Detecting 802.11 Wireless Hosts from Remote Passive Observations

Valeria Baiamonte[1], Konstantina Papagiannaki[2], and Gianluca Iannaccone[2]

[1] Politecnico di Torino
[2] Intel Research Cambridge

Abstract. The wide deployment of 802.11 WLANs has led to the coexistence of wired and wireless clients in a network environment. This paper presents a robust technique to detect 802.11 wireless hosts through passive observation of client traffic streams at the edge of the network. It is based on the estimation of entropy of packet interarrival times and on the analysis of variation in the measured entropy values across individual end host connections. With the aim of generating a physical layer "signature" that can be easily extracted from packet traces, we first perform controlled experiments and analyse them through Spectral Analysis and Entropy evaluation. Based on the gained insight we design a methodology for the identification of 802.11 wireless clients and test it on two data sets of packet-level traces collected in different networks. Our results demonstrate that wireless identification is highly precise in the presence of a sufficient traffic sample.

1 Introduction

The proliferation of 802.11 WLANs has led to the emergence of hybrid local area networks where wireless and wired clients can seamlessly access the network's resources. However, network access through 802.11 imposes a different kind of constraints on the network design. Wireless clients are likely to feature increased mobility, whereby the client point of attachment to the network may differ from one point in time to the next, affecting different parts of the infrastructure. Wireless client requirements with respect to throughput and latency may be harder to meet given the unpredictable nature of the wireless medium. Security solutions implemented in the core of the network may need to incorporate knowledge on the type of client to enforce appropriate solutions. Lastly, even the provisioning of services addressed to network clients may significantly benefit from knowledge regarding the capabilities of such clients.

Recent research work has focused on generally characterizing and studying wireless traffic with the aim of highlighting essential differences and analogies with wireline traffic [1,2]. While particular network environments make the identification of wireless and wired clients trivial, say if clients obtain IP addresses from different address blocks, there exist network environments where such a task is much harder to perform [3,4,5]. For such environments previous approaches have advocated the use of active measurements [3] or passive inspection of TCP

I.F. Akyildiz et al. (Eds.): NETWORKING 2007, LNCS 4479, pp. 356–367, 2007.

traffic [4,5]. The fundamental issue with the first approach is that it requires client cooperation for the collection of the appropriate information. The last two approaches, furthermore, precludes the identification of clients that may use protocols other than TCP. In this work we look into the problem of passive identification of wireless clients as observed at a location inside a network where a diverse set of client traffic is aggregated.

We put forth the idea that a primary difference between wired and wireless systems is *uncertainty* which manifests itself at the physical and MAC layers. Uncertainty is naturally quantified using information theoretic measures such as that of entropy [6]. Techniques based on uncertainty estimation have lately been largely applied to traffic analysis. In [7,8,9], entropy of packet arrivals is used to detect traffic anomalies in wired networks.

In this work, we exploit the concept of entropy to measure the "uncertainty" of wireless traffic. Variation of link quality and channel conditions together with the channel access mechanism employed by the 802.11 protocol, "brands" client traffic with a unique signature that can distinguish wireless from wired traffic. We look into the fundamental operations of the 802.11 protocol and the way it can perturb client traffic due to the random access mechanism employed. More practically, we first use active measurements in controlled experiments and deterministic traffic patterns. Our preliminary study draws on spectral analysis applied to packet interarrival times in such controlled environments. The inspection of the spectral density, in the signal built from the packet arrival process, lets us observe how intrinsic periodicity of sourced traffic can be either preserved or distorted depending on the physical medium traversed. Furthermore, spectral analysis allows us to define the signal bandwidth that needs to be filtered to extract information from a traffic flow, i.e. to define the time scales of interest for our purposes.

