Short-term traffic forecasting in a campus-wide wireless network

Maria Papadopouli[a,c] Haipeng Shen[b] Elias Raftopoulos[c] Manolis Ploumidis[c] Felix Hernandez-Campos[a]

***Abstract*— Our goal is to characterize the traffic load in an IEEE802.11 infrastructure. This can be beneficial in many domains, including coverage planning, resource reservation, network monitoring for anomaly detection, and producing more accurate simulation models. The key issue that drives this study is traffic forecasting at each wireless access point (AP) in an hourly time-scale. We conducted an extensive measurement study of wireless users on a major university campus using the IEEE802.11 wireless infrastructure. We observed a spatial locality in the most heavily utilized APs. We propose several traffic models that take into account the periodicity and recent traffic history for each AP and present a time-series forecasting methodology. Finally, we build and evaluate these forecasting algorithms and discuss our findings.**

I. Introduction

Wireless networks are increasingly being deployed and expanded in airports, universities, corporations, hospitals, residential, and other public areas to provide wireless Internet access. Furthermore, there is an increase in peer-to-peer, streaming, and VoIP traffic over the wireless infrastructures[9], [8]. At the same time, empirical studies and performance analysis indicate dramatically low performance of real-time constrained applications over wireless LANs (such as [2] on the VoIP). Currently APs do not perform any type of forecasting or admission control and clients frequently experience failures and disconnections when there is high demand in the wireless infrastructure.

The shared medium wireless LANs have more vulnerabilities and bandwidth and latency constrains than their wired counterparts. The bandwidth utilization at an AP can impact the performance of the wireless clients in terms of throughput, delay, and energy consumption. For quality of service provision, capacity planning, load balancing, and network monitoring, it is critical to understand the traffic characteristics. While there is a rich literature characterizing traffic in wired networks ([11], [10], [15], [7]), there are only a few studies available that examined wireless traffic load. The key issue that drives this study is forecasting in an hourly time scale. We aim to enable APs to perform short-term forecasting in order to perform better load balancing, admission control, and quality of service provisioning. Specifically, they can use the expected traffic estimations to decide whether or not to accept a new association request or advise a client to associate with a neighboring AP. In addition, the traffic models can assist in detecting abnormal traffic patterns (e.g., due to malicious attacks, AP or client misconfigurations and failures).

a. Department of Computer Science, University of North Carolina at Chapel Hill, US.

b. Department of Statistics & Operations Research, University of North Carolina at Chapel Hill, US.

c. ICS-FORTH, Greece.

In this paper, we study a large wireless infrastructure[1] using a lightweight data acquisition methodology. Our data was collected using the Simple Network Management Protocol (SNMP), the most widely available monitoring service in wireless platforms. Any AP in the market supports monitoring using SNMP, so it is important to understand how much operators and researchers can learn from SNMP data. Furthermore, this type of data is the most appropriate one to understand daily and long-term trends in the usage of wireless networks, and it provides a good foundation for traffic forecasting and capacity planning. Other types of data, such as packet or flow level data, are generally too detailed for this purpose, and their acquisition is much more resource-intensive. This paper makes use of SNMP data for analyzing traffic characteristics, such as total load and periodicities. Then, we suggest some models and build forecasting algorithms based on them.

We summarize our main contributions: We distinguish the most heavily utilized APs and analyze their traffic load. We discover that several of them exhibit spatial locality and diurnal periodicities. We also observe diurnal periodicities at the total traffic load of the wireless infrastructure and also at several APs. Based on its periodicity and recent traffic history, we propose several models for the traffic load at an AP. We build a traffic forecasting methodology that employs these models and evaluates their performance. To the best of our knowledge, this is the first study on traffic forecasting using actual traces from an IEEE802.11 infrastructure.

Section II describes briefly the wireless infrastructure at UNC, and data acquisition process. We define the hotspots and discuss their spatial locality property in Section III. Section IV focuses on forecasting algorithms and Section V presents their performance evaluation. In Section VI, we discuss previous related research. Section VII summarizes our main results and discuss future work.

II. Background

A. Data acquisition

The IEEE802.11 infrastructure at the University of North Carolina at Chapel Hill provides coverage for 729-acre campus and a number of off-campus administrative offices. The university has 26,000 students, 3,000 faculty members, and 9,000 staff members. Undergraduate students (16,000) are required to own laptops, which are generally able to communicate using the campus wireless network. A total of 488 APs were part of the campus network at the start of our study. These APs belong to three different series of the Cisco Aironet platform: the state-of-the-art 1210 Series (269 APs), the widely deployed 350 Series (188 APs) and the older 340 Series (31 APs).

The data in this paper was collected using SNMP for polling every AP on campus every five minutes. We developed a custom data collection system, being careful to avoid the pitfalls described in [8]. First, the system was implemented using a non-blocking SNMP library for polling each AP precisely every five minutes in an independent manner. This eliminates any extra delays due to the slow processing of SNMP polls by some of the slower APs. The system ran in a multiprocessor system and the CPU utilization in each of the three processors we employed never exceeded 70%. Second, our characterization of the workload of the APs is derived only from those clients associated with the AP at polling time (and not from roaming ones associated with a different AP).

The results in this paper were derived from SNMP data collected between 9:09 AM September 29th 2004 and 12 AM November 30th, 2004. The total number of polling operations during the 63 days was 8,247,479. The data collection system ran flawlessly for the entire period, but APs were sometimes unresponsive. This is generally due to maintenance downtimes, reboots, or overloads. If an AP did not respond to a poll, the data collection system tried again after 5s (and if necessary, again after 10s and 15s). It is therefore unlikely that datagram losses created holes in our dataset.

