Computer Networks 56 (2012) 1968–1982

Contents lists available at SciVerse ScienceDirect

Computer Networks

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

Traffic sensing and characterization in multi-channel wireless networks for cognitive networking

Bheemarjuna Reddy Tamma [a,*], B.S. Manoj [b], Ramesh R. Rao [c]

[a] Indian Institute of Technology Hyderabad, India
[b] Indian Institute of Space Science and Technology, Trivandrum, India
[c] California Institute for Telecommunications and Information Technology – UC, San Diego, USA

ARTICLE INFO

Article history:
Received 16 December 2009
Received in revised form 11 October 2011
Accepted 2 December 2011
Available online 10 December 2011

Keywords:
Packet sampling
Traffic characterization
Multi-channel wireless networks
Traffic sensor
Cognitive networking

ABSTRACT

Traffic sensing and characterization is an important building block of cognitive networking systems, however, it is very challenging to perform traffic characterization in multi-channel multi-radio wireless networks. Due to the presence of network traffic in multiple channels, the existing count-based packet sampling methods demand continuous capture on each channel to be effective; this requires a dedicated wireless interface per channel, and hence the existing sampling methods require a very expensive infrastructure and have poor scalability. Time-based sampling methods, on the other hand, offer a cost-effective and scalable solution by reducing the amount and cost of the resources necessary to monitor and characterize the wireless spectrum.

The contributions of this paper include the following: (i) a discussion of packet sampling techniques for traffic sensing in multi-channel wireless networks, (ii) a comparison of various time-based sampling strategies using the Kullback–Leibler divergence (KLD) measure, (iii) a study on the effect of the sampling parameters on the accuracy of the sampling strategies, (iv) development of sampling accuracy graphs for easing the process of best sampling scheme selection in multi-channel wireless networks, (v) the proposal of a new metric (traffic intensity) which estimates the busyness of channels by taking into consideration not only the successfully received packets but also corrupt or broken packets, (vi) implementation of time-based sampling in a prototype traffic sensor device for multi-channel traffic sensing in IEEE 802.11 b/g networks, and (vii) characterization of a campus IEEE 802.11 network environment in a spatio-temporal–spectral fashion using sampled traffic traces collected by traffic sensors.

© 2011 Elsevier B.V. All rights reserved.

1. Introduction

Cognitive networking involves developing wireless systems that will have much deeper awareness about their own operations and the network environment, learn relationships among network parameters, network protocols, and the network environment, plan and make decisions in order to achieve local, end-to-end, and network-wide performance, as well as resource management, goals. In a cognitive network paradigm, all network elements track the spatial, temporal, and spectral dynamics of their own behavior and the behavior associated with the environment, and report that information to a cognitive controller. The information so gathered is used by the cognitive controller to learn, plan and act in a way that meets network or application requirements [1,2]. Cognitive networking differs from cognitive radios or cognitive radio networking [3,4] in that the latter two apply cognition only to the physical layer to dynamically detect and use *whitespaces* in the spectrum, focusing strictly on dynamic spectrum

* Corresponding author. Tel.: +91 40 2301 7001; fax: +91 40 2301 6032.
E-mail addresses: tbr@iith.ac.in, bheemarjun@gmail.com (B.R. Tamma), bsmanoj@iist.ac.in (B.S. Manoj), rrao@ucsd.edu (R.R. Rao).

1389-1286/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.comnet.2011.12.002

access, whereas the objective of cognitive networks is to apply cognition to all layers of the network protocol stack for achieving network-wide performance goals.

A cognitive controller acts as the brain of a cognitive networking system. The cognitive controller has to gather, compact, and analyze large amounts of wireless network traffic across several dimensions such as space, time, and frequency in order to characterize the wireless environment and the state of the network. However, in most of the existing studies on the characterization of wireless LANs, measurements have been conducted on the wired portion of the network using wired monitoring hardware and/or combined with SNMP statistics collected from APs [5]. While it is easier to make consistent measurements in the wired portion of the network, such measurements cannot monitor the actual IEEE 802.11 frames spread across multiple channels and hence fail to observe the significant vagaries present in wireless spectrum. In [6,7] the authors proposed wireless monitoring systems for enterprise IEEE 802.11 b/g networks. The proposed monitoring system consists of a set of wireless traffic sensors, each one equipped with one or more wireless network interface cards (NICs) that are configured on one of the orthogonal channels available in the 2.4 GHz ISM band in order to monitor all the traffic in those channels; traces obtained from multiple sensors are then merged to construct a global view of network activity, from which one can extract characteristics like application workloads, user session durations, user mobility patterns, and temporal network usage patterns. The cognitive networking paradigm, however, requires a characterization of the wireless network traffic present in each of the channels in a spatio-temporal fashion. Such characterization helps the network to tailor its optimization criteria to different network elements present in the system. To address this issue, we propose *not to merge* the network traffic traces collected from different channels and locations with the help of traffic sensors, and store them separately across three dimensions: space (location of traffic sensor), time, and spectrum (channel).

1.1. Challenges in multi-channel wireless network traffic sensing and characterization

The traffic in wireless LANs is spread across a number of channels, whereas a single wireless NIC can typically monitor only one channel at a time. The 802.11 b/g-based wireless networks operate on the 2.4 GHz ISM band and have 11 (13) channels in USA (Europe) to choose from.[1] Even though orthogonal channels are typically used for configuring APs, in some scenarios (*e.g.,* non-802.11 sources such as Bluetooth, microwave ovens, and noise rendering an orthogonal channel useless) other channels are also being used in the configuration of APs. In some scenarios, APs belonging to different wireless LANs co-exist on the same channel and compete for radio resources in the same geographic region. To characterize traffic in such network environments, the wireless monitoring system should have the capability of monitoring all wireless channels in a spatio-temporal fashion. In order to characterize wireless traffic on all channels, the traffic sensor has either to have a dedicated wireless NIC per channel in order to measure all the traffic in that channel (1:1 sampling ratio), or to switch a single wireless NIC across all the channels in a round-robin fashion, thus measuring a fraction of traffic on each channel (*i.e.*, 1:11 sampling ratio as we have 11 channels in 2.4 GHz ISM band for Wi-Fi operation). In the first case, the sensor device would be very complex due to the large number of wireless NICs being used, and moreover the transmission, storage, and analysis of the captured packets from a large number of channels simultaneously could be problematic. Therefore, such a multi-interface complete capture solution would be very expensive and would scale poorly to large scale multi-channel wireless network environments. The multi-channel traffic sampling scheme is, therefore, essential. The data reduction resulting from sampling not only reduces the bandwidth consumed in transmitting the measurement data to the collection point (*i.e.,* a cognitive controller with a central database repository) but also decreases the cost incurred for analysis and storage of the data. The deployment of sampling methods aims at estimating some specific characteristics of the parent population (*i.e.,* the complete network traffic trace) at a lower cost than a complete census would demand. Sampling trades off the opposing goals of controlling estimation accuracy and measurement costs.

Unlike wired networks, the wireless environment has a number of unique characteristics that would affect the accuracy of the traffic sampling strategy. Due to the presence of aggregate traffic in the local wireless environment, the traffic characteristics do not follow well known statistical distributions such as Poisson or exponential. In addition, the presence of broken or corrupt packets can contribute to the consumption of channel resources exactly similar to a successful packet. The inability of the monitoring device to take into account such packets can lead to erroneous characterization. Above all, unlike in wired networks, the wireless environment offers multiple transmission rates. Therefore, the channel resources consumed by a 1 Mbps packet is several times more than the channel resources consumed by the same length packet transmitted at 11 Mbps. Unlike the wired network traffic characterization, the wireless traffic characterization can be affected by unknown interference sources as well. For example, the presence of Bluetooth traffic or microwave traffic can potentially affect the IEEE 802.11 b/g network traffic, thereby resulting in an erroneous characterization of WLAN traffic.

The organization of the rest of the paper is as follows: In Section 2, we give a brief overview of packet sampling schemes and then present the metric of importance for studying accuracy of sampling schemes. Section 3 reviews existing works in network traffic sensing and characterization. In Section 4, we compare and study the accuracy of sampling schemes by varying sampling parameters. Section 5 outlines the design and implementation of the

[1] In the rest of this paper, we consider the 2.4 GHz ISM band in USA and, therefore, discuss in terms of 11 channels permitted for use in Wi-Fi networks. However, this work can be easily extended to 13 channels available in the 2.4 GHz ISM band in other regions for Wi-Fi network operation.

traffic sensor device, and presents the monitoring testbed infrastructure that we use for traffic characterization in our campus IEEE 802.11 wireless network environment. Section 6 briefs the historical traffic statistics that we derived from the sampled traffic traces of the campus wireless environment, and finally Section 7 contains concluding remarks.

2. Packet sampling

As a scalable means of monitoring network traffic, packet-level traffic sampling has attracted much attention from the industrial and research communities. Sampling is a form of passive traffic measurement, in which not all packets are measured, but only a fraction of them, which are selected based on the sampling method and parameters associated with the sampling process. For instance, count-based systematic sampling method such as "1 out of N packets" represents a popular sampling design employed in Cisco and Juniper routers.

