degree in computer science from Columbia University, and is currently Distinguished University Professor in the Department of Computer Science at the University of Massachusetts. Professor Kurose has been a Visiting Scientist at IBM Research, INRIA, Institut EURECOM , the University of Paris, LIP6, and Thomson Research Labs.

His research interests include network protocols and architecture, network measurement, sensor networks, multimedia communication, and modeling and performance evaluation. Kurose has served as Editor-in-Chief of the IEEE Transactions on Communications and was the founding Editor-in-Chief of the IEEE/ACM Transactions on Networking. He has been active in the program committees for IEEE Infocom, ACM SIGCOMM, and ACM SIGMETRICS for a number of years, and has served as Technical Program Co-Chair for these conferences.

He has received a number of awards for his educational activities, including the IEEE Taylor Booth Education Medal. With Keith Ross, he is the co-author of the textbook, "Computer Networking, a top down approach (4th ed.)" published by Addison-Wesley Longman.



Don Towsley holds a B.A. in Physics (1971) and a Ph.D. in Computer Science (1975) from University of Texas. From 1976 to 1985 he was a member of the faculty of the Department of Electrical and Computer Engineering at the University of Massachusetts, Amherst. He is currently a Distinguished Professor at the University of Massachusetts in the Department of Computer Science. He has held visiting positions at IBM T.J. Watson Research Center, Yorktown Heights, NY; Laboratoire MASI, Paris, France; INRIA, Sophia-Antipolis, France; AT&T Labs - Research, Florham Park, NJ; and Microsoft Research Lab, Cambridge, UK. His research interests include networks and performance evaluation.

He currently serves as Editor-in-Chief of IEEE/ACM Transactions on Networking and on the editorial boards of Journal of the ACM, and IEEE Journal on Selected Areas in Communications, and has previously served on numerous other editorial boards. He was Program Co-chair of the joint ACM SIGMETRICS and PERFORMANCE '92 conference and the Performance 2002 conference. He is a member of ACM and ORSA, and Chair of IFIP Working Group 7.3.

He has received the 2007 IEEE Koji Kobayashi Award, the 2007 ACM SIGMETRICS Achievement Award, the 1998 IEEE Communications Society William Bennett Best Paper Award, and numerous best conference/workshop paper awards. Last, he has been elected Fellow of both the ACM and IEEE.