

Research Article

Available Bandwidth Estimation in Network-Aware Applications for Wireless Campus e-Learning System

Mohd Faisal Ibrahim,¹ Masita Jamal,¹ Saadiah Yahya,¹ and Mohd Nasir Taib²

¹ Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, 40450 Shah Alam, Malaysia

² Faculty of Electrical Engineering, Universiti Teknologi MARA, 40450 Shah Alam, Malaysia

Correspondence should be addressed to Mohd Faisal Ibrahim, faisal@tmsk.uitm.edu.my

Received 30 May 2011; Accepted 30 September 2011

Academic Editor: Youyun Xu

Copyright © 2012 Mohd Faisal Ibrahim et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The constraint of a wireless network has motivated many researchers to develop network-aware applications that can dynamically adjust the users' demand based on network resources. For this to happen, applications need to have some mechanism that can estimate the network bandwidth by simply adjusting their behavior based on the collected network characteristics information. In the past, there have been several proposals that provide passive and active bandwidth estimation approaches for wired and wireless network. However, little effort has been spent to address the crucial issues of reliability and congestion control especially in a wireless network environment, which stay as a sticking point for the success of network-aware application. This paper focuses on providing accurate, low-intrusiveness, and fast-convergence time bandwidth estimation for network-aware application architecture. The experimental results validate the efficiency of the proposed solution in terms of accuracy, intrusiveness, and timelines.

1. Introduction

Wireless local area networks (WLANs) are becoming more widely used in homes, offices, public facilities, and university campuses. Recently, the focus is to establish Wi-Fi hotspots in convention centers, airports, shopping malls, hotels, public libraries, and cafes in which people can access email, download attachments, browse web sites or establish VPN connections to corporate networks while on the move. WLANs can be implemented as an extension or alternative for a wired LAN within a building or campus. As the number of hotspot locations is increasing, academic and industrial research have started to think of applications that can effectively utilize the network, which in return could satisfy their users' demands for reliable wireless Internet or web access. This trend has initiated the needs of network-aware applications that can dynamically adapt the users' demands to match the varying supply of network resources.

In order to make it possible for the user to use suitable application or download appropriate file based on the current bandwidth condition, the application must be aware of network resource availability which is also known as

network-aware application. With the exception of [1, 2], most of the approaches are directed mostly towards conceptual framework or have been pursued to address network-aware application issues for wired network. Furthermore, network-aware applications in wireless networks are difficult to develop [3] and implement due to the fact that bandwidth estimation techniques can be affected by reception signal strength [4], path loss [4], fading [4], interference [5], resource sharing [5, 6], and fluctuating nature [6–8]. This results in imprecise estimates with high convergence times and intrusiveness.

In this paper, we incorporate active available bandwidth estimation technique to our proposed network-aware application architecture for e-learning system. To address accuracy, convergence, and intrusiveness issues, we employ WBest algorithm to deal with the significant quality fluctuations, performance degradation, and weak connectivity of wireless network.

The rest of the paper is structured as follows. Section 2 describes bandwidth estimation, packet dispersion techniques, WBest algorithm, and related works. Section 3 explains the architecture of NApEL, and Section 4 clarifies the

goals and usage scenario of NApEL. Section 5 examines the experimental results. Finally, we briefly discuss future work and conclude in Section 6.

2. Preliminaries

2.1. Bandwidth Estimation. End-to-end measurement of bandwidth-related metrics is referred to as bandwidth estimation. Its features consist of capacity, available bandwidth, and bulk TCP transfer capacity. Within this paper, the term available bandwidth (AB) refers to the maximum unused bandwidth at a link or end-to-end path in a network. AB depends on the link capacity and the traffic load during a certain time period. In a given hop i the available bandwidth A_i of end-to-end link during time interval $(t; t + \tau)$ is given by the unutilized part of the capacity:

$$A_i(t, t + \tau) = \frac{1}{\tau} C_i \int_t^{t+\tau} [1 - \lambda_i(t)] dt, \quad (1)$$

where λ_i is the average utilization of hop i in the given time interval, and C_i is the capacity of hop i . By considering the available bandwidth definition to an n number of hops in a path the available bandwidth of the end-to-end path A will be the minimum available bandwidth of all n hops:

$$A(t, t + \tau) = \min_{i=1 \dots n} \{A_i(t, t + \tau)\}. \quad (2)$$

The term tight link of a network path is employed to illustrate the minimum available bandwidth along the path and the term narrow link is used to portray the minimum capacity of the end-to-end path.