Sampling methods can be characterized by the sampling algorithm, which describes the basic process for the selection of packets from each sampling interval, and the type of trigger used to start the packet capture. Based on the sampling algorithm, there are three main classes of sampling methods: simple random sampling, systematic sampling, and stratified random sampling [13]. As shown in Table 1, for each class, one can use either packet counts or timers to trigger the selection of packets for inclusion in a sample. In the case of simple random sampling, packets are selected randomly from the parent population by treating the whole trace as a single sampling interval. In the other two methods, the whole population trace is divided into several small sampling intervals and some packets are chosen from each such interval. In systematic sampling, packets are selected deterministically from each sampling interval, whereas stratified random sampling involves selecting packets randomly from each sampling interval.

Based on the type of trigger used to start the packet capture, sampling methods can be broadly classified into count-based and time-based sampling schemes. In count-based sampling, the packet count triggers the start of a sampling interval. Here, the sampling period (SP) is defined in terms of the number of packets. The sampling duration (SD) or length is defined as the number of packets selected for inclusion in the sample from each sampling interval. In time-based sampling, a timer triggers the start of a sampling interval or sampling period. Hence, SP is defined as a time interval. However, SD can either be timer-driven or count-driven for time-based sampling schemes. Hence, time-based sampling schemes can be further classified into timer-driven time-based sampling and count-driven time-based sampling schemes.

Fig. 1 shows an example of simple random sampling with SD = k pkts and SP = N pkts, where N is the total number of packets in the complete packet trace. Each pattern filled circle represents one packet selected by the sampling scheme from the population trace. Since this kind of sampling requires availability of the complete packet trace in advance, it is useful for only off-line studies and hence not a candidate for implementing in real-time traffic sensor systems. An example of count-driven count-based sampling is to select two packets out of every five packets in the packet stream. Fig. 2(a) and (b) show systematic and stratified random versions of this sampling (namely SCC and SRCC), respectively. Unlike in SCC, two packets are selected randomly from the pool of five packets in each instance of sampling period in the SRCC scheme. An example of count-driven time-based sampling is to sample one packet every 1 s. Fig. 3(a) and (b) show systematic and stratified random versions of this kind of sampling (namely SCT and SRCT), respectively. It is to be noted that different SPs may have different number of packets available due to burstiness in network traffic. Sometimes it may even become impractical to realize count-driven time-based sampling schemes for certain combinations of sampling parameters, for example, when the number of packets available in SP is less than the number of packets to be sampled as per a given value of SD. This is a problem because one parameter is defined in terms of packets (SD) while other one is defined as a time interval (SP). However, this kind of problem will not arise in case of timer-driven time-based sampling, an example of which is to capture whatever packets that arrive in the first 100 ms of every 1 s. Fig. 4(a) and (b) show systematic and stratified random versions of this sampling (namely STT and SRTT), respectively.

2.1. Why not count-driven count-based sampling for multi-channel wireless traffic sensing?

We now answer the question of why we cannot employ count-based sampling schemes like SCC and SRCC, very popular in the (wired) Internet domain for traffic sensing and characterization, for the multi-channel wireless network domain. One of the main reasons is that there are multiple channels to be monitored and we want to sample

Table 1
Various packet sampling schemes.

Sampling scheme	Sampling algorithm	Trigger type (sampling duration, sampling period)	Example
SR	Simple random	Count-driven count-based (only one interval)	k pkts randomly out of *ALL* pkts
SCC	Systematic	Count-driven count-based	k pkts out of every N pkts
SRCC	Stratified random	Count-driven count-based	k pkts randomly out of every N pkts
SCT	Systematic	Count-driven time-based	k pkts in every 11 s
SRCT	Stratified random	Count-driven time-based	k pkts randomly in every 11 s
STT	Systematic	Timer-driven time-based	t s in every 11 s
SRTT	Stratified random	Timer-driven time-based	t s randomly in every 11 s

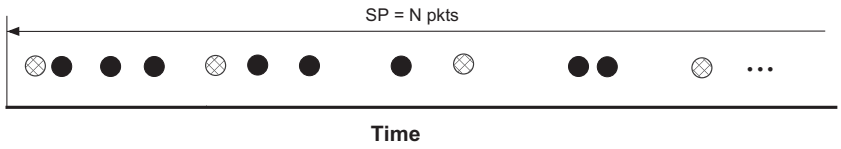


Fig. 1. Simple random sampling with SD = k pkts and SP = N pkts. Each pattern filled circle represents one sampled packet chosen from the population trace by the sampling scheme.

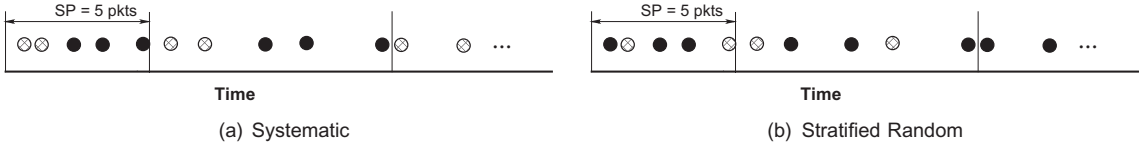


Fig. 2. Count-driven count-based sampling schemes with SD = 2 pkts and SP = 5 pkts.

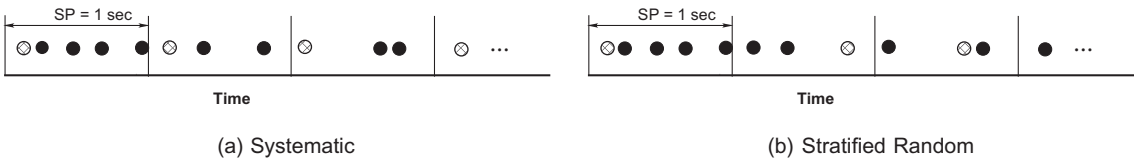


Fig. 3. Count-driven time-based sampling schemes with SD = 1 pkt and SP = 1 s.

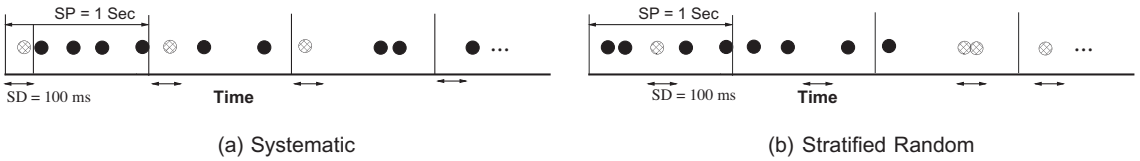


Fig. 4. Timer-driven time-based sampling schemes with SD = 100 ms and SP = 1 s.

traffic from these channels with just one wireless NIC to reduce the cost and complexity of the traffic sensing process. The other reasons are that traffic characteristics like inter arrival times (IATs), packet sizes, and data rates vary *spectrally*, *temporally*, and *spatially*, and the channel switch time (CST, the time needed for tuning a wireless NIC from one channel to another) is not negligible. Although hardware specifications of several manufacturers indicate CST in the order of 200 μs, actual times may be higher as a result of driver code paths and delays in waiting for any pending receive dynamic memory access (DMA) operations on the old channel to finish [8]. A conservative CST estimate is between 1 ms and 5 ms for Orinoco, Atheros, and Intersil wireless chipsets [9].

We now illustrate the problem in employing count-based sampling for multi-channel traffic sensing by considering a hypothetical scenario in which traffic characteristics *do not vary* spectrally and temporally. More specifically, we assume that IAT is constant and remains the same for all channels that we want to monitor. Let us also assume that packet transmission times are negligible, CST = 5 ms, IAT = 625 μs, SD is one packet, and there are only three channels to sample as shown in Fig. 5. Since CST is greater than IAT, SP that can be realized in this hypothetical scenario is given by

$$\mathrm{SP} \geqslant \frac{\mathrm{CST} \times \mathrm{Number\ of\ Channels}}{\mathrm{IAT}}. \quad (1)$$

Hence SP should be at least 24 packets and we can implement "1 out of 24 packets" SCC sampling for monitoring a three-channel wireless system. For implementing the "1 out of 11 packets" SCC sampling for monitoring an IEEE 802.11 b/g system with 11 channels, according to Eq. (1), it requires IAT $\geqslant$CST (5 ms) which is not realistic in high traffic environments. Further, packet transmission times are not negligible and they affect IATs in real wireless systems. The IATs also vary widely across channels. In any given channel, IATs are not constant and, instead, vary widely on per packet basis from a few μs to several seconds due to burstiness of network traffic. Lack of constant IAT on a channel means we do not know in advance when to switch back to that channel for collecting the next sample. In such chaotic real wireless systems, Eq. (1) does not hold any more and it is impractical to realize any count-based sampling scheme even when CST is zero. Hence, in practical multi-channel wireless network scenarios, count-based sampling schemes are not realizable with a single wireless NIC.