Our dataset includes 14,712 unique MAC addresses which were associated with one or more APs during the data collection period. This means that the population of wireless devices at UNC during our measurement was three times larger than the one considered in Kotz *et al.*'s studies [9], [8]. We were unable to collect association information from the oldest APs (the 31 Cisco Aironet 340s) using SNMP, so we do not consider them in our study.

B. Traffic load notation

Based on the SNMP trace for each AP, we produce a time series of its traffic load at hourly intervals. This traffic is the total amount of bytes received and sent from all clients that were associated with the AP at that time interval. In the rest of the paper, depending on the mathematical expression, we will use two notations for these time series. Specifically, the traffic of the AP i during the h-th hour of day d, that corresponds to time t, is $T_i(h, d) = X_i(t)$.

III. Hotspots APs and their spatial locality

We would like to distinguish the most heavily utilized APs. For that, we define the *hotspots* of the wireless infrastructure based on three metrics, namely, the maximum hourly traffic,the total traffic and the maximum daily traffic.

Hotspots based on maximum hourly traffic (set 1)
These are the top $\alpha\%$ APs ordered by their maximum traffic during an hour in the entire tracing period.

Hotspots based on total traffic (set 2)
These are the top $\alpha\%$ APs ordered by their total traffic during the tracing period.

Hotspots based on maximum daily traffic (set 3)
These are the top $\alpha\%$ APs ordered by their maximum traffic during a day in the entire tracing period.

Hotspot (main definition)
We define as a *hotspot* an AP that belongs in the top $\alpha\%$ of APs with the highest maximum hourly traffic and in the top $\alpha\%$ of APs with either the highest total traffic load or the highest maximum daily traffic load (i.e., the set $(set1 \cap (set2 \cup set3))$). We will use this definition in the following sections.

We first investigate the spatial locality of the hotspots and name two APs co-located, if they are placed in the same building. How likely is to find co-located hotspots in the campus? We found that for α=20, the percentage of co-located hotspots is above 76% and 79% for the hourly and total-traffic based definitions, respectively. 62% of the co-located APs belong in the $(set1 \cap (set2 \cup set3)))$. For $\alpha = 10$, the corresponding percentages are about 11% smaller than their respective values for $\alpha = 20$. Note that, if using the uniform distribution, we had randomly selected the same number of APs, the mean percentage of co-located APs in those selections is 48%. We are currently investigating other spatial locality properties of the hotspots (such as visit duration, applications, number of distinct clients, and usage patterns) and plan to report these results in a followup study. For $\alpha = 10$, there are 19 such APs.

IV. Traffic load modeling and forecasting

We will describe two different forecasting approaches, namely, simple predictions based on historical means and recent traffic and time-series forecasting. Our general methodology consists of the following steps: (A) Time-series extraction, data cleaning, and treatment of missing values; (B) Power spectrum and partial autocorrelation analysis; (C) Data normalization and traffic load modeling; and (D) Forecasting using the traffic load models.

A. Time-series extraction and treatment of missing values

While our monitoring system requested traffic load information from each access point precisely every five minutes, missing values are relatively frequent in our dataset. They are due to several reasons: (1) an access point may be down for maintenance, or in the middle of an accidental reboot; (2) an access point may be too busy to reply to an SNMP query; (3) the network path between our monitor and the access point may be temporarily broken; and (4) query packets and response packets may be lost (they are transported using UDP). While these pathologies are expected to be infrequent, our dataset is large enough to contain numerous instances of each of them. Thanks to the cumulative nature of SNMP counters, we were able to reconstruct missing values quite accurately.

The basic technique for extracting an equally-spaced time-series $X = \{x_1, x_2, \ldots, x_n\}$ from SNMP data is to subtract the cumulative counters from two consecutive *polling* operations. In order to detect missing values and reboots, our polling samples include not only the cumulative counters but also the time of each polling operation, and the cumulative time that the access point has been running since the last reboot (*up time*). This means that the i-th polling sample for an access point has the form (t_i, u_i, c_i), where t_i is time of the polling operation, u_i is the cumulative up time, and c_i is some cumulative counter (*i.e.*, total load in bytes). Given two consecutive polling samples, the load x_i observed between t_{i-1} and t_i is generally equal to $c_i - c_{i-1}$. There are two exceptions. First, SNMP counters are represented using 32 bits, so counters often wrap-around. We consider that a counter has wrapped around whenever $c_i < 2^{30}$ and $c_{i-1} > 3{*}2^{30}$. In this case, x_i is equal to $c_i+(2^{32}-1-c_{i-1})$. Second, after a reboot, all the counters in an access points are reset. Therefore, if a reboot occurs at some point between t_{i-1} and t_i, x_i is equal to c_i and the value of c_{i-1} should not be subtracted from c_i. Reboots can be detected by checking the value u_i in each polling sample. If u_i is significantly less than $t_i - t_{i-1}$, the access point has been reset, and x_i is equal to c_i. Otherwise, x_i is equal to the subtraction of the two cumulative counters. Note that resets may create situations that look like a wrap-around, so the detection of the reboots should be performed before the detection of the wrap-arounds.

When all of the polling operations are succesful, $t_i - t_{i-1}$ is equal to the polling interval (*i.e.*, 5 minutes). However, when a polling operation fails, $t_i - t_{i-1}$ is a multiple of the polling interval. If this is due to an access point reboot, the counter c_i only reports on the activity since the reboot operation. Therefore, c_i becomes the last value of the time-series. The values between t_{i-1} and t_i, for which no polling samples were available, are set to zero (access points have no load while off-line). If no reboot took place, the $c_i - c_{i-1}$ does not correspond to a single x_i but to the m values of the time-series between t_{i-1} and t_i. In this case, we perform linear interpolation and set each intermediate value of the time-series to $(c_i - c_{i-1})/m$. Finally, note that $t_i - t_{i-1}$ is not always exactly equal to the polling interval (or a multiple of it). The most significant cause is the retransmission mechanism in our SNMP monitor, which retransmits unanswered requests up to three times. Each new request is spaced by 5 seconds. Therefore, the maximum deviation of $t_i - t_{i-1}$ with respect to the polling interval is 20 seconds, and our time-series extraction program takes into account this deviation.