Correctly filtering the traces, we then perform entropy evaluation of packet interarrival times, observing the differences in the content information of wired and wireless traffic flows. Based on these observations, we design an algorithm that can identify wireless clients through passive observations at an aggregation point. Using two different data sets collected from an enteprise and a campus environment we demonstrate that our algorithm achieves high accuracy in host detection when provided with a sufficient traffic sample.

2 Extracting a Wireless Signature

Our goal is to capture the physical layer "signature" in a traffic flow through the inspection of the packet interarrival times. We exploit two main features that a-priori differentiate wired from wireless access: (*i*) the unpredictability of the wireless medium, and (*ii*) the impact of the 802.11 medium access control (MAC) mechanism.

Our conjecture is that both these mechanisms are bound to introduce random delays in the transmission of packets from a wireless host. Due to the unpredictability of propagation conditions on the shared radio medium, packets sent

inside a wireless network may not be correctly decoded at the receiver. Packet loss causes wireless stations to retransmit, delaying the packet delivery at the receiver. Furthermore, every time a wireless station has a packet to send, it needs to contend for channel access. When many wireless clients co-exist in a WLAN, collisions are highly likely to happen, causing retransmission and increased backoff delays. Consequently, delay and increased jitter in packet reception is one fundamental feature of wireless traffic. If a host were to transmit the same constant bit rate (CBR) flow using the wired and wireless medium, we would expect to see increased variation in interarrival times in the wireless transmission compared to a purely periodic stream if access were wired.

However, even in the case of a purely periodic data source inference of the access technology may still become complicated due to the behavior of the host and the location of the observation point. The transmission of a purely periodic traffic stream from a highly loaded host is unlikely to maintain its periodicity when observed at a remote location. Moreover, network congestion and multiplexing with other traffic may distort the original signature in the traffic.

Such a task becomes even more challenging when the original traffic stream is not periodic. If the application used by the client does not generate periodic traffic (like VoIP for instance) we need to focus our attention to those parts in the traffic stream that relate to back to back packets. If packets are transmitted back to back, we would expect that the wireless medium will be able to distort their time of arrival at the receiver. Particular transport layers are bound to generate such back to back packets, for instance when TCP is transmitting a window of packets. However, if the transport layer protocol itself does not allow the observation of interarrival times of packets that have been generated closely spaced to each other, then the wireless signature will be very hard to recover.

In what follows we use controlled experiments to study the behavior of periodic traffic when transmitted through the wireless medium. This section allows us to identify ways in which we can capture "clean" physical layer signatures that do not suffer from the above limitations. Different wireless scenarios are tested to investigate whether this signature is kept and how it varies across different conditions. The results of these experiments can clarify and better motivate the design of the methodology presented in Section 3.

Experimental scenario. We generate traffic flows from wired and wireless clients inside a private LAN to destinations outside the LAN (several Planet Lab nodes. We use the *tg* traffic generator to send UDP and TCP traffic streams. The UDP streams are at constant bit rate of 1.6 Mbps, with 1000 byte packets sent every 5 ms. The TCP connections consist of the bulk transfer of large files (23Mbytes). All experiments are repeated both when clients attach to the network using a ethernet interface and when they use an 802.11b wireless NIC. All experiments are accompanied by sniffers at all clients, the AP (running HostAP), as well as a collection point at the edge of the private LAN that could represent a typical aggregation point where our algorithms would be deployed.

More specifically, when it comes to wireless experiments we test the following different scenarios that may have a significant impact on the wireless signature captured:

1. **Good:** a topology where one client is placed in a good location and there is no contention (1 wireless client close to the AP, signal level: -43 dBm);
2. **G-Cong:** 3 clients are placed in good locations relative to the AP and contend for channel access (client-AP signal level: -43 dBm);
3. **Bad:** a topology where 1 client is placed in a bad location and there is no channel contention (1 wireless client far from the AP, signal level: -65 dBm);
4. **Bad-Cong:** a topology where 3 clients are placed in a bad location and contend for channel access (client-AP signal level: -65 dBm).