We conclude that implementing a count-based traffic sampling method is very expensive in a multi-channel

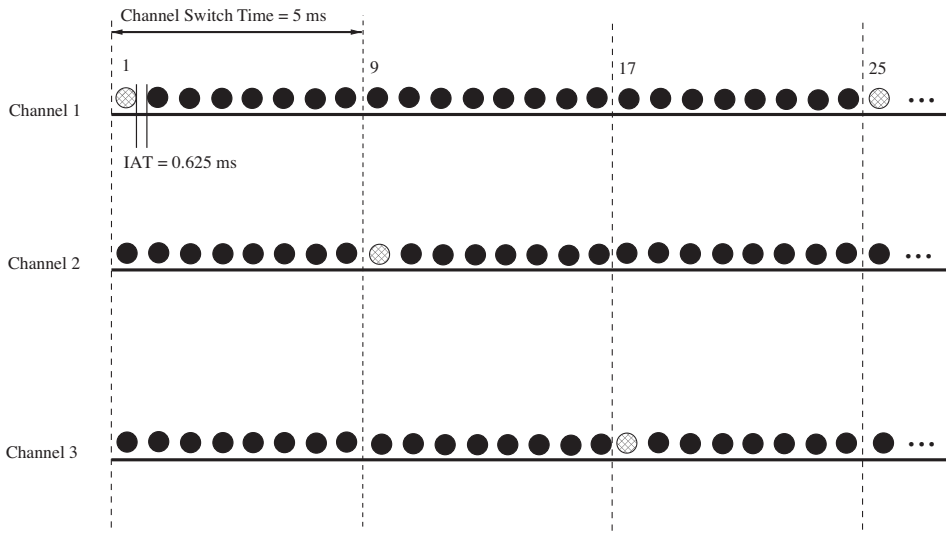


Fig. 5. Systematic count-driven count-based sampling with SD = 1 pkt and SP = 24 pkts.

wireless network environment, as it requires one dedicated wireless NIC to sense each channel. The time-based sampling methods seem to offer a cost-effective and scalable solution by reducing the cost of resources necessary to accurately characterize the wireless traffic. For example, the use of timer-driven time-based sampling enables us to make use of a single wireless NIC to sample multiple channels in a round-robin fashion. However, in order to achieve a high sampling accuracy, we need to determine the relationship between parameters SD and SP, and identify the right set of values for these parameters for time-based sampling.

In order to reduce the implementation cost of wireless traffic characterization by using time-based sampling schemes, we need to reduce the value of SD and maximize the value of SP without affecting the accuracy of characterization. If we have only one wireless interface for sampling traffic in IEEE 802.11 b/g networks, which have 11 channels for traffic characterization, SP should be at least $11 \times \mathrm{SD}$. Otherwise, it is impractical to perform multi-channel sampling with just one interface card in IEEE 802.11 b/g networks. Hence, ideally SD and SP should maintain a 1:11 ratio.

2.2. Accuracy of sampling

We can characterize a sampling method with the following four parameters: sampling algorithm, trigger type, sampling period, and sampling duration. The values of these parameters should be chosen based on the accuracy requirements and the sampling overhead, as well as the characteristics of the parent population and the traffic metrics being measured. In this work, we take four traffic metrics and compare the sampling methods based on how correctly the information can be extracted from the sampled traces. Metrics of interest are: packet size distribution, packet PHY data rate distribution, packet received signal strength indicator (RSSI) distribution, and packet inter-arrival time (IAT) distribution. We implemented four time-based sampling methods (the bottom four in Table 1) at different granularities by varying sampling duration and sampling period. Our goal is to study the effect of sampling parameters on the integrity of the resulting samples.

To measure and compare the performances of various sampling methods, we need a metric to measure how close the distributions given by the sampling methods are, compared to the actual distributions of the parent populations. We now describe one such statistical measure that quantifies the distance, or relative entropy, between two probability distributions.

Let $p(x)$ denote the probability mass function for the distribution of the parent population packet trace. In this work, we studied characteristics of four different population packet trace distributions: packet size distribution, packet PHY data rate distribution, packet received signal strength indicator (RSSI) distribution, and packet inter-arrival time (IAT) distribution. Therefore, random variable x refers to either packet size, PHY data rate, RSSI, or IAT. For example, for the distribution of packet sizes, the term $p(i_1 \leqslant x < i_2)$ gives the probability of packet size lying in the interval, $[i_1, i_2)$. The probability mass function defines the probability that the traffic metric under study (*e.g.*, packet size and RSSI) lies in certain intervals. Let $s(x)$ denote the probability mass function for the sampled packet trace obtained by applying one of the sampling methods. The normalized Kullback–Leibler divergence (NKLD) [14] is defined as

$$NKLD(p(x)\|s(x)) = \frac{KLD(p(x)\|s(x))}{H(p(X))}, \tag{2}$$

where

$$KLD(p(x)\|s(x)) = \sum_{x\in\chi} p(x) log \frac{p(x)}{s(x)},$$

and $$H(p(X)) = \sum_{x\in\chi} p(x) log \frac{1}{p(x)}.$$

$H(p(X))$ is the entropy of the random variable X with distribution $p(X)$. The KLD is zero when the distributions are identical and strictly positive otherwise. The calculation of NKLD requires the selection of a set of intervals (χ) for the traffic metric distribution under investigation. In our experiments, for each traffic metric distribution we choose χ in such a way that each interval contains at least 1% of data from the population data set. For example, in case of packet size distribution we start from the lowest packet size value found in the population data set and increase the value until we get at least 1% of data to obtain the boundary of the first interval. Then beginning with the boundary of the first interval, we repeat the above procedure to obtain the boundary of the second interval, and so on. Once intervals are known, it is straightforward to obtain the probability mass functions for the population packet trace and sampled packet traces.

3. Related work

In literature, several studies addressed the issue of characterizing wireless LANs (see [5–7] and the references in them). Most of them, however, relied upon wired monitoring and/or the use of SNMP statistics, and hence did not characterize the wireless spectrum in a spatio-temporal fashion. Some work which employed wireless traffic sensors was focused on the merging of traces collected from all channels and locations for the characterization of global network activity [5–7]. However, the cognitive networking paradigm requires the characterization of the traffic on all channels in a spatio-temporal fashion. Hence the wireless monitoring system should have the capability to sample a fraction of the traffic from all wireless channels.

On the characterization of data network traffic, Claffy et al. [10] presented a detailed study of the performance of various sampling methods for wide area network traffic. Their work focused on the sampling accuracy of time and count-based methods for both random and systematic periods of sampling. However, they only studied count-driven time-based sampling. Their result mainly pointed to the inappropriateness of using the count-driven time-based techniques because they do not perform as well as the count-based ones. This is due to traffic patterns, *i.e.,* it is well established that Internet traffic is bursty over a range of time scales. Consequently there can be inhomogeneous bursts of many packets with small inter-arrival times. Time-based sampling methods more easily miss these than a count-based method, and estimators built on them have higher variance. However, as explained in the previous section, implementing a count-based sampling method is very expensive in a multi-channel wireless network environment.

In [11], the authors proposed two methods for channel-based sampling in IEEE 802.11 b/g networks: the first method is a timer-driven time-based sampling with fixed sampling parameters, while the second method adaptively varied the sampling duration as a function of the packet arrivals seen on each channel. However, the effect of sampling parameters on the accuracy of sampling, which is one of the most important aspects, was not reported in their work. Time-based sampling methods are not studied thoroughly in the literature. Most of the conclusions derived on the accuracy of these methods are based on count-driven time-based sampling with some arbitrarily fixed sampling parameters [10].

In this work, we first study the accuracy of various time-based sampling schemes by varying the sampling periods and sampling durations for traffic sensing in multi-channel wireless networks. Then we outline the design and implementation of a traffic sensor system for multi-channel wireless networks. Finally, we present some results on the characterization of a campus IEEE 802.11 network environment in order to show the effectiveness of the traffic sensor system. In [12], we conducted a preliminary study on traffic sampling in wireless networks. In this paper, we thoroughly extend that work by (i) presenting a tutorial on packet sampling schemes, (ii) studying sampling accuracy measures for several important network parameters such as packet inter arrival times (IATs) and packet PHY data rates, (iii) providing sampling accuracy graphs for helping network engineers to choose the best sampling scheme and sampling parameters for characterizing IEEE 802.11 networks, (iv) outlining the design and implementation of a prototype system for performing traffic sensing activity, (v) providing historical traffic statistics of our campus IEEE 802.11b/g network based on samples collected through a monitoring testbed, and (vi) attributing a meaning to NKLD values by also measuring relative error (RE) between sample traces and the population trace.

4. Comparison of sampling schemes

In order to get samples at different granularities for various time-based sampling schemes and measure their sampling accuracy using the NKLD measure, we need population (*i.e.,* complete) packet traces. Therefore, we configured two of the traffic sensor nodes[2] to do continuous packet capture on one particular orthogonal channel for four weeks. We treat these traces as our parent population data sets for that channel and generate different sample traces for that channel by varying sampling duration and sampling period for various time-based sampling schemes. Complete population traces from multiple channels are not necessary for our study because in cognitive networking we are mainly interested in *channel wise* traffic characterization, not aggregate traffic behavior from all channels. The packet sampling accuracy is not directly related to the presence of multiple channels in the network; instead it depends on the sampling algorithm, sampling duration, and sampling period. Hence, we studied the effect of these parameters on the accuracy of packet sampling by considering the complete population trace from one particular channel. We also repeated experiments by varying channel number, but excluded those performance results from the paper as we observed similar trends in the performance. In this study our target per-packet traffic distributions are packet sizes, IATs, packet PHY data rates, and RSSIs. In the graphs all performance results are shown with 95% confidence intervals.