B. Spectrum analysis

We find that the aggregate hourly traffic for all APs in the infrastructure exhibits diurnal and weekly periodicities. Similar trends are observed in the hourly traffic for several APs by autocorrelation plot and spectrum analysis. 10 out of the 19 hotspots have a clear spike at 24 hours/cycle and do not have a high frequency variation. Also, some APs have weekly patterns at around 168 hours/cycle.

Figure 1(a) and (b) show the time series and spectrum plots of the hotspot AP 472. This AP exhibits strong diurnal periodicity. There are other APs with no clear periodic pattern, for which there is little prediction power among the historical data. Further smoothing does not appear to be helpful, at least with our current relatively short traces.

C. Forecasting using historical means and recent traffic

First, we model the traffic load at an AP during an hour. The model facilitates the diurnal and weekly periodicity of the traffic load. We define the *historical mean hour* traffic of an AP as the mean of the traffic during that hour for each day in the history of that AP (N_{days} days). We only consider weekdays. For example, the historical mean-hour traffic for AP i is defined as

$$\mu_i(h) = (1/N_{weekdays}) \times \sum_{d=1}^{N_{days}} T_i(h,d) * IsAWeekday?(d),$$

where $h = 1, \ldots, 24$ and $IsAWeekday?(d)$ is a binary indicator function that specifies whether or not the d-th day is a weekday, and $N_{weekdays} = \sum_{d=1}^{N_{days}} IsAWeekday?(d)$.

Similarly, the *historical mean hour-of-day* traffic is the mean of the traffic at such hour of day in the history of that AP. For example, the mean hour-of-day for AP i is defined as

$$\mu_i(h,l) = (1/nw(l)) \times \sum_{k=1}^{N_{days}} IsWeekday?(k,l) \times T_i(h,k),$$

where $h = 1, \ldots, 24$, l "runs" from "Mon" through "Sun", and

$$nw(l) = \sum_{k=1}^{N_{days}} IsWeekday?(k,l).$$

The $IsWeekday?(x,l)$ is a binary indicator function that specifies whether or not the x is a weekday l. The $nw(l)$ counts the total number of weekdays l (e.g., the number of Mondays). For example, for the $\mu_i(2)$, we take the historical mean of the traffic at AP i for all days in the history at 2am. Similarly, for the $\mu_i(2, \text{``}Mon\text{''})$, we compute the mean of the traffic of all Mondays at 2am.

We taylor two simple models based on the historical mean hour and mean hour-of-day. Specifically, for each AP (e.g., AP i), we define the models Z_i^1 and Z_i^2, as follows:

$$\begin{aligned} (P1) \quad & Z_i^1(h,d) = \mu_i(h) \\ (P2) \quad & Z_i^2(h,d) = \sum_{l \in \{Mon,..,Sun\}} IsWeekday?(d,l) \times \mu_i(h,l). \end{aligned}$$

To incorporate the recent traffic information in the traffic model, we compute the mean traffic during the last w hours. For each AP (e.g., AP i), we introduce the *weighted average of the recent traffic mean and the historical mean hour and hour-of-day*, Z_i^3 defined as

$$\begin{aligned} (P3) \quad Z_i^3(h,d) = a \times (1/w) \sum_{k=t-w}^{t-1} X(k) + \\ b \times \mu_i(h,d) + c \times \mu_i(h). \end{aligned}$$

We experiment with different window sizes and weights to evaluate the impact of the recent history and periodicity on forecasting. Note that the P3 with weights (a,b,c) equal to (1,0,0) and history window w has the form of an autoregressive process of order w, $AR(w)$. In that case, the prediction takes into account only the recent traffic history instead of the periodicity. We can specify the weights of the P3 using multiple linear regression. The purpose of the multiple linear regression is to establish the relationship among the group of predictors, namely, the history window, historical mean hour traffic, and the historical mean hour-of-day. This allows us to understand which predictors have the greatest effect. The linear model takes the form $y = Xb + e$, where y is a vector of observations, X is a matrix of independent variables (regressors/predictors) and e

is a vector of random disturbances. Multiple linear regression aims to obtain the best fitting curve by minimizing the least square errors ($\sum_{i=1}^{n}[y - f(X_i)]^2 = \sum_{i=1}^{n}[y_i - (bX_i)]^2$). P3 with weights defined using multiple linear regression is denoted as P3-MLR.

We propose three simple prediction algorithms based on the aforementioned models. P1 and P2 use the historical means to compute the $Z_i^k(h, d), k = 1, 2$ for P1 and P2, respectively, and predict the traffic load of AP i during the t-th time interval (that corresponds to the h-hour of day d). P3 integrates the historical means of hour and hour-of-day with the recent traffic history. More specifically, P3 is an *one-step ahead prediction algorithm*, since for the recent traffic, it uses the *actual* traffic values as opposed to the predicted ones (for the next-hour prediction).