2.2. Packet Dispersion Techniques. There has been several bandwidth estimation techniques such as packet dispersion, probe gap model, and self-loading periodic streams aimed at measuring the end-to-end available bandwidth of a network path. Since in our solution we employ WBest algorithm [4], packet dispersion techniques will be discussed in detail in this section.

Figure 1 illustrates the basic concept of packet dispersion. Packet train probing sends packet train of length N with the same back-to-back packets of size L to travel into the network. After the packet traverses the narrow link C_i , the receiver measures the time spacing between the arrival of the first packet and the arrival of the last packet as follows:

$$b(N) = (N - 1) \times \frac{L}{\Delta(N)}, \quad (3)$$

where $b(N)$ is the bandwidth estimate, N is the length of the packet train, L is the size of each packet, and $\Delta(N)$ is the difference in time between (dispersion) the first packet and the final packet of the train.

The probe packet size can significantly affect the performance of the bandwidth estimation [9]. Sending larger packets will lead to a wider dispersion and can cause additional traffic. However, if packets are small, the estimation tends to be overrating because of shorter dispersion times [10].

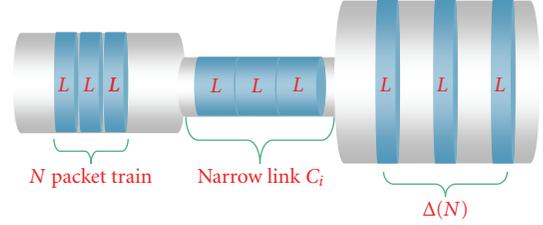


FIGURE 1: Illustration of bandwidth estimation using packet train probing.

- (1) Send n packet pairs to client
- (2) $C_e \leftarrow \text{median}(C_i), i = 1, \dots, n$
- (3) Send m packet train at rate C_e to Client
- (4) $R \leftarrow L / \text{mean}(T_i), (i = 1, \dots, m - 1)$
- (5) **If** $R \geq C_e/2$ then $AB \leftarrow C_e [2 - C_e/R]$
- (6) **Else** $AB \leftarrow 0$
- (7) $p \leftarrow$ packet loss rate in train
- (8) **if** $p > 0$ then $AB \leftarrow AB \times (1 - p)$

ALGORITHM 1: WBest algorithm.

2.3. WBest Algorithm. This section briefly describes the WBest algorithm that estimates the available bandwidth of an end-to-end path. A more detail information regarding WBest can be found in [4]. The WBest algorithm, summarized in Algorithm 1, has two phases: effective capacity measurement phase and available bandwidth estimation phase. In the first phase, a packet pair technique estimates the effective capacity of the wireless network. In the second phase a packet train scheme determines achievable throughput and infers available bandwidth.

2.3.1. Effective Capacity Measurement Phase. During the effective capacity measurement phase, the algorithm sends n packet pairs to estimate the effective capacity C_e . In order to minimize the impact of crossing and contending traffic, the median of the n packet pair capacity estimates is used to approximate C_e in the estimation time period:

$$C_e = \text{median}(C_i), \quad i = 1, \dots, n. \quad (4)$$

This will allow the algorithm to capture the impact of rate adaptation measurements of effective capacity.

2.3.2. Available Bandwidth Estimation Phase. For the second phase of the algorithm (lines (3–8)), the available bandwidth is estimated by sending m packet train at rate C_e . The available bandwidth AB can be derived by using the following equation:

$$AB \leftarrow C_e \left[2 - \frac{C_e}{R} \right], \quad (5)$$

where R is the average dispersion rate at the receiver.

For a wireless network, achievable throughput is the average dispersion rate at the receiver for a probing rate of C_e [11]. If the achievable throughput is greater or equal to half of C_e then the AB can be calculated by using (5), otherwise the AB is zero. A suitable pair of packet pairs and trains will be removed from the computation once WBest algorithm identifies a packet loss in order to reduce estimation errors and convergence time. Loss rate p will be recorded in each packet train to reduce the available bandwidth estimation (lines (7–8) of Algorithm 1).