[2] We outline the design and implementation of the traffic sensor node in the next section.

Fig. 6(a) shows the variation of the NKLD measure for various samples of STT and SRTT sampling schemes for the distribution of packet sizes. In this experiment, we kept SP constant at 11 s and varied SD from 100 ms to 11 s to obtain various samples from the parent population packet size trace. As SD increases (*i.e.,* the sample size increases) the values of NKLD gradually decrease. Overall, we see a small divergence between the population packet size trace and sampled packet size traces of STT and SRTT schemes. In addition, when the SD goes beyond 1 s, the variation in the NKLD decreases. In other words, even samples generated with an SD of 1 s are closely matching the distribution of the parent population. From the figures we can also observe that the performance of the STT and the SRTT sampling schemes is almost identical. However, as SRTT requires the generation of pseudo-random numbers, it is more expensive to implement it in real traffic sensor systems.

Fig. 6(b) shows the NKLD measures of count-driven time-based schemes, SCT and SRCT. For these sampling schemes, SD is defined as the number of packets to capture for inclusion in the sample during SP. As shown in the figure, the performance of SCT and SRCT schemes are almost identical. However, when we compare these schemes against the timer-driven time-based schemes (STT and SRTT), the NKLD measures are significantly lower for the timer-driven time-based schemes than for the count-driven time-based sampling schemes (SCT and SRCT). This is because in count-driven time-based schemes the sample size is fixed (as SD is defined in terms of packet count) and does not grow proportionally with the traffic load in the network, while in the case of the STT and SRTT schemes sample size grows proportionally to the traffic load, as SD is defined as a time interval. From the above results, we conclude that the STT sampling scheme is the best sampling strategy in terms of sampling accuracy and ease of implementation in real systems. We now employ STT and compute NKLD measures for samples obtained from the population of packet PHY data rates, IATs, and packet RSSI traces. Fig. 7(a)–(c) show their corresponding NKLD scores. Comparing these figures with Fig. 6(a) we can observe that the inter-arrival time distribution has the lowest NKLD values. Out of the four traffic metrics studied, the samples of packet PHY data rate distribution exhibit higher divergence from the population packet trace. This could be due to fewer intervals ($|\chi|$, refer Section 2.2 for more details) as packet PHY data rate can take only a fixed number of pre-determined values for IEEE 802.11 b/g networks. Since the maximum divergence is still very small and negligible, we can configure the traffic sensor nodes with STT sampling to accurately sense and characterize all the traffic metrics in multi-channel wireless network environments.

4.1. Effect of the sampling period

In this experiment, we study the effect of sampling period on the sampling accuracy for various packet distributions. We kept SD constant at 1 s and computed the NKLD from the samples obtained using the STT sampling scheme by varying SP from 1 s to 32 s. Fig. 8 shows NKLD values for the distributions of packet RSSIs, packet sizes, packet IATs, and packet PHY data rates. As expected, for all the traffic metrics the NKLD increases when SP increases due to decrease in sample size. We can also observe that the packet inter-arrival time distribution has the lowest divergence values. We also conducted similar experiments by varying channel number for four weeks in our campus; the results are very similar to what we presented in this paper, and have therefore been omitted.

4.2. Sampling accuracy graph for IEEE 802.11 b/g networks

In order to reduce the implementation cost of wireless traffic characterization, we would like to reduce the value of SD (defined in terms of time units) and maximize the value of SP (defined in terms of time units) without sacrificing sampling accuracy. If we have only one wireless interface for sampling traffic in an IEEE 802.11 b/g network, which has 11 channels for traffic characterization,

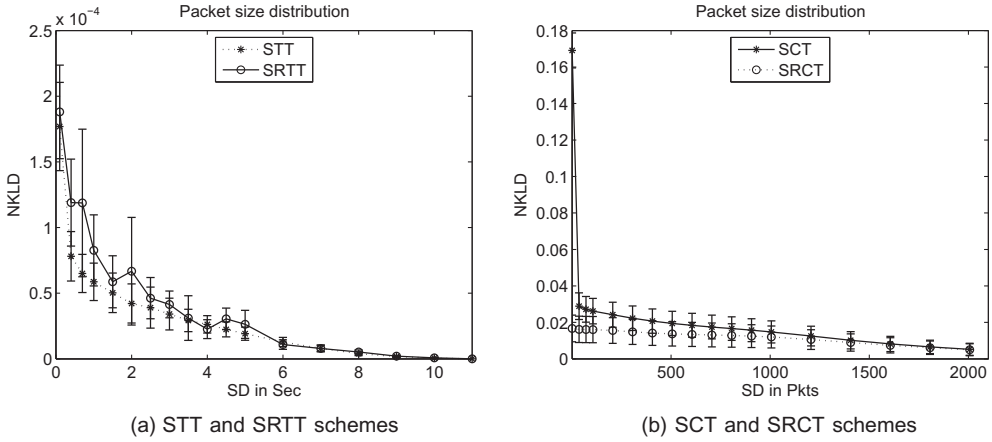


Fig. 6. NKLD vs. SD for packet size distribution for different time-based sampling schemes. Note that *y*-axis ranges of (a) and (b) are different and NKLD scores of time-driven time-based schemes (STT and SRTT) are better by two orders of magnitude compared to count-driven time-based schemes (SCT and SRCT).

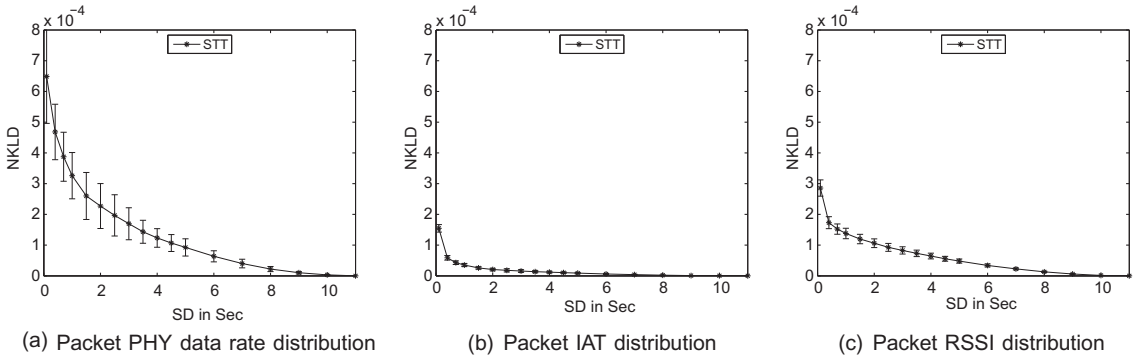


Fig. 7. NKLD vs. SD for STT sampling scheme for different packet distributions.

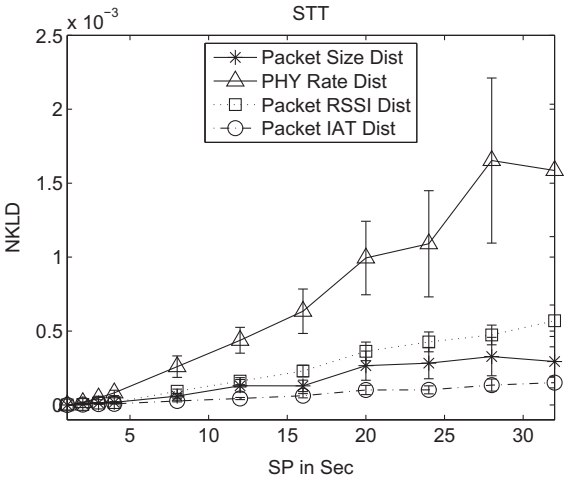


Fig. 8. NKLD vs. SP for STT sampling scheme for different packet distributions.

SP should be at least 11 × SD. Otherwise, it is impractical to perform multi-channel sampling with just one interface card in IEEE 802.11 b/g networks. Hence, ideally SD and SP should maintain a 1:11 ratio.

We developed sampling accuracy graphs for helping network engineers in choosing the best sampling scheme for characterization of multi-channel wireless networks such as enterprise IEEE 802.11 b/g networks. Fig. 9(a) and (b) show variation of NKLD scores for different (SD, SP) combination pairs of STT/SRTT/SCT/SRTT schemes. As one can see from Fig. 9(a) and (b), which show variation of NKLD with SP and SD on log-scale for systematic and stratified random schemes, respectively, there are several possible (SD, SP) combination pairs with 1:11 sampling ratio. Let us consider three such pairs (SD = 250 ms, SP = 2750 ms), (SD = 1 s, SP = 11 s), and (SD = 2 s, SP = 22 s) for STT/SRTT sampling schemes. Here we exclude SCT/SRCT schemes from the discussion because their performance is worse by one to two orders of magnitude compared to STT/SRTT schemes. For any given (SD, SP) combination pair, we can observe that the performance of STT and SRTT sampling schemes is almost identical. However, SRTT schemes are more expensive to implement in real networks as they require the generation of pseudo-random numbers and, therefore, we exclude them from the candidate schemes for best sampling scheme. Among the remaining three (SD, SP) combination pairs of STT sampling, as shown in Fig. 9(a), (SD = 250 ms, SP = 2750 ms) has the best sampling accuracy (*i.e.,* the lowest NKLD score) while (SD = 2 s, SP = 22 s) has the highest NKLD score. This is because when SD is small the sampling scheme is switching across channels very fast and, therefore, able to capture burstiness of traffic more closely, which resulted in better sampling accuracy. However, rapid channel switching at ms or μs granularity may not be practical due to hardware/software limitations of off-the-shelf wireless cards. Hence we have chosen STT with (SD = 1 s, SP = 11 s) as the sampling scheme for IEEE 802.11 b/g networks.