D. Normalized ARIMA based time-series forecasting

There are hotspot APs whose traffic load shows strong diurnal periodicity. Figure 1(a) shows the time series plot of the hourly traffic load at AP 472, from which one can observe a clear diurnal pattern as well as a possible weekly pattern. To verify their existence, we plot the corresponding power spectrum in Figure 1(b). The plot indicates that the most dominate period is the 24-hour one, with smaller ones corresponding to 12 hours (day/night), and 168 hours (weekly period). Similar periodicities are observed in several other APs as well. The existence of such strong periodicities motivates us to consider forecasting traffic load using some time series models like ARIMA. Intuitively, one would expect such models will have a better forecasting performance than the aforementioned three algorithms. Because such models take into account the strong periodic patterns as well as the auto-correlation among the hourly traffic load. Below we propose one such time series forecasting model using illustration with the traffic load observed at AP 472.

Suppose $X_i(t)$ is the traffic load within hour t at a particular AP i (e.g., AP 472). Due to the nature of the wireless network traffic, $X_{472}(t)$ has local spikes that are very hard to predict as illustrated in Figure 1(a). In addition, it most likely has a skewed marginal distribution. Figure 1(c) plots the normal quantile plot of $X_{472}(t)$ for AP 472, which clearly suggests the marginal distribution of the traffic load is heavily skewed to the right. This calls for a suitable transformation to make the data closer to a normal distribution. Such a transformation can reduce the effect of those local spikes on the forecasting performance. In addition, standard time series modelling procedures are most suitable for situations with normal data [5]. After experimenting with different transformations, the $1/4$ power transformation, $Y(t) = X_{472}^{1/4}(t)$, seems to give the best result. In particular, Figure 1(d) gives the normal quantile plot for the transformed load $Y(t)$ at AP 472. As one can see, $Y(t)$ is much closer to be normally distributed, and does not have extreme outliers as those in Figure 1(d). The following model will be performed on $Y(t)$.

We first point out that $Y(t)$ exhibits strong non-stationarity in both the mean and the variance. Figure 2(a) plots the bimodal changing patterns of its mean, median, 25-th percentile and 75-th percentile as functions of hour-of-day ($h(t)$), which shows that both the mean and the percentiles change across the day. For example, the mean curve suggests that there is very little traffic between midnight and 7-8AM; then the load starts to increase until it reaches the first mode around 10AM and stays flat until noon; after lunch-break, the load increases again to the second mode around 3PM before it starts to decrease until midnight. Very sensible explanations can be given for such a diurnal pattern. Similarly, Figure 2(b) indicates the diurnal patterns for the standard deviation and Inter Quartile Range (IQR) (*i.e., the difference between the 25-th and 75-th percentiles*). The plot suggests that there is increasing variability in the traffic load during 7AM-10AM and 1PM-3PM, exactly when the load increases. In addition, the variability stays small between 10AM and 1PM.

The above exploratory data analysis motivates us to normalize the transformed load $Y(t)$ in the following way,

$$e(t) = \frac{Y(t) - \mu_{h(t)}}{\sigma_{h(t)}},$$

where $h(t)$ is the corresponding hour-of-day for time t, $\mu_{h(t)}$ is the mean of $Y(t)$ during those time periods with the hour-of-day being $h(t)$ while $\sigma_{h(t)}$ is the standard deviation of $Y(t)$ during those time periods, and $e(t)$ can be treated as a normalized version of $Y(t)$. Note that $\mu_{h(t)}$ and $\sigma_{h(t)}$ have been plotted in Figure 2(a) & (b) for AP 472.

After the normalization, we can assume $e(t)$ to be a stationary time series as shown in Figure 2(c). The corresponding partial autocorrelation function (Partial ACF) (Figure 2(d)) suggests that an AR(1) model is reasonable for the normalized time series, $e(t)$. Thus, we fit a family of AR(p) models to $e(t)$ using the Yule-Walker method and select the approximate order p by minimizing the Akaike Information Criterion (AIC). See Brockwell and Davis (1998) for details about the estimation method and the model selection criterion, AIC. Note that the order p specifies the number of lagged variables in the time series model and the AR(p) model is written as

$$e(t) = a_1 e(t-1) + \ldots + a_p e(t-p) + n(t),$$

where $n(t)$ is the model residual.

As for the load at AP 472, p is selected to be 1 and the fitted AR(1) model is

$$e(t) = 0.5689e(t-1) + n(t) \tag{1}$$

with the residuals $n(t)$ being normally distributed with mean 0 and variance 0.6349. One can then use (1) to predict the traffic load during the next hour, corresponding to time $(t+1)$, $X_{472}(t+1)$. First, a point prediction for $e(t+1)$ can be obtained as

$$\hat{e}(t+1) = 0.5689e(t);$$

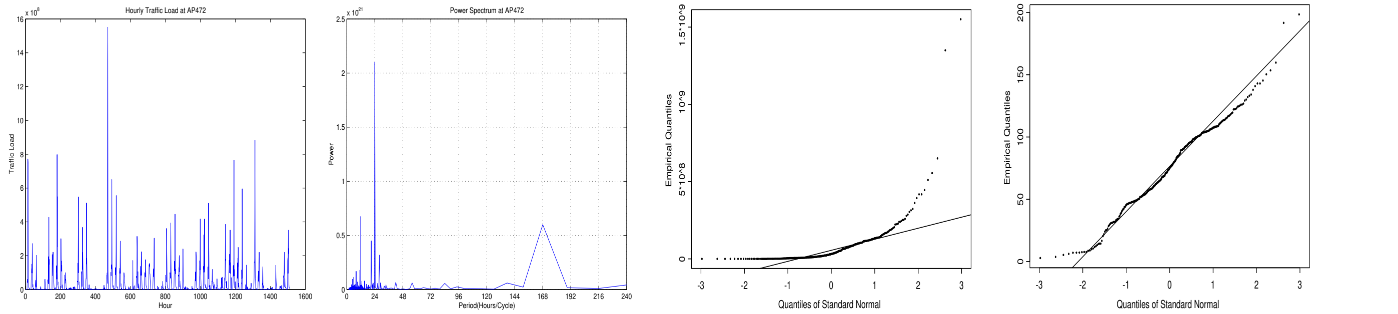
then $Y(t+1)$ can be predicted as

$$\hat{Y}(t+1) = \mu_{h(t+1)} + \sigma_{h(t+1)} \times \hat{e}(t+1).$$

Finally, a point forecast for $X_{472}(t+1)$ is obtained by back-transforming $\hat{Y}(t+1)$,