2.4. Related Work. This section describes previous work in bandwidth estimation and network-aware application. Research in these two areas has become increasingly widespread in recent years. As our work stresses the need for a web application to be sensitive to the dynamic network conditions, some of the previous works have concentrated mainly on bandwidth estimation algorithms. The efforts have been strongly oriented towards either proposing improvements to bandwidth estimation algorithm specific to wired or wireless networks.

For example, pathload [12], TOPP [13, 14], pathChirp [15], and PTR [16] present the end-to-end AB estimations for wired networks. They are based on self-loading techniques, including the Self-loading Periodic Streams (SLoPS) [12] and Train of Packet Pairs (TOPP) [13, 14], which applies multiple rate traffic, linearly increased probing rate, or exponentially increased probing rate. In SLoPS, if a flow is inserted that induces a temporal congestion in the bottleneck link of the path, the input and the output probing packet rate can be compared in order to find the ABW.

Other methods such as IGI [16], Spruce [17], and Delphi [18], use Probe Gap Model (PGM) to estimate the AB. This technique calculates the cross-traffic rate by measuring the time gap between the arrivals of two successive probes at the receiver and then estimates the AB indirectly.

However, these probe-based techniques provide unreliable bandwidth estimation for wireless network [19, 20], due to the characteristics of wireless networks such as contention, link layer retransmissions, fading, and interference. Furthermore, these estimation methods presume that they are dealing with an unvarying characteristic of wired networks, which does not apply to wireless networks due to the shared physical medium, unreliable nature of the transmission medium, and the multiaccess coordination function.

Nevertheless, all these existing bandwidth estimation techniques have been the basis for many wireless bandwidth estimators such as SenProbe [21], AdhocProbe [22, 23], WBest [4], DietTOPP [9], and SLOT [24]. AdhocProbe and SenProbe are an end-to-end path capacity estimation tools for ad hoc and wireless sensor networks.

SLOT, combines two packet probing techniques which are SLoPS and TOPP. The estimated AB is obtained by implementing two different probing strategies which are linear (incremental) and binary (dichotomy).

Another method, DietTOPP, determines the AB by modifying and reducing TOPP search algorithm. Although this

technique improves accuracy, it does not consider convergence time and intrusiveness.

WBest measures the effective capacity and available bandwidth of wireless networks by utilizing packet dispersion techniques. It uses a two-step estimation algorithm: (1) the effective capacity estimation based on a packet pair technique and (2) the achievable throughput and available bandwidth estimation based on a packet train technique.

A lot of effort has been spent in the field of network-aware application to address the issues of heterogeneity and bandwidth variations of pervasive web environment. Some of the works such as Bolliger and T. Gross [25], Bates et al. [26], and Tierney et al. [27] were aimed towards architecture or conceptual framework of network-aware application. Transend [28] is a proxy-based architecture, employing so-called distillation services, that automatically converts image formats (e.g., from TIFF format to JPEG) or adjusts the image by reducing the resolution or color depth based on a given client's network bandwidth. Odyssey [1] provides adaptation for mobile information access application that adapts to changing network environments. It was designed to cope with network heterogeneity and dynamic changes of wireless environments [29]. However, the adaptation strategy in Odyssey is not suitable for Internet-based hosts and the approach is towards passive measurements rather than active Mobeware [2], which is based on CORBA and Java distributed object technology, and is a middleware toolkit that supports the adaptation of applications to varying mobile network conditions.

Although Odyssey and Mobeware have shown promise in providing a network-aware application solution that can adapt to mobile devices' network constraints, they have not taken into consideration the tradeoffs of bandwidth estimation approach with respect to timeliness, accuracy, and overhead in real systems. The usefulness of bandwidth estimator for the needs of network-aware applications rests on the accuracy of the information. This is important because accurate AB estimation will be very useful especially for applications that have critical response time requirements.

If an application wants to adjust to network changes by adapting the content, it is important to make sure that the AB estimation is accurate. The intrusiveness and convergence time required for bandwidth estimation process are also important because the time taken to estimate the AB and the probing packet sent to the network traffic should be low [30].