Spectral Analysis

The main purpose of this preliminary analysis is to investigate how the physical layer characteristics influence the explicit (e.g. CBR traffic) or implicit (e.g. TCP window of packets) periodicity of a traffic flow. Spectral analysis is carried out through the Discrete Fourier Transform DFT and specifically applied to the process of packet interarrival times. It aims at estimating the harmonic content of a traffic stream, such as to point out the frequency bandwidth whose power spectral density can be related to the MAC and physical layer behavior. The power spectrum of packet interarrivals not only keeps statistical information of a traffic flow, as a Probability Mass Function (PMF) would do, but it also describes how the "energy content" is spread over the frequency domain and how the "information content" is distributed over the signal bandwidth.

More in detail, the signal to be processed is a discrete sequence defined on the time domain and describes the process of packet arrivals:

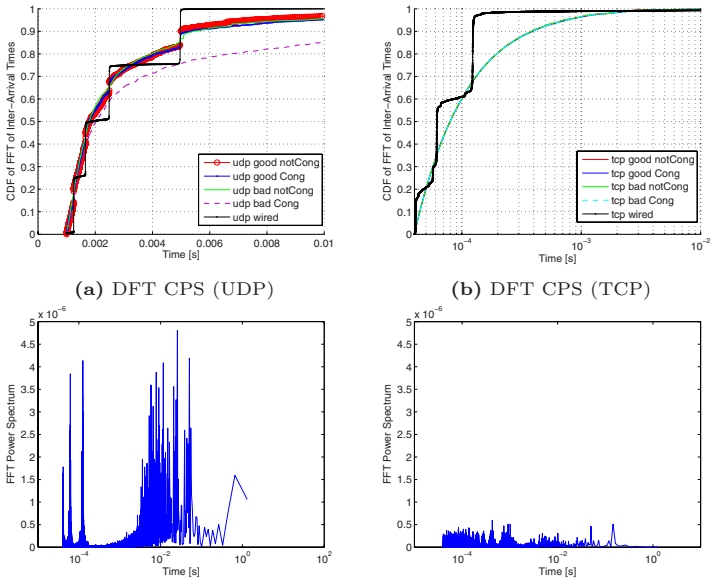
$$x[n] = \begin{cases} 1 \; nT_0 \leq t_{arr} < (n+1)T_0 \\ 0 \; elsewhere \end{cases} \tag{1}$$

$x[n]$ is built as a sequence of pulses of amplitude 0 or 1 spaced by T_0. Pulses of amplitude 1 at nT_0 correspond to packet arrivals occurring at times t_{arr}, with $nT_0 \leq t_{arr} < (n+1)T_0$. The time granularity T_0 has to be precise enough to keep trace of all packet arrivals, with no overlapping of two or more packet arrivals in the discrete time interval $[nT_0, (n+1)T_0]$. To this end, since packet arrivals are never spaced by less than $20\mu s$[1], we have chosen $T_0 = 20\mu s$, thus the resolution on the frequency domain will be $f_0 = 1/T_0$, equal to $50KHz$. The observation window is NT_0 wide, i.e. the total number of observed samples is equal to N.

The Fourier transformed signal, defined on the discrete frequency domain is then:

$$X[kf_0] = \frac{1}{N} \sum_{n=0}^{N-1} x[nT_0] e^{-j2\pi \frac{nk}{N}} \tag{2}$$

[1] Time slot duration in IEEE802.11.