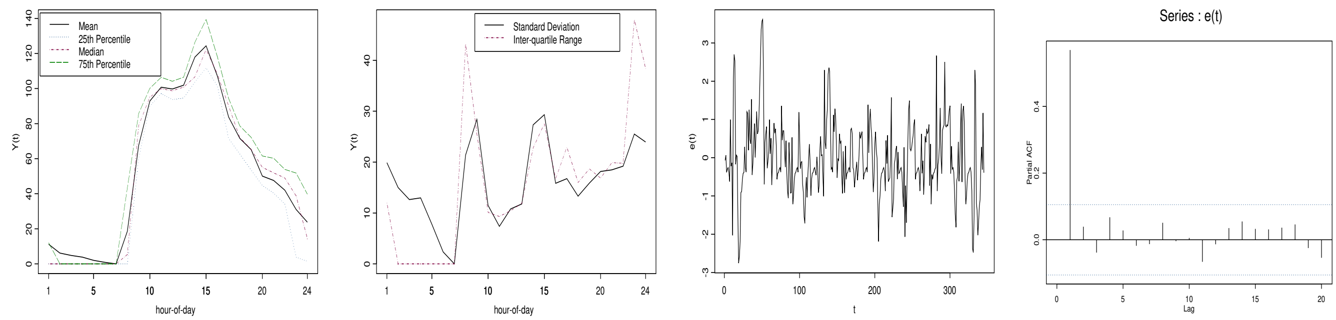
$$\hat{X}(t+1) = \hat{Y}^4(t+1).$$

Also, we can define the ϵ-tolerance prediction intervals for this point prediction as described in Section V-A.



(a) (b) (c) (d)

Fig. 1. Traffic load at AP 472: (a) time series (b) power spectrum (c) normal quantile plot for $X_{472}(t)$ (d) normal quantile plot for $Y(t)$.



(a) (b) (c) (d)

Fig. 2. (a)Changing patterns of mean, median, and quartiles of $Y(t)$. (b) Changing patterns of standard deviation (SD) and inter-quartile range of $Y(t)$. (c) Time series for $e(t)$. (d) Partial ACF plot for $e(t)$.

In general, our proposed time-series forecasting approach can be summarized as follows:

1) Transform the load in a reasonable way to make the data more normally distributed. Note that the transformation here is subjectively chosen, and it seems to be working well in the current application. We intend to work out a more automatic procedure to decide on the appropriate transformation in the future for better generalization.
2) Investigate time-varying patterns of the mean and variability of the transformed load.
3) Normalize the transformed load if the mean and variability are indeed time-varying.
4) Develop standard time series models like AR(p) for the normalized series, and employ rigorous model selection procedures like AIC to select the optimal model.
5) Perform one-step-ahead or multi-step-ahead forecasting on the normalized series using the fitted model, and then back-transform the forecast to the original scale.

V. Evaluation of the Performance of the Forecasting Algorithms

A. Metrics: prediction error ratio and percentage of correct predictions

To evaluate the performance of the prediction algorithms, we compute the *prediction error ratio* which is the ratio of the absolute difference of the predicted from the actual traffic over the actual traffic (r). For the prediction of the traffic of AP i at time t, the prediction error ratio $r(t)$ is defined as $r(t) = |Z_i^k(t) - X_i(t)|/X_i(t)$, for prediction algorithms $k = 1, 2, 3$. A perfect prediction algorithm has prediction error ratio equal to 0.

The prediction algorithms apply a predicted interval based on the historical mean and a tolerance (or precision) error level. Specifically, we define the *ϵ-tolerance prediction interval from a mean μ* to be the interval $[(1-\epsilon) * \mu, (1+\epsilon) * \mu]$. The prediction algorithm computes the percentage of times that the actual traffic is in the predicted interval. For example, in the case of the prediction P_k, $k = 1, 2, 3$, for the traffic of AP i during the h-th hour of day d, it computes the prediction interval

$$[(1-\epsilon) * Z_i^k(h,d), (1+\epsilon) * Z_i^k(h,d)],$$

and checks if $X_i(t)$ is in that interval.

A good prediction algorithm should have a high correct prediction percentage and low prediction error ratio. A large prediction error ratio indicates large prediction estimates and may result in conservative prediction and resource underutilization.