But SP is going to stretch if channel switching is not instantaneous (*i.e.,* CST > 0). While reporting our sampling accuracy results, we assumed that CST = 0 and therefore the 1:11 ratio holds between SD and SP. Our assumption is justified because we are generating samples offline from the complete capture obtained for one particular channel. Ideally, CST should be considered while determining SP time. The following equation gives SP in terms of SD and CST.

$$SP \geqslant 11 \times (SD + CST).$$

CST estimate is between 1 ms and 5 ms for Orinoco, Atheros, and Intersil wireless chipsets. When SD = 1 s and CST = 5 ms, SP ⩾ 11.055 s. From Fig. 9(a) and (b), one can observe that (SD = 1 s, SP = 11.055 s) and (SD = 1 s, SP = 11 s) pairs have almost same NKLD scores. Hence, the effect of CST on sampling accuracy is negligible for commercially available Wi-Fi chipsets and the 1:11 ratio still holds between SD and SP.

For designing an efficient multi-channel sampling system, engineers could choose a desired NKLD value in the sampling accuracy graphs shown in Fig. 9, check a variety of sampling parameters (*i.e.,* (SD, SP) combination pairs for STT/SRTT/SCT/SRTT sampling schemes) that could meet the NKLD requirement, and then decide on the scheme with the minimal implementation cost associated with it. Different schemes require different storage and processing resources, and associated implementation cost. The number of channels to sense plays a decisive role in

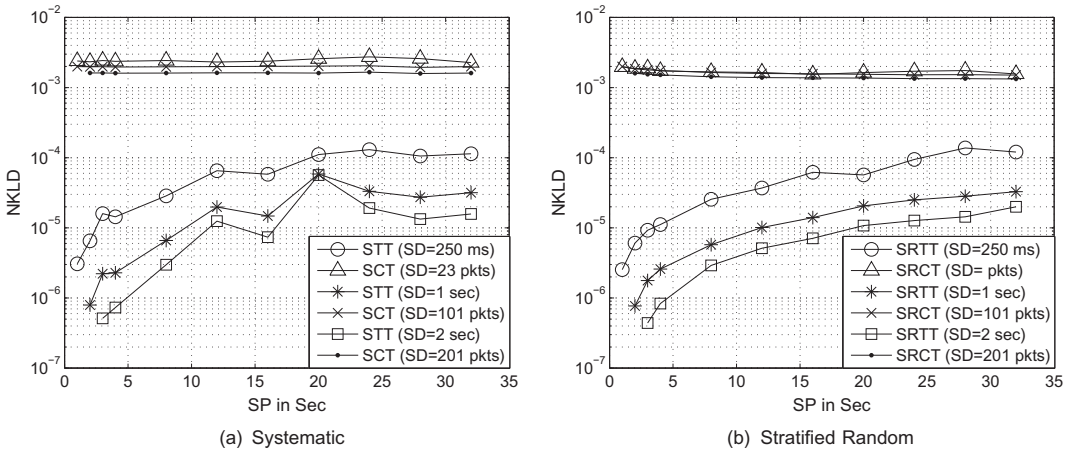


Fig. 9. Sampling accuracy graphs for packet IAT distribution. Note that NKLD values on *y*-axis are plotted in log-scale.

choosing the best scheme because some of the schemes may not meet the NKLD requirement within the feasibility constraints of real systems.

5. Traffic sensor architecture and monitoring testbed

Fig. 10 shows the schematic diagram of the multi-channel traffic sensor. The traffic sensor obtains sensorial information about the surrounding communication environment. This sensorial information is expected to be available in the form of measurements of different types, *e.g.,* traffic information, signal and noise power measurements, as well as time and location coordinates. The traffic sensor node uses an IEEE 802.11 based wireless NIC for sensing or monitoring the traffic related information from each of the IEEE 802.11 channels in the 2.4 GHz spectrum. The main components of the traffic sensor are the time-based sensing control module, traffic sensor module, network error sensor module, and the synchronization and control module. The time-based sensing control module controls the time for which the traffic samples from each of the channels are to be collected. This enables the traffic sensor to sample traffic in all the channels in a predefined static or dynamic fashion. The synchronization and control module performs synchronization of the traffic sensor to a public NTP server over the wired network to achieve time synchronization. The network error sensor module is used to gather additional measurements which can be used to better characterize the wireless medium. In particular, we are interested in collecting additional statistics like the number of CRC errors per unit time (*i.e.,* failed packet receptions due to a bad CRC in the MAC trailer) and PHY errors per unit time (failed packet receptions due to the CRC failure on the PLCP header). The data transfer module dumps all these traffic related measurements into a database repository and thereby provides historical as well as instantaneous traffic information for learning and decision making by the cognitive controller. The cognitive controller can either be implemented locally or realized in a central controller system for global decision making.

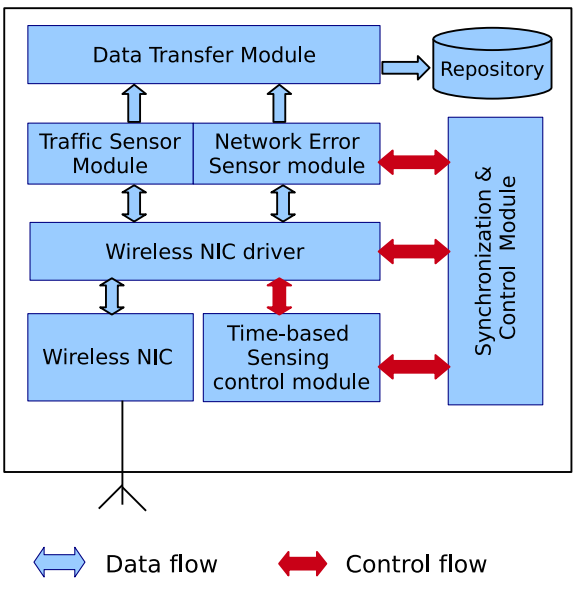


Fig. 10. Architecture of multi-channel traffic sensor node.

All of our wireless traffic monitoring activity took place within the University of California San Diego (UCSD) division of CALIT2, a large six-story building. Avaya APs provide production wireless service, configured for IEEE 802.11 b/g service. Furthermore, there exist some experimental mesh networks in the testbed environment. Among production APs located on 4th and 6th floors, we have deployed 11 traffic sensor nodes as shown in Fig. 11. Each traffic sensor [15] consists of a Soekris net4521 system board with one Ubiquity IEEE 802.11 a/b/g wireless NIC based on the Atheros AR5213 chipset. The wireless NIC is configured in *monitor* mode to capture all IEEE 802.11 packets in the air. Each sensor node runs on Voyage Linux with kernel 2.6.x and uses the open source MadWiFi driver [16]. Each sensor node is connected to the campus intranet via one of the Ethernet interfaces. Traffic sensors use the *tcpdump* packet sniffer [17] to create capture files and remit them to the central database repository of the

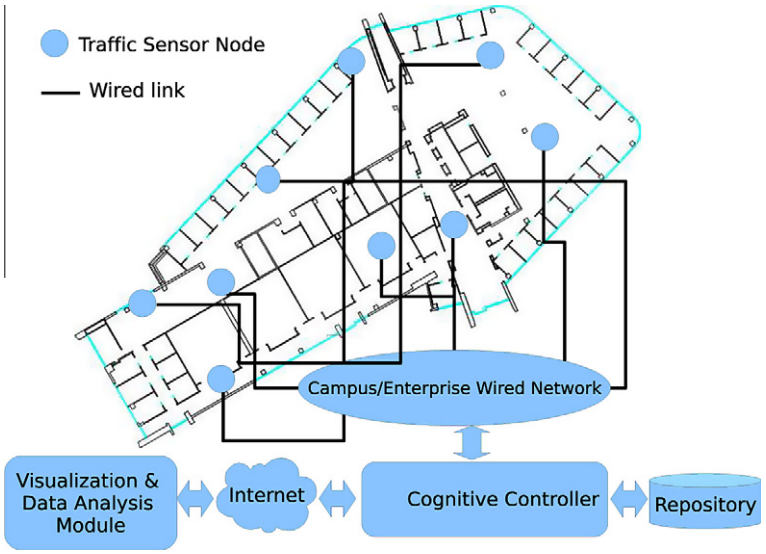


Fig. 11. Traffic sensing testbed infrastructure.

cognitive controller over the wired network. To reduce the storage cost, we configured *tcpdump* to capture only the first 250 bytes (144 bytes for Prism monitoring header) of each sampled packet. This is a reasonable solution, since the TCP header and other protocol headers are located at or near the start of the packet. Implementation of the time-based sensing module is done using a combination of shell scripts running `wireless-tools` [18] and MadWiFi tools. These are used to periodically switch the wireless NIC's channel setting in order to gather traffic samples from all IEEE 802.11 b/g channels; channel switching is done in parallel to the traffic monitoring activity. In order to report additional information about the packet currently being captured, the MadWiFi driver generates the Prism monitoring header, 144 bytes in length, and adds it to the packet. The Prism monitoring header contains, among other information, the RSSI, channel and data rate of the packet.