(a) DFT CPS (UDP) **(b)** DFT CPS (TCP)

(c) DFT PS (wired TCP) **(d)** DFT PS (wireless TCP)

Fig. 1. DFT Cumulative Power Spectrum Analysis

Further, before treating $x[n]$ with spectral analysis, the discrete sequence is subtracted of its mean value. The mean value results in a large DC component in the spectrum that does not provide any useful information for our classification study. The DFT power spectrum $P[n]$ is expressed by

$$P[kf_0] = \mid X[kf_0] \mid^2 \tag{3}$$

To compare different power spectra, the *Cumulative Power Spectrum CPS* is used. It is normalized to the total power of the signal as

$$C(n) = \frac{\sum_{k=0}^{n} P[kf_0]}{\sum_{k=0}^{N-1} \mid X[kf_0] \mid^2}, \quad 0 \leq n < N \tag{4}$$

Power Spectrum results presented here are plotted as a function of time, instead of frequency, so as to relate the periodicity content information to the time domain and achieve an intuitive meaning. This is simply obtained rescaling the x-axis.

In Figure 1(a), CPS of UDP wired and wireless CBR flows is computed. Interestingly, the distortion caused by the physical layer is clearly visible and deeply affects the periodicity of CBR flows. Packets have been generated as one every 5ms. If their transmission were not delayed, we would expect the ideal frequency spectrum of a *pulse train* with a pulse every N/T, $N = 1, 2...$ and $T = 5ms$. The CPS, hence, would appear as a step function with a step every T.

Focusing on the "udp wired" curve, indeed, we notice a *step behavior* due to bumps of power corresponding to the main harmonic components at N/T. This confirms that periodicity is actually preserved on the wire, whereas it is significantly reduced for wireless streams. The "wireless good and not congested" case still presents a step behavior, even though the amplitude of bumps decreases, while, when congested and bad located flows are analysed, the periodicity of CBR traffic completely disappears.

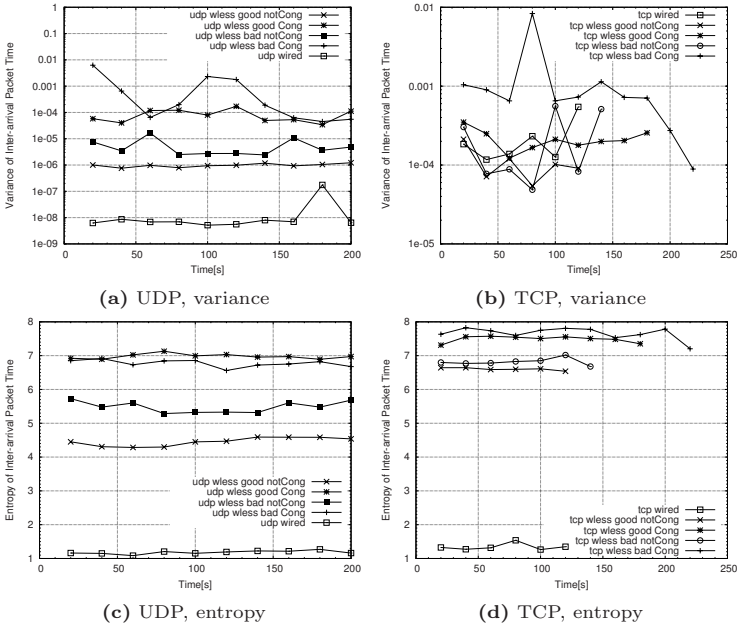
In order to observe a similar behavior for TCP streams, we first need to observe how the signal power is distributed over the frequencies of a TCP signal. For this reason, Figures 1(c) and 1(d) are reported. There, the DFT Power Spectrum of a TCP wireless and a TCP wired connection are respectively compared. In the first, clear high-powered spectrum components are concentrated at the low time scale. These harmonic components refer to those packets sent inside the TCP congestion window, whose transmission is not delayed by the reception of TCP acknowledgements. A large amount of power is then concentrated at the higher time scales and expresses the effect of *Round Trip Times RTTs*. In the second power spectrum, shown in Figure 1(d) the transmission over the wireless medium has the effect of removing most of the periodicity, by reducing the signal power and distributing it over the whole bandwidth. Dominant frequencies are no more discernible.