3. Architecture—NAPEL

In this section we elaborate on the architectural design of the network-aware applications for E-Learning System (NAPEL). The term “network-aware applications” refers to the capability to monitor network conditions such as available bandwidth, network capacity, packet loss rate, and delay that can be used by the application to adjust their behavior to best utilize the network resources availability [25, 31]. It can be divided into two parts which are network awareness and network adaptation. Network awareness implies the capability of the web application to monitor network conditions and

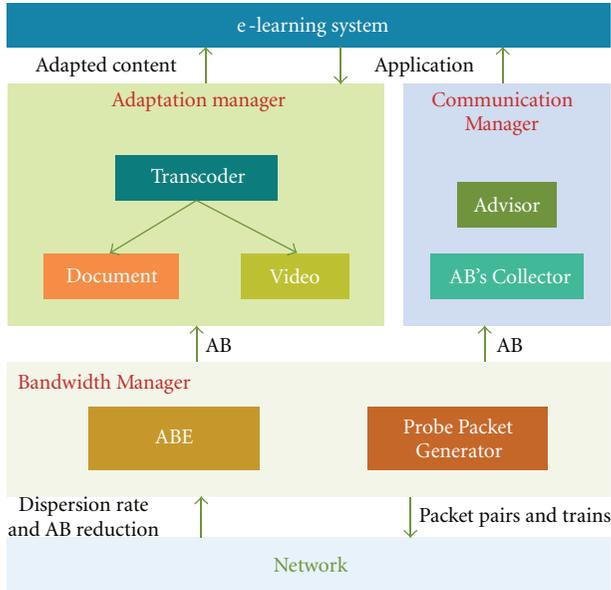


FIGURE 2: Architectural overview of Network-Aware Application for e-Learning System (NApEL).

network adaptation refers to the ability of utilizing network resources and optimizing content delivery.

The architecture of NApEL is illustrated in Figure 2, contains three important components which are Adaptation Manager, Communication Manager, and Bandwidth Manager. These components are explained in what follows in more detail.

3.1. Bandwidth Manager. This section presents an overview of the Bandwidth Manager and its two components, the Probe Packet Generator (PPGen) and Available Bandwidth Estimator (ABE). We will explain how PPGen sends network probes to capture raw estimation data and then how ABE collects these data to estimate available bandwidth. We have customized WBest in order to deploy ABE in NApEL. To automate the bandwidth estimation loops, we added the loop command *for (Loop=0; loop < MaxLoop; loop++)* in *wbest_snd.c*. This instruction will get the maximum number of loops *MaxLoop* as well as the IP address (*hostname*) of the sender node *SND* from a text file. Then *SND* starts to establish connection with the receiver node *RCV* by executing this command: `./wbest_rcv -s %s", hostname`. To change the packet probe size, we altered the *i_PktNumbPP* and *i_PktNumbPT* variables in *wbest_snd.c* file. The main idea of ABE is to supply information regarding the AB to users or applications in a database server which can be accessed by the web server. Therefore, we have modified some parts of the original code to allow AB estimation results to be stored in the database.

3.1.1. Available Bandwidth Estimator (ABE). The main function of ABE is to provide the information regarding the end-to-end available bandwidth estimation. ABE will use the average dispersion rate R and the available bandwidth

reduction S values obtained from PPGen to estimate available bandwidth.

3.1.2. Probe Packet Generator (PPGen). PPGen is responsible for sending at an effective capacity (C_e) rate, a packet train of length m to estimate available bandwidth. C_e is defined as follows:

$$C_e = \frac{\int_{t_0}^{t_1} (L/T(t)) dt}{t_1 - t_0}, \quad (6)$$

where $T(t)$ is the dispersion time t between packets and L is the packet size.

3.2. Communication Manager

3.2.1. Advisor. The purpose of Advisor in Communication Manager is to give some suggestions on suitable communication tool to be used by users based on network condition. A dialog box will come up on user's screen whenever Advisor's recommendation is invoked. If the available bandwidth is low, a user can decide whether to still choose the same communication tool or not.

3.2.2. AB's Collector. The function of AB's Collector is to collect all information concerning each AP's network condition by performing a system call to periodically obtain up-to-date bandwidth information

3.3. Adaptation Manager. When Adaptation Manager receives available bandwidth information from Bandwidth Manager, it simply calls the transcoder to transcode the content and generates the adapted content. An open-source FFMPEG tool version 0.8.2 is used for transcoding and it is running under a Windows XP machine. Currently, the transcoding process is being implemented by utilizing the knowledge of the network characteristics only.