The cognitive controller with a central database repository is a Dell PowerEdge 1900 server with 7.2 TB of storage. At the central cognitive controller, a modified version of `tcpdump` is employed to read the capture file to extract Prism monitoring header fields and header field values from the MAC through transport layers of the TCP/IP protocol stack. These values are stored in the central database, implemented using MySQL, from which they can be queried for analysis purposes. The cognitive controller stores the information in two separate forms: (i) instantaneous traffic records and (ii) historical traffic records. The instantaneous traffic records contain large amounts of wireless network data traffic tagged with space and time information. The historical traffic records are created from instantaneous traffic records, by averaging data referring to the same location and channel over one-minute intervals. The purpose of having historical traffic records is to speed up subsequent reprocessing of information by not needing to process the complete traffic records every time (this operation being very costly). In addition to traffic samples, we gather other measurements which can be used to better characterize the conditions of the wireless medium. In particular, the MadWIFI driver has a tool (*athstats*) which provides additional statistics like the number of CRC errors and PHY errors.

6. Historical traffic statistics

Though metrics such as the packet rate (packets per second) and the traffic load (bits per second) indicate the busyness of wireless channels to some extent, they fail to quantify the utilization of radio resources in a clear and compact fashion. With this respect, a more representative metric is the traffic intensity (TI), defined as the fraction of time in which the channel is occupied. This metric quantifies utilization of channel resources in a scale from 0 to 1, where 0 indicating channel is completely idle and 0% utilized, and 1 indicating channel bandwidth is completely utilized. However, the challenge is in the accurate estimation of TI in wireless networks that use commercial off-the-shelf NICs. We now present an effective strategy for estimating each channel's TI from the sampled packet traces.

Let K denote the number of packets observed in an interval of duration T. Let d_i denote the transmission duration of the ith packet which also takes into calculation the time required to transmit PHY layer convergence protocol (PLCP) header. Then TI in the interval being considered is TI_T:

$$TI_T = \frac{\sum_{i=1}^{i=K} d_i}{T}. \tag{3}$$

In the case of sampled traffic traces obtained from the STT sampling scheme with SD = 1 s and SP = 11 s, the per-hour calculation of TI in given channel has $T = 60 * 60/11 = 327.27$ s. As per Eq. (3), TI takes into consideration

only the channel resources consumed by successfully received packets. However, corrupt or broken packets also consume a significant amount of channel resources. For this reason, we use the collected CRC error and PHY error measurements, to estimate the amount of resources consumed by corrupt or broken packets. CRC and PHY error measurements are performed synchronously to channel switching, therefore, SD = 1 s and SP = 11 s. We use the average transmission duration of successfully received packets in a given time interval to estimate the TI contribution provided by CRC error packets. Furthermore, we use 24 μs (time taken to transmit ERP-OFDM PLCP header at the basic PHY rate in IEEE 802.11 g cards) as the duration of the contribution to TI given by PHY errors. Clearly this is a lower bound because packets that are transmitted using IEEE 802.11 b cards can have up to 192 μs as their PLCP durations. Let CRC_T and PHY_T denote CRC and PHY error counts in time interval T. Then we define TI_{ERROR} as

$$TI_{ERROR_T} = \frac{\left[\frac{\sum_{i=1}^{i=K} d_i}{K}\right] \times CRC_T + 24\ \mu s \times PHY_T}{T}. \quad (4)$$

The spectral measurements obtained with help of spectrum analyzers can give more accurate estimation of busyness of wireless channels compared to CRC and PHY errors reported by MadWiFi driver. Hence, the proposed TI_{ERROR} metric gives only an approximate bound on channel resources consumed by corrupt or collided packets. In the absence of a sophisticated and expensive spectrum analyzer, the only possibility available to estimate the resources consumed by broken packets is to use our TI_{ERROR} metric.

The total TI for a time interval T is defined as the sum of TI_T and TI_{ERROR_T}, given, respectively by Eqs. (3) and (4). Unlike other metrics, TI metric is able to estimate the busyness of channels by taking into consideration both successfully received packets and corrupt or broken packets. We, therefore, assume that TI metric provides us with an effective way of estimating the amount of channel resources consumed by both IEEE 802.11 transmissions on adjacent channels as well as by non-802.11 interferers in the absence of sophisticated spectrum analysers.

We implemented the STT sampling scheme in all our traffic sensors with SD = 1 s and SP = 11 s and collected samples for several months. We make use of the sampled traffic to estimate population traffic statistics like mean packet rate, mean traffic load, mean data rate, and TI. Ours is the first work reporting spatio-temporal-spectral dynamics from sampled traces of all channels in IEEE 802.11 b/g networks by employing time-based sampling schemes. We now show various traffic statistics for one traffic sensor node (#11) deployed at the middle of the 4th floor of the building. Figs. 12 and 13 show mean traffic loads (bps) and PHY data rates (Mbps) during normal working days for all 11 channels, respectively. Even though orthogonal channels (1, 6, and 11) contain higher traffic due to presence of production network APs, other channels also face traffic to a certain extent due to the leakage of wireless signals (especially those transmitted at basic rate of 1 Mbps, as evident from Fig. 13) into adjacent channels and presence of some experimental wireless

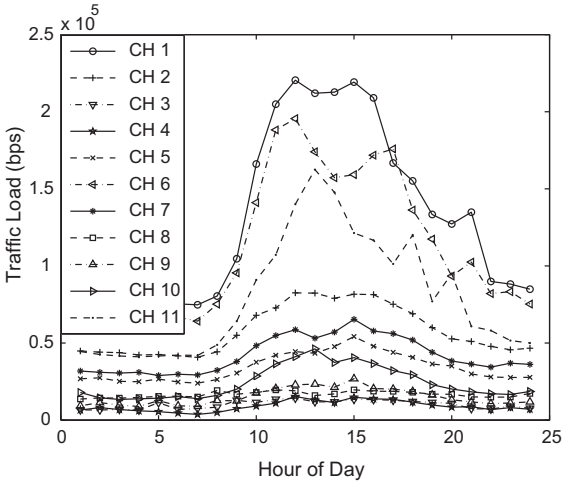


Fig. 12. Working days: traffic load vs. channel vs. time.

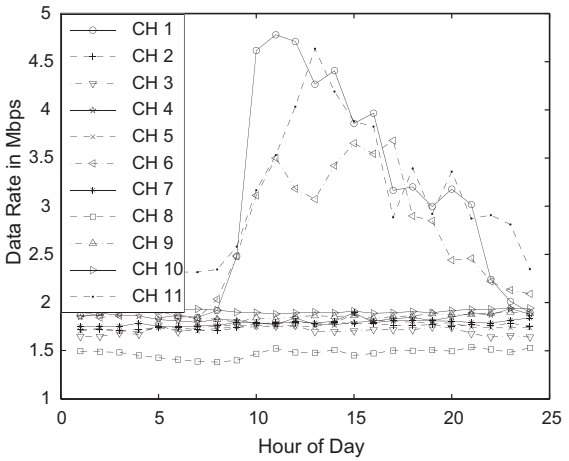


Fig. 13. Working days: PHY data rate vs. channel vs. time.

networks. It can also be observed from Fig. 12 that leakage of packets is more into immediate adjacent channels: 2, 5, 7, and 10. From the above figures, one could come to a conclusion that the traffic is very similar across the orthogonal channels. However, this is not true in terms of the traffic intensity metric, as shown in Figs. 14 and 15, because radio resource consumption is different in those channels. Since wireless signals from orthogonal channels leak into adjacent channels, those channels experience a lot of CRC and PHY errors as shown in Figs. 16 and 17, and actually face some consumption of radio resources (estimated using TI_{ERROR} metric) as shown in Fig. 14. Again immediate adjacent channels report higher number of CRC and PHY errors. Fig. 15 shows the mean total TI (including both the contribution due to successfully received packets and corrupt or broken packets) during working days for all 11 channels. It is to be observed that spectrum utilization is high in Channel 6 compared to the other two orthogonal channels.