When the original traffic stream is not periodic, in order to catch the periodicity distortion, we need to focus our analysis on those parts of the traffic streams where packets are sent in a back to back fashion, thus where the access network effects are more visible. In the case of TCP traffic, this means taking into account the packets transmitted within a TCP window: we need to apply a filter and preserve only the high frequency bandwidth (i.e. the low time scales) where these phenomena can be caught. The filter can be easily applied over the sequence of packet interarrivals, dropping all the big interarrival times exceeding a certain threshold, T_{RTT}. In this work T_{RTT} has been choosen to be set to $10ms$, so as to exclude big RTTs but include all the delays introduced by consecutive retransmissions in highly congested wireless networks.

Thanks to this observation, we are able to isolate TCP wired flow from all the TCP wireless ones through the CPS, as in the UDP/CBR case (Figure 1(b)).

Sample Entropy

The DFT methodology, is computationally expensive and is difficult to be incorporated within an automated algorithm. A natural way to measure the "uncertainty' is to use the concept of Entropy. We describe here the concept of empirical entropy, since we deal with empirical distributions. Let the random variable X denote the value of interarrival times, i.e. the time between back to back packets. X is randomly sampled or observed for m times inducing the empirical probability distribution on X, $p(x_i) = m_i/m, x_i \in X$ and m_i is the frequency of times X is observed to take the value x_i. The empirical entropy of interarrival time distribution is then $H(x) := -\sum_{x_i \in X} p(x_i) \log_2 p(x_i)$. Empirical entropy values computed on the same distribution can be different, depending on the



(a) UDP, variance **(b)** TCP, variance

(c) UDP, entropy **(d)** TCP, entropy

Fig. 2. Variance and Entropy of interarrival packet times

time scales considered to discretize the samples. For our computations, a bin size of $100\mu s$ has been chosen to compute the PMF.

We compute the entropy of the PMF of packet interarrivals across time. Interarrivals are filtered as in the case of spectral analysis and each single point is computed acquiring the PMF of interarrival packet times every 20 seconds at the remote monitoring point.

Figure 2 presents variance and entropy for UDP and TCP streams as a function of time. Entropy evaluation, differently from standard metrics like average value, median and variance, provides indeed a faithful estimation of the information content of a set of outcomes. Variance, for instance, is a measure of the average distance of the outcomes from the mean and quantifies how spread a distribution is around its mean value. This is not enough, though, to quantify the amount of observational variety and randomness retained in a set of observations. As a matter of fact, distributions that present low values of variance may assume high entropy values and vice versa. In the case of our experiments, the variance evaluation for UDP experiments, shown in Figure 2(a), is already able to differentiate wired flows from wireless ones. It also consistently increases in presence of congestion and low signal level within wireless environments. Delays and retransmissions result in a higher variability in packet arrivals, that affects the innate periodicity of CBR traffic. This does not hold for TCP flows though, whose variance is reported in Figure 2(b). Values range from 10^{-4} to 10^{-3} without a clear distinction among the different cases.

In Figure 2(c) and Figure 2(d), instead, entropy values are clearly separated for all the cases, no matter which transport protocol is used. The information content is very low (around 1 bit) for the wired interarrivals, while it grows significantly for wireless packet flows. The difference between wired and wireless is still evident even when just one wireless client is transmitting in the WLAN.

3 Detection Algorithm

From the lessons learnt during the controlled experiments, we can define a generic algorithm to classify end hosts based on their access media. The input of our scheme is a packet-level trace collected at a monitoring point. In the first stage, packets are aggregated on the basis of the IP address source. Within each IP-source set, 5-tuple flows are then isolated. Interarrival times between consecutive packets are computed and then filtered: only those, falling below T_{RTT}, with $T_{RTT} = 10ms$, are kept. The algorithm then computes two values: 1) the empirical entropy H_{IP}, evaluated on the whole IP-source aggregated traces; 2) the empirical entropy of the largest (in terms of number of interarrivals) 5-tuple flow of each IP-source trace, $H_{IP,5}$. We then define *Variation of Entropy* the difference $\Delta H = H_{IP} - H_{IP,5}$.