B. Forecasting using historical means and recent traffic (P1,P2, P3)

For all the aforementioned prediction algorithms, we computed the means based on the history for each AP. The history corresponds to three weeks of the trace, excluding weekends and starting on Monday, October 18th, 2004. We predict the traffic for each AP, for all the hours during the weekdays of the following week (Monday, November 8th until Friday, November 12th). We call this period *forecasting period*. For P3, we varied the recent history window size to be 2, 3, 4, and 5 hours. We evaluated P3 for various values of a,b, and c, including also

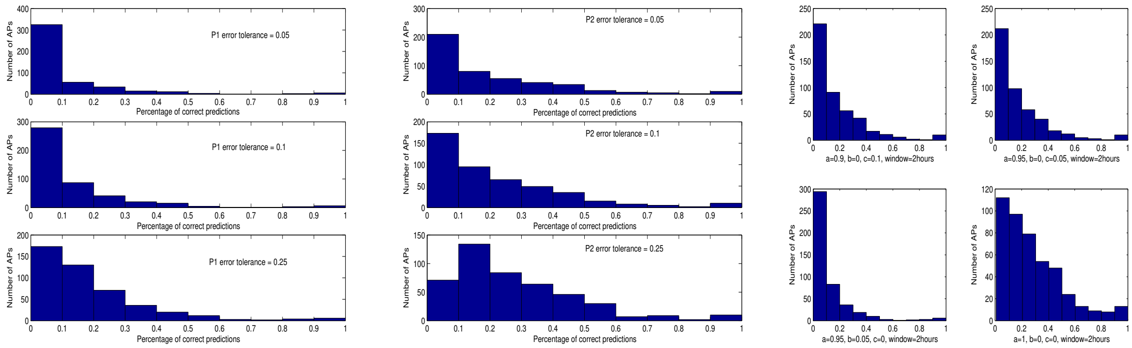


Fig. 3. Performance of prediction algorithms P1, P2, and P3 considering all APs. P3 has a 25% error tolerance.

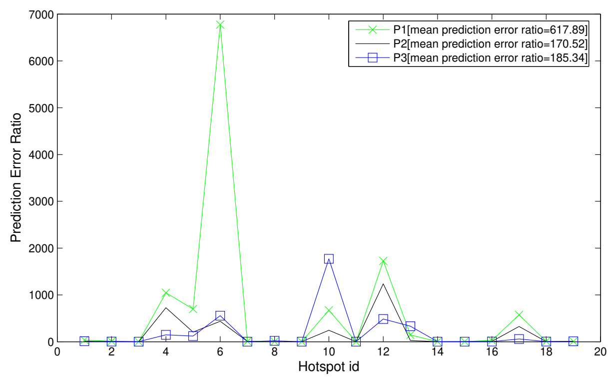


Fig. 4. Mean prediction ratios for P1, P2, and P3 with weights (a,b,c)=(1,0,0) for each hotspot.

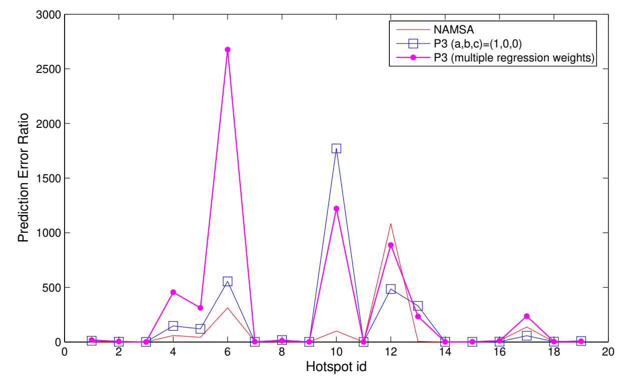


Fig. 6. Mean prediction ratio for the P3 and NAMSA forecasting algorithms for each hotspot.

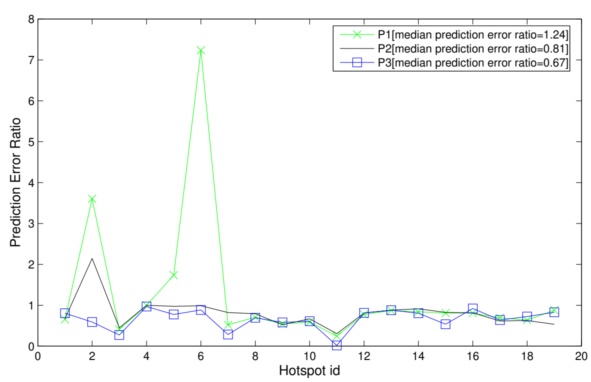


Fig. 5. Median prediction ratio for P1, P2 and P3 with weights (a,b,c)=(1,0,0) for each hotspot.

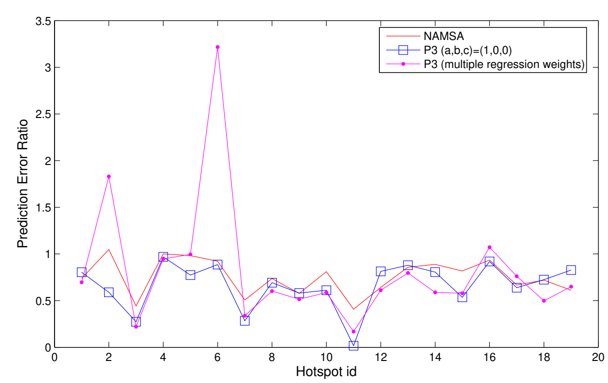


Fig. 7. Median prediction ratio for the P3 and NAMSA forecasting algorithms for each hotspot.

values resulted from applying multiple linear regression for each AP. Figures 3 show the histograms of the percentage of correct predictions for the P1, P2, and P3 considering all APs. P3 outperforms P2 and P1 with respect to the correct predictions percentage. P3 also outperforms P2 and P1 with respect to the correct predictions percentage, when we only consider the hotspots. Specifically, for a window of two hours and (a,b,c) equal to (1,0,0), P3's percentage of correct predictions for a 25%-tolerance prediction interval has a (mean, median, std. deviation) equal to (34.17%, 24.17%, 22.86%).