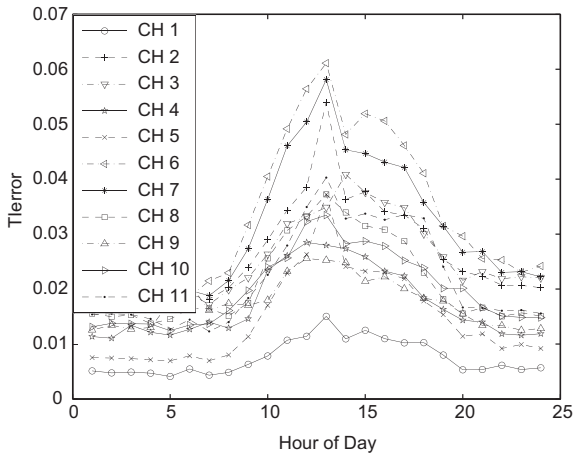


Fig. 14. Working days: TI error vs. channel vs. time.

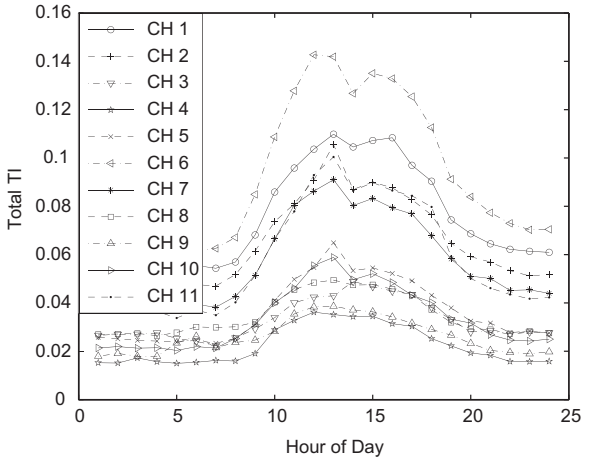


Fig. 15. Working days: total TI vs. channel vs. time.

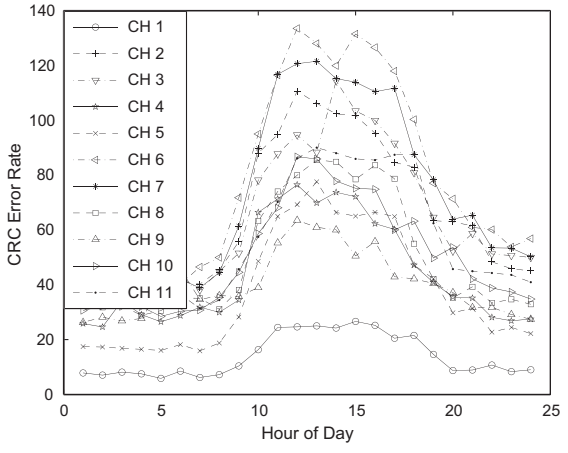


Fig. 16. Working days: CRC error rate vs. channel vs. time.

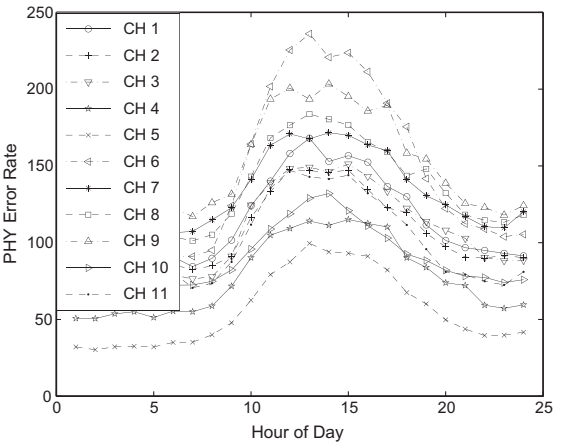


Fig. 17. Working days: PHY error rate vs. channel vs. time.

Table 2
Relative errors.

	RE for packet arrival distribution	RE for traffic load distribution
Loc #1	0.8%	1.19%
Loc #2	2.66%	3.3%

Furthermore, there is a significant difference in traffic patterns of working days and holidays. While the mean traffic increases during business hours of working days, it stays pretty much flat for the whole day during non-working days. It is to be noted that traffic is non zero for non-working hours/days as the APs of the campus production network always send periodic beacons and the small variation in traffic is due to the presence of some experimental mesh networks in the vicinity of our monitoring testbed infrastructure.

The sample traces obtained by traffic sensors through STT sampling scheme with sampling parameters, SD = 1 s and SP = 11 s, hence NKLD values of $\approx 10^{-4}$. In order to better understand how bad NKLD of 10^{-4} and attribute a meaning to NKLD values, we also measured relative error (RE) between sample traces and population traces. The metric RE is defined as follows:

$$RE = \frac{\sum_{h=1}^{24} \frac{|P_h - S_h|}{P_h}}{24}, \tag{5}$$

where $P_h(S_h)$ denotes mean traffic rate of population trace (sampled trace) for a 1-h interval.

We conducted this experiment at two different locations in our office building in the UCSD campus (*i.e.*, done in the same environment used for earlier experiments and no test traffic is injected into the network). For this experiment, we modified the architecture of the traffic sensor node by adding an additional IEEE 802.11 NIC for performing complete capture of traffic on one dedicated channel (*e.g.,* channel 6). So, this modified traffic sensor

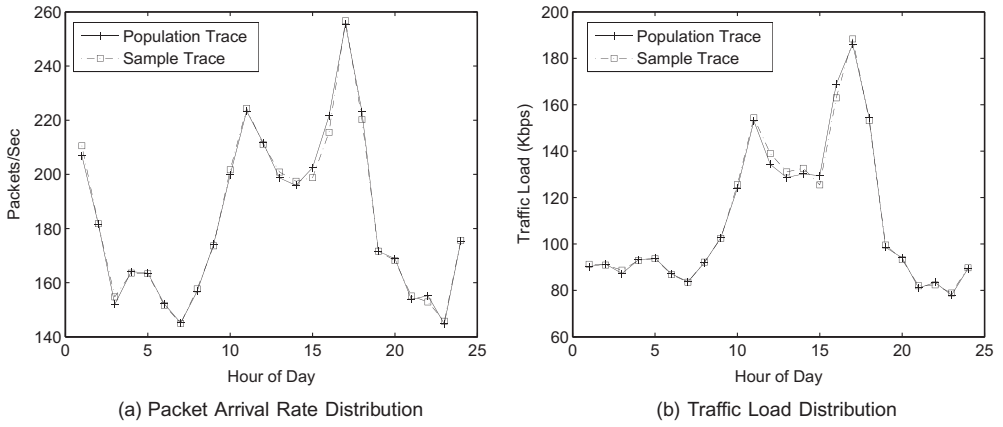


Fig. 18. Comparison of sampled traffic traces with population traffic traces.

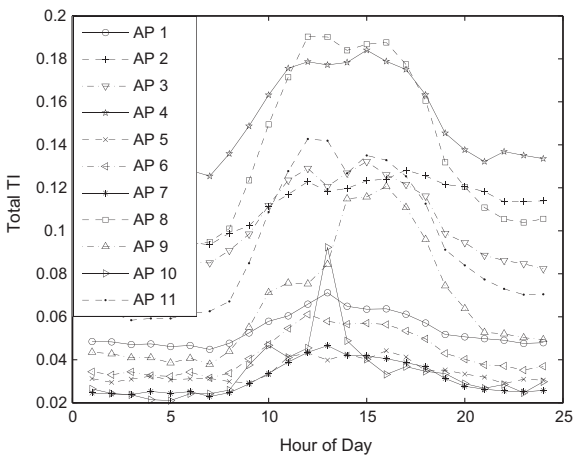


Fig. 19. Working days, channel 6: total TI vs. space vs. time.

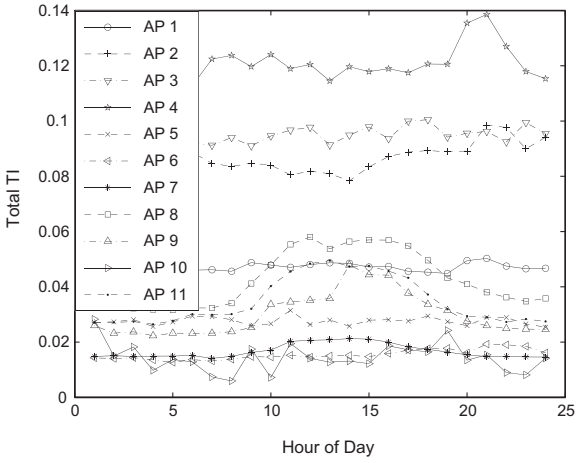


Fig. 20. Working days, channel 8: total TI vs. space vs. time.

provides us with two traces: one sampled traffic trace of all 11 channels based on STT with sampling parameters, SD = 1 s and SP = 11 s, and one complete packet trace of channel 6. The traces are collected over a period of one week. We then extracted the channel 6 sampled trace from the 11 channels sampled trace and compared it with the complete trace of channel 6 in terms of NKLD and RE. We observed NKLD of $\approx 10^{-4}$ for different per-packet traffic distributions such as packet sizes and traffic loads.