The two values H_{IP} and ΔH are used to classify hosts. The pseudo-code of the proposed methodology is reported below:

```
If H_IP ≤ H_lower
     then the host is wired
else if H_IP ≥ H_upper
     then the host is wireless
else if H_lower < H_IP < H_upper
     if ΔH ≥ ΔH_THR
            then the host is wired
     else if ΔH ≤ ΔH_THR
            then the host is wireless
```

The thresholds are trained using a small set of passive traces and chosen as $H_{lower} = 3.5bits$, $H_{upper} = 5bits$ and $\Delta H_{THR} = 0.5$. Figure 3 shows the PMF of entropy computed over the training dataset and helps explaining the selection of thresholds. The mass of entropy for wireless flows is concentrated for $H_{IP} \geq H_{upper}$ (the vertical dotted line on the right), while wired entropy values have higher probability for $H_{IP} \leq H_{lower}$ (vertical line on the left). The PMF exhibits an overlapping area in the range $[H_{lower}, H_{upper}]$ bits, where the two distributions are superimposed.

For the flows that fall into that region we use the variation of entropy as discriminator. With wireless hosts, the uncertainty measured by $H_{IP,5}$, i.e. on the largest 5-tuple flow, already accounts for the effects introduced by the wireless transmission. As a consequence, adding other smaller 5-tuple flows has a marginal impact on the value of the aggregate entropy resulting in a low ΔH.

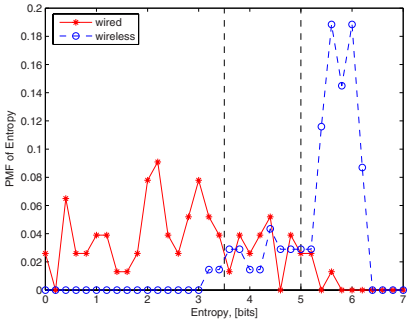


Fig. 3. Probability Mass Function of Entropy

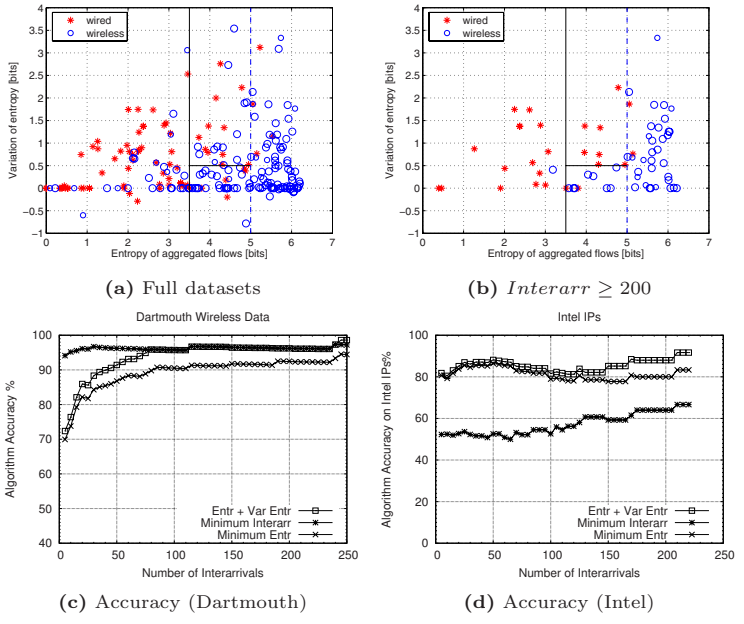
In the case of wired hosts, instead, the variation of entropy is driven by different factors. When H_{IP} is large, ΔH is measuring the impact of aggregating different flows. By adding more outcomes the distribution defined over a limited interval becomes more informative, i.e., the aggregate entropy grows [6].