The *mean percentage of correct predictions* of hotspots for an ϵ-tolerance is the average of the percentages of correct predictions for that ϵ-tolerance considering all hotspots. The *mean prediction error ratio* of hotspots is the average of the mean prediction error ratios considering all hotspots. In the same manner, we compute their median and std. deviation. For the same ϵ-tolerance, P2 has a lower percentage of correct predictions than P3 but higher than P1 (for both median and mean prediction of correct percentages). Similarly, the median prediction error ratio for P3 is lower than for P1 and P2 (see Figure 5). On the other hand, P3's mean prediction error ratio is lower than P1's and higher than P2's one. The high mean prediction error ratio of P1, P2, and P3 are due to the high variability in the traffic.

C. Normalized ARIMA multi-step ahead time-series forecasting (NAMSA)

Using the same 3-week data (as in the other prediction algorithms), this normalized ARIMA multi-step ahead time series forecasting performs as follows. As Figures 6 and 7 illustrate, the prediction error ratio of the AP 472 (hotspot id 18) has a mean, median, and SD of 1.42, 0.72, and 3.77, respectively. Its correct percentages are 17.5%, 9.17%, and 6.67%, for a 25%, 10%, and 5%- tolerance prediction interval, respectively. The corresponding percentages for P1 are 20%, 10% and 6.67%, and for P2 20%, 18.33%, 16.67%, respectively. For a 25%-tolerance prediction interval, P3 with a two-hour window size and (a, b, c)=(1, 0, 0) has a 24.17% correct prediction percentage.

We apply the NAMSA algorithms (described in Section IV-D) to the 19 hotspots APs and the result is compared with the three aforementioned algorithm below. Note that the order of the AR(p) model is adaptively selected using AIC for each AP separately. Figures 6 and 7 illustrate the mean and median prediction error ratio of the P3 with weights (a,b,c)= (1,0,0), P3 with weights fitted using multiple linear regression, and NAMSA forecasting algorithm for all hotspots. Compared to the simple prediction algorithms P1, P2, and P3, the NAMSA algorithm results in better values for the mean and the SD of the error ratio (Figure 6). On the other hand, the median of its error ratio is a bit worse than that of the P3 algorithm (Figure 7). This forecasting algorithm is a *multi-step-ahead* forecasting. That is, to predict a value, apart from the traffic model, the multi-step-ahead forecasting uses the recent *predicted* values instead of the actual ones. This makes the prediction even harder than the one-step ahead forecasting that uses the *actual* recent values like P3. We expect better performance when we use this algorithm for one-step ahead forecasting.

Note that P3 with weights fitted using multiple linear regression performs worse than P3 and NAMSA (with respect to both mean and median error ratio). This is due to the difference in the metrics used: The prediction error ratio (as defined in Section V-A) is the ratio of the absolute difference of the predicted from the actual traffic over the actual traffic, whereas the multiple regression minimizes the square difference. When we use as metric the difference of the predicted from the actual traffic in square, we can observe that the mean of the overall improvement of P3 (with multiple regression) for hotspots reaches 26%. Furthermore, we found that the dominant regressor in the weighted sum of P3 is history (for all hotspots). Specifically, in average, the recent history predictor participates in P3 with a percentage of 43.8% while historical mean hour and historical mean hour-of-day percentages are 41.1% and 15.1%, respectively.

VI. RELATED WORK

There is only a small number of measurements studies that have examined the workload of 802.11 APs in production environments. In general, these studies have considered a wider range of issues, such as overall usage of a wireless infrastructure, and client mobility patterns, providing only a limited picture of the utilization of APs. Our work characterizes the workload of APs in a more systematic manner, and the results should have implications for the design of new wireless equipment and its evaluation.

Tang and Baker [14] used tcpdump traces and SNMP data to study a building WLAN with 12 access points and 74 users. Their only AP-specific results have to do with the variability in the maximum number users (between 3 and 12), and small number of handoffs (at most five within a five-minute period). Balazinska and Castro [4] used SNMP to characterize a much larger wireless network in three IBM buildings (177 APs). Their study examined the maximum number of simultaneous users per AP (mostly between 5 and 15), total load and throughput distributions. Two interesting observations found in this paper are that offered load and number of users are weakly correlated, and that user transfer rates are dependent on the location of the AP. Balachandran *et al.* [3] performed measurements in a three-day conference setting, also focusing on the offered network load and global AP utilization. They characterized wireless users and their workload and addressed the network capacity planning problem. The overall bursty behaviour and peaks and troughs are similar at all APs, though the absolute peak throughout at each AP varies. They observed that offered load is more sensitive to individual client traffic characteristics rather than just the total number of clients.

In an earlier study [6], we evaluated the performance of different caching paradigms in a wireless infrastructure. For example, we found tha unlike other measurement studies in wired networks in which 25% to 40% of documents draw 70% of web access, our traces indicate that 13% of unique URLs draws this number of web accesses.

Kotz *et al.* [9], [8] studied the wireless network at Dartmouth College using syslog, SNMP, and tcpdump traces. Their first study [9] reported the distribution of average daily traffic for 451 APs, which ranged from 39 MB to more than 2 GB, and observed that maximum daily traffic was far larger than the average daily traffic. In their follow-up study [8], they reported the average number of active cards per active AP per day (2-3 in 2001, and 6-7 in 2003/2004), and average daily traffic per AP by category (2-3 times higher in 2003/2004; twice or thrice more inbound than outbound traffic). A subset of the same data (syslog messages and tcpdump traces from 31 APs in 5 buildings) was revisited by Meng *et al.* [12] for flow modeling purposes. The authors proposed a two-tier (Weibull regression) model for the arrival of flows at APs and a Weibull model for flow residing times, and they also observed high spatial similarity within the same building. This paper makes a compelling case against Poisson modeling of wireless flows (at least for busy APs). They discovered that the flow size can be best approximated by a lognormal distribution.