We measured RE values of mean packet arrival rate (packets/s) and mean traffic load (Kbps) metrics for traces collected at two different locations and reported them in Table 2. We observe some variation in observed RE values for the two measurement locations, but still RE is less than 5%. Fig. 18 shows a snapshot of a time-series comparison between population and sampled traces for the distributions of packet arrival rates and traffic loads for one of the two measurement locations. We can see that the sampled trace matches the population trace very closely. Hence, we conclude that the proposed STT sampling scheme with sampling parameters, SD = 1 s and SP = 11 s, has very good sampling accuracy and therefore is a good choice for multi-channel sampling in IEEE 802.11 b/g networks.

Table 3
List of acronyms.

CRC	Cyclic Redundancy Check
CST	Channel Switch Time
IAT	Inter-Arrival Time of packets
KLD	Kullback–Leibler Divergence
NIC	Network Interface Card
NKLD	Normalized Kullback–Leibler Divergence
NTP	Network Time Protocol
PLCP	Physical Layer Convergence Protocol
RSSI	Received Signal Strength Indicator
SCC	Systematic Count-driven Count-based sampling
SCT	Systematic Count-driven Time-based sampling
SD	Sampling Duration
SP	Sampling Period
SR	Simple Random sampling
SRCC	Stratified Random Count-driven Count-based sampling
SRCT	Stratified Random Count-driven Time-based sampling
SRTT	Stratified Random Timer-driven Time-based sampling
STT	Systematic Timer-driven Time-based sampling
TI	Traffic Intensity

We finally show plots of Total TI in a spatio-temporal fashion by fixing the channel. Traffic sensors (nodes 1–7 and 8–11) are deployed on the 6th and 4th floors of the UCSD CALIT2 building, respectively. Fig. 19 shows the total TI of an orthogonal channel (6). From this figure we can observe that Total TI is also varying in the spatial dimension and it is highest at node 8 deployed on the 4th floor of building. Due to poor user population in some parts of the building, some sensors consistently report low spectrum utilization for all the channels. Fig. 20 shows total TI of a non-orthogonal channel (8). It is to be noted that double counting of same packet by different sensors is possible if two nearby traffic sensors overhear same set of packets. However, our characterization does not try to remove over counting because for spatial characterization we are interested in seeing what is the traffic at each location in the network environment. Cognitive access points observing such information can take their decisions (*e.g.,* dynamically selecting the best channel for serving wireless clients) based on their perceived local traffic load conditions.

7. Conclusions

In this paper, we studied the sampling accuracy of time-based sampling schemes such as STT, SRTT, SCT, and SRCT schemes by using packet traces collected in a campus IEEE 802.11 network. From our experiments, we found that the systematic timer-driven time-based (STT) sampling strategy is the best sampling strategy for traffic sensing and characterization in multi-channel wireless networks. We developed a prototype traffic sensor device which implements the STT sampling scheme and then presented the characterization of our campus IEEE 802.11 b/g network based on sampled traces collected by traffic sensor devices. We further developed sampling accuracy graphs that can be used by network engineers for designing multi-channel wireless traffic sampling systems by choosing the combination of SP and SD parameters and sampling scheme for a desired NKLD score. Among the various schemes that meet the NKLD requirement a scheme that meets the implementation cost can be chosen. Traffic sensing and characterization is essential for realizing cognitive networking systems. Our future work includes the design of a cognitive network controller which can use the sampling strategies and traffic statistics that we discussed to observe the surrounding environment (in this case, the wireless spectrum) in a spatio-temporal fashion, to understand the current channel conditions and the status of the network, and finally to select the most desirable network configuration for achieving network-wide performance goals.

Appendix A

See Table 3.

References

[1] R.W. Thomas, D.H. Friend, L.A. DaSilva, A.B. MacKenzie, Cognitive networks: adaptation and learning to achieve end-to-end performance objectives, IEEE Communications Magazine 44 (12) (2006) 51–57.

[2] B.S. Manoj, M. Zorzi, R.R. Rao, Architectures, protocols, and analytical approaches for next generation cognitive networking, in: Katz Marcos et al. (Eds.), Book Chapter in Cognitive Wireless Networks: Concepts, Methodologies and Visions, Springer, 2007.

[3] Ian F. Akyildiz, Won-Yeol Lee, Mehmet C. Vuran, Shantidev Mohanty, Next generation/dynamic spectrum access/cognitive radio wireless networks: a survey, Computer Networks 50 (9) (2006) 2127–2159.

[4] T. Weiss, F. Jondral, Spectrum pooling: an innovative strategy for the enhancement of spectrum efficiency, IEEE Communications Magazine 42 (3) (2004) 8–14.

[5] M. Balazinska, P. Castro, Characterizing mobility and network usage in a corporate wireless local-area network, ACM Mobisys (2003) 303–316.

[6] J. Yeo, M. Youssef, T. Henderson, A. Agrawala, An accurate technique for measuring the wireless side of wireless networks, in: Workshop on Wireless Traffic Measurements and Modeling, 2005, pp. 13–18.

[7] Y. Cheng, J. Bellardo, P. Benkö, A.C. Snoeren, G.M. Voelker, S. Savage, Jigsaw: solving the puzzle of enterprise 802.11 analysis, SIGCOMM Computer Communication Review 36 (4) (2006) 36–50.

[8] A. Sharma, E.M. Belding, FreeMAC: framework for multi-channel MAC development on 802.11 hardware, in: ACM Workshop on Programmable Routers for Extensible Services of Tomorrow, 2008, pp. 69–74.

[9] D. Murray, M. Dixon, T. Koziniec, Scanning delays in 802.11 networks, in: Proceedings of the International Conference on Next Generation Mobile Applications, Services and Technologies, 2007, pp. 255–260.

[10] K.C. Claffy, G.C. Polyzos, H.-W. Braun, Application of sampling methodologies to network traffic characterization, Computer Communication Review 23 (4) (1993) 94–203.

[11] U. Deshpande, T. Henderson, D. Kotz, Channel sampling strategies for monitoring wireless networks, in: Proceedings of WiOpt, April 2006, pp. 1–7.

[12] B.R. Tamma, B.S. Manoj, R.R. Rao, Multi-channel wireless traffic sensing and characterization for cognitive networking, in: Proceedings of IEEE ICC, June 2009.

[13] N. Duffield, Sampling for passive internet measurement: a review, Statistical Science 19 (3) (2004) 472–498.

[14] S. Kullback, Information Theory and Statistics, John Wiley and Sons Inc., New York, 1959.

[15] <http://calnode.calit2.net>.

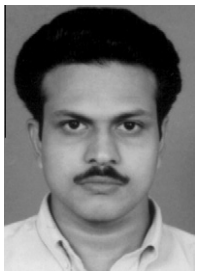
[16] <http://madwifi-project.org/>.

[17] <http://www.tcpdump.org/>.

[18] <http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Tools.html>.



Bheemarjuna Reddy Tamma received the B.Tech. degree in Computer Science and Engineering from Andhra University, India, in 2000 and the M.E. degree in Computer Science and Engineering from the National Institute of Technology (NIT), Rourkela, India, in 2002. He was an iNautix doctoral student during 2002–2006 in the Department of Computer Science and Engineering at the Indian Institute of Technology (IIT) Madras, India, where he focused on QoS provisioning and Multimedia transport in Ad hoc wireless networks. From March 2007 to October 2010, he was a post-doctoral researcher at the University of California, San Diego, USA. Since October 2010, he is working as an Assistant Professor in the Department of Computer Science and Engineering, IIT Hyderabad. His research interests include Quality of Service, Multimedia transport, Cognitive networks, Green communication networks, and Cyber-Physical systems.



B.S. Manoj completed his graduation in 1995 and post graduation in 1998 both in Electronics and Communication Engineering from Institution of Engineers (India) and Pondicherry Central University, Pondicherry, India, respectively. He has worked as a Senior Engineer with Banyan Networks Pvt. Ltd., Chennai, India from 1998 to 2000 where his primary responsibility included design and development of protocols for real-time traffic support in data networks. He was an Infosys doctoral student during 2000–2003 in the Department of Computer Science and Engineering at the Indian Institute of Technology (IIT) Madras, India, where he focused on the development of architectures and protocols for Ad hoc wireless networks and next generation hybrid wireless network architectures. During January 2004–January 2005, he was a Project Officer at IIT, Madras, India. From January 2005 to January 2011, he was at the University of California, San Diego, USA, working initially as a post-doctoral researcher and later as an associate research scientist and lecturer. Since January 2011, he is working as an Associate Professor at the Indian Institute of Space Science and Technology (IIST), Trivandrum, India. His research interests include Ad hoc wireless networks, next generation wireless architectures, delay tolerant networks, and wireless sensor networks.



Ramesh Rao is the Director of University of California San Diego division of the California Institute for Telecommunications and Information Technology (Calit2). He has been a faculty member in the Department of Electrical and Computer Engineering, Jacobs School of Engineering, UC San Diego since 1984. Prior to becoming the Calit2 UCSD division director in 2001, he was the director of the Center for Wireless Communications (CWC) at UCSD. He is involved on a day-to-day basis with a wide variety of research initiatives at Calit2. He leads several major interdisciplinary and collaborative projects and has been a PI on dozens of federal-, state-, foundation- and industry-funded grants. He is an IEEE fellow and also the QUALCOMM Endowed Chair in Telecommunications and Information Technologies, UC San Diego.