4 Evaluation

Performance evaluation is carried out on two different datasets, namely Intel and Dartmouth traces. In order to evaluate the algorithm accuracy, results are compared with the ground truth, i.e. we know which IP addresses have been assigned to Ethernet and WLAN hosts form predefined blocks. Intel traces are collected on the access link of the Cambridge laboratory using a CoMo monitoring system [10]. The traffic contains a mix of connections sourced from 93 different hosts, out of which 27 use the wireless LAN. The Dartmouth traces are publicly available and refer to wide area wireless measurements taken at different APs in the university campus of Dartmouth college. We analysed connections from 162 distinct wireless IP addresses.

All results from both the datasets are summarised in Figure 4(a). Hosts with at least two interarrival times are scattered on the basis of the aggregated entropy (x-axis) and variation of entropy (y-axis). Vertical lines at 3.5 and 5 bits delimit the interval $[H_{lower}, H_{upper}]$, while the horizontal line at 0.5bits reports the threshold ΔH_{THR}, used to select on the basis of the variation of entropy.

Most of the IP addresses map unambigously in the wireless or wired regions. A dense cloud of wireless hosts can be noticed where the aggregate entropy is above H_{upper} and the variation of entropy below ΔH_{THR}. Most of the wired hosts, on the other hand, are either spread in the region below H_{lower}, or in the overlapping zone where the variation of entropy is higher. However, several hosts fall in the region where the classification is not certain. This is mainly due to the very small number of observations (i.e. few packets belonging to few connections) that we see in the traces from those end systems. Indeed, if we consider just those IP address for which we can measure at least 200 interarrival



(a) Full datasets **(b)** $Interarr \geq 200$

(c) Accuracy (Dartmouth) **(d)** Accuracy (Intel)

Fig. 4. Performance on Dartmouth and Intel datasets

times, the accuracy improves significantly. Figure 4(b) shows the scatter plot for those end systems.

We have also compared the accuracy of our method with the accuracy of alternative (and simpler) approaches. Figures 4(c) and 4(d) compare the accuracy of three methods as a function of the number of observed packets per source.

The curve labelled as "Entr + Var Entr" refers to our proposal. "Minimum Interarr" refers to an approach that just looks at the minimum interarrival between all packets from the same source. This approach is based on the fact that on the wireless media two back-to-back packets still need to contend for the medium. Therefore, if the interarrival is above $100\mu s$ (i.e., the minimum interarrival of two 802.11a/g packets with no payload) the source is considered a wireless host.

Interestingly, the minimum interarrival method performs very well on the Dartmouth dataset (Figure 4c) while it performs poorly on the Intel dataset (Figure 4d). The reason behind this is that some packets are queued on the wired interface right before the monitoring point leading to very small interarrival times. By picking always the minimum interarrival, this method leads therefore to large errors.

The curve "Minimum Entr" computes the minimum Entropy among all 5-tuple flows of a given source IP address. An Entropy below H_{lower} indicates that the source is wired. This approach always underperforms our proposed mechanism even when a large number of samples is considered.

5 Related Works

Recent research work has focused on characterizing and studying wireless traffic with the aim of highlighting essential differences and analogies with wireline traffic [1] [2] [11]. The results from these studies were found informative to develop the proposed detection scheme.

Studies closer to that presented in this paper are in [3,4,5]. In the first, the authors propose an access network type classification scheme, where a host interested in determining the connection type of a remote user, requests to send a known sequence of packets. Interarrival times of received packets are then recorded. At this stage, the host is able to infer the user's connection type based on the median and entropy of the sequence of packet interarrival times. The scheme proves to be strongly reliable, but, unlike our solution, it requires the remote user cooperation.

In the second, the classification scheme is also strongly related to our solution. A Bayesian inference algorithm is developed to estimate and classify whether TCP flows have traversed an 802.11 WLAN. The methodology relies on the time intervals occurring between TCP-ACK pairs and infers the connection type of a user based on this observations. Unlike the previous classification scheme, detection is performed through passive measurements, but the identification is possible only for clients that are employing the TCP protocol.