Recently, Papagiannaki *et al.* [13] modeled the evolution of aggregated IP backbone traffic at large time scales, and developed long-term (up to 6-month ahead) forecasting models that can be used for capacity planning purposes. In particular, they analyzed eight inter-PoP aggregate demand time series from October 2000 to July 2002 with a granularity level of 90 minutes. Wavelet multi-resolution analysis (MRA) and ANOVA techniques are employed to pick out two dominant signals in the traces, the overall long-term trend and the 24-hour diurnal periodicity. Linear time series models are then proposed on the weekly-aggregated long-term trend and deviation to perform traffic forecasting.

There are some similarities between their study and our current work. Both studies make use of the daily/weekly periodicity in the traces, and propose time series forecasting models. However, our current work uses wireless traffic load measurement, and focuses on short (or middle) term forecasting (2-week ahead) on individual APs. As a result, there is no obvious long-term trend in our traces. In addition, our traces are much shorter (63-days) and less aggregated, which results in a much larger variability and a harder prediction task. Another difference is that we use the tolerance interval as a performance measure, which is usually much narrower than the variance-based prediction confidence interval. This explains partly the seemingly bad coverage property in Sections IV and IV-D.

VII. Conclusions and future work

We noticed in the APs's traffic some hours with unexpectedly low (compared to the historical means) values. In the current work, we proceed with the prediction without pre-processing these values. A more rigorous approach is to impute those entries with some estimates, such as the mean traffic load during the same hour-of-day from the other days. We expect that it will improve the prediction performances of the algorithms and plan to investigate this further. We intend to study more systematically the spatial correlations of APs and classify APs based on various parameters (e.g., traffic characteristics, building type, number of associations, and distinct clients). Furthermore, we aim to explore the impact of the above parameters and spatial correlations on forecasting.

The trace collection is still ongoing. We plan to investigate forecasting in various time-scales. Shorter-term forecasting (e.g., next minute) can assist in designing more energy-efficient clients. Long-term forecasting is essential for capacity planning and understanding the evolution of the wireless traffic and networks. For that, we will study the performance of MRA on a longer trace once it becomes available, and compare that with [13].

This research is a part of a comparative analysis study on wireless access patterns in various environments, such as a medical center, research institute, campus, and a public wireless network. We intend to analyze traces from testbeds in these environments and contrast their traffic models. We believe that understanding and forecasting the traffic of APs can have a dominant impact on the operation of wireless APs and clients and this study sets a direction for exploring further these issues.

Acknowledgment

This work was partially supported by the IBM Corporation under an IBM Faculty Award 2003/2004 grant.

References

[1] America's most connected campuses. http://forbes.com/home/lists/2004/10/20/04conncampland.html.

[2] F. Anjum, M. Elaoud, D. Famolari, A. Ghosh, R. Vaidyanathan, A. Dutta, P. Agrawa, T. Kodama, and Y. Katsube. Voice performance in WLan networksån experimental study. In *Proceedings of the IEEE Conference on Global Communications (GLOBECOM)*, Rio De Janeiro, Brazil, December 2003.

[3] Anand Balachandran, Geoffrey Voelker, Paramvir Bahl, and Venkat Rangan. Characterizing user behavior and network performance in a public wireless lan. In *Proceedings of the ACM Sigmetrics Conference on Measurement and Modeling of Computer Systems*, 2002.

[4] Magdalena Balazinska and Paul Castro. Characterizing mobility and network usage in a corporate wireless local-area network. In *First International Conference on Mobile Systems, Applications, and Services (iMobiSys)*, May 2003.

[5] P.F. Brockwell and R.A. Davis. *Time Series: Theory and Methods*. New York: Springer-Verlag, New York, 1998.

[6] Francisco Chinchilla, Mark Lindsey, and Maria Papadopouli. Analysis of wireless information locality and association patterns in a campus. In *Proceedings of the Conference on Computer Communications (IEEE Infocom)*, Hong Kong, March 2004.

[7] Mark Crovella and Azer Bestavros. Self-similarity in world wide web traffi c: Evidence and possible causes. In *Proceedings of SIGMETRICS '96*, 1996.

[8] Tristan Henderson, David Kotz, and Ilya Abyzov. The changing usage of a mature campuswide wireless network. In *ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, Philadelphia, September 2004.

[9] David Kotz and Kobby Essien. Analysis of a campus-wide wireless network. Technical Report TR2002-432, Dept. of Computer Science, Dartmouth College, September 2002.

[10] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson. On the self-similar nature of ethernet traffi c. *ACM Computer Communication Review*, 25(1):202–213, 1995.

[11] W. E. Leland, W. Willinger, M. S. Taqqu, and D. V. Wilson. Statistical analysis and stochastic modeling of self-similar datatraffi c. In *Proc. 14th Int. Teletraffi c Cong., 6-10*, volume 1, pages 319–328, Antibes Juan Les Pins, France, June 1994.

[12] Xiaoqiao Meng, Starsky Wong, Yuan Yuan, and Songwu Lu. Characterizing fbws in large wireless data networks. In *ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, pages 174–186, Philadelpia, 2004.

[13] K. Papagiannaki, N. Taft, Z. Zhang, and C. Diot. Long-term forecasting of internet backbone traffi c: Observations and initial models. In *Proceedings of the Conference on Computer Communications (IEEE Infocom)*, San Francisco,CA, April 2003.

[14] Diane Tang and Mary Baker. Analysis of a local-area wireless network. In *ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, pages 1–10, Boston, Massachusetts, USA, August 2000.

[15] W. Willinger, M. S. Taqqu, R. Sherman, and D. V. Wilson. Self-similarity through high-variability: Statistical analysis of ethernet lan traffi c at the source level. *ACM Computer Communication Review*, 25(4):100–113, October 1995.