The output of this algorithm is the fraction of wireless flows with a certain degree of belief. The error, estimated as the difference between the inferred fraction of wireless flows and the actual one, is bounded within +/- 0.05. The output of our algorithm, instead, extends this information by providing the identification of each single host within a certain traffic trace. The accuracy error, computed as the number of mistaken detections over all considered hosts, reaches 1% and 6% in the two data sets.

Moreover, the methodology in [4] makes an inefficient use of traces because the detection is carried out only using TCP ACK-pairs with an inter-ACK time of 400us. (Due to TCP self clocking, wireless flows show a low number of such ACK-pairs, thus often 90% of wireless flows contain less than 10 ACK-pairs.) Differently our algorithm converges faster because it considers inter-packet pairs belonging to both UDP and TCP traffic connections whose time distance is below 10ms.

The TCP-ACK pairs technique is also employed in [5]. In this work, some of the aforementioned limitations, e.g. the efficiency in the trace analysis, the promptness and accuracy of results, are overcome by using two effective sequential hypothesis tests, with and without training data, respectively. However, the detection is still limited to hosts sourcing TCP connections only.

Valid references to information-theoretic techniques applied to traffic analysis can be found in [7], [8], [9], where entropy of packet arrivals is used to detect traffic anomalies in wired networks. Research work in [12] discusses the use of information theory and uncertainty to characterize wireless networks. Finally, Spectral Analysis of traffic flows is applied in [13], with the purpose of detecting and classifying denial of service attacks.

6 Conclusion

We have presented a method to classify wired and wireless hosts only based on passive traffic observations at a remote location. Our method does not require any cooperation from the end-systems and is protocol agnostic. The accuracy is comparable with previously proposed approaches and outperforms naïve methods. Our future work is focused on testing the methodologies over larger datasets in an attempt to isolate the specific sources of errors in our classification.

References

1. Hernandez-Campos, F., et al.: Assessing the real impact of 802.11 WLANs: A large scale comparison of wired and wireless traffic. In: LANMAN. (September 2005)
2. Balachandran, A., et al.: Characterizing user behavior and network performance in a public wireless LAN. ACM PER **30**(1) (2002) 195–205
3. Wei, W., et al.: Classification of access network types: LAN, wireless LAN, ADSL, cable or dialup? In: Proceedings of IEEE Infocom. (March 2005)
4. Wei, W., et al.: Identifying 802.11 traffic from passive measurements using iterative bayesian inference. In: Proceedings of IEEE Infocom. (April 2006)
5. Wei, W., et al.: Passive online rougue access point detection using sequential hypothesis testing with tcp ack-pairs. Technical report, University of Massachussets Computer Science (November 2006)
6. Cover, T., Thomas, J.: Elements of Information Theory. John Wiley (1991)
7. Adya, A., et al.: Architecture and techniques for diagnosing faults in IEEE 802.11 infrastructure networks. In: Proceedings of ACM Mobicom. (September 2004)
8. Lakhina, A., et al.: Mining anomalies using traffic feauture distributions. In: Proceedings of ACM Sigcomm. (August 2005)
9. Xu, K., et al.: Profiling Internet backbone traffic: Behavior models and applications. In: Proceedings of ACM Sigcomm. (August 2005)
10. Iannaccone, G.: Fast prototyping of network data mining applications. In: Proc. of PAM. (March 2006)
11. Ridoux, J., Nucci, A., Veitch, D.: Seeing the difference in IP traffic: Wireless versus wireline. In: Proceedings of IEEE Infocom. (April 2006)
12. Das, S., Rose, C.: Coping with uncertainty in mobile wireless networks. In: PIMRC. (September 2004)
13. Hussein, A., Heidemannan, J., Papadopoulos, C.: A framework for classifying denial of service attacks. In: IEEE Globecom. (December 2004)