4. Goals and Usage Scenario of NApEL in Campus Learning Environment

The NApEL project has been developed at the faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, in order to provide a prototype which could demonstrate how the architecture can be implemented. With the intention of introducing the goals of NApEL, in this section we present the scenario to describe the requirements of network-aware applications and demonstrate an application of the scenario.

Melissa is working on a class project together with her other friends Sam and Natasha. All of them agree to meet at the cafeteria for a quick project discussion at 10 am. Upon arrival, she discovered that her friends have not yet arrived. So, she grabs her notebook and connects to NApEL through access point 1. She needs to check the status of "who is online now" (Figure 3) to see whether Sam and Natasha are connecting to NApEL because she suspects that both of

The screenshot shows the NApEL user interface. At the top right, it says "Logged in as Melissa" and "Monday 31st of January 2011" with a "Logout" link. The main content area is divided into two columns. The left column, titled "My Corner", contains links for "Profile", "User Preference", "Reset Password", and "Course Registration", followed by an "Instant Messaging" section with a "Message" input field and a "Shout it!" button. Below this are three message entries: "Naszuan - test" (2011-01-30 21:01:54), "Melissa - How to setup user preference" (2010-12-10 12:04:41), and "Admin - Welcome to NApEL" (2010-11-16 10:04:13). The right column, titled "Quick Links", contains a table with columns "# COURSES", "GROUP", "Announcement", "i-Discuss", and "myDrawer". The table lists three courses: "CSC125 - Fundamental of Computer Problem Solving", "MAT149 - Calculus I", and "CSC175 - Object Oriented Programming I". Below the table is a "Who's online now?" section showing three users: Naszuan, Sam, and Natasha, each with a small profile picture and a Skype status icon.

FIGURE 3: NApEL user interface. We have integrated Skype into NApEL. Each Skype user on a PC, notebook, or PDA can send messages to other Skype users and check online or offline presence of these users in real time.

The screenshot shows the NApEL user interface with an "ADVISOR" dialog box overlaid. The dialog box has a blue header "ADVISOR" and contains the text "Currently you are using Access Point 1". Below this is a table with columns "Access Point", "1", "2", and "3". The table has two rows: "AB (kbps)" and "Recommendation". The "AB (kbps)" row shows values 29.0, 930, and 259. The "Recommendation" row shows values "TM", "Voice, TM", and "Voice, TM". Below the table are links for "Cancel" and "Proceed >". At the bottom of the dialog box, it says "AB = Available bandwidth" and "TM = Text Messaging".

FIGURE 4: NApEL provides a mechanism for detecting insufficient bandwidth and providing recommendation on suitable communication tools to be used based on available bandwidth.

them forget about this meeting. When she opens NApEL, she sees that both of her friends are online now. She decided to contact Natasha using VOIP. Unfortunately, bandwidth is miserable because many students at the cafeteria are surfing the web using AP1. NApEL observes that at the current bandwidth Melissa will not be able to communicate with Natasha smoothly. By checking other APs nearby, NApEL discovers that wireless bandwidth is excellent at AP2, and if Melissa still wants to communicate through AP1 she can still use text messaging. A dialog box (Figure 4) pops up on Melissa's screen suggesting that she can still use AP1 if she wants to communicate through text messaging or go to AP2 if she still prefers VOIP. Melissa accepts NApEL's advice and

connects her notebook to AP2. She realizes that the available bandwidth at AP2 is more than 900 Kbps. She manages to contact Natasha and ask her to quickly come to the cafeteria.

5. Experimental Design and Results

The goal of our experiments is to quantify how effectively NApEL platforms are able to track the dynamic rate adaptation of highly varying characteristics of wireless network in a timely and accurate manner. In addition, we quantify the effectiveness of the platforms under reproducible and controllable conditions for different traffic loads in order

to ensure a fair comparison. We describe the network testbeds, the evaluation metrics, the experimental setup, and validation methodology used for our experiments. We have performed several experiments to validate our proposed method. The first aims at comparing the ABE result with passive monitoring tools. The second experiment is to determine whether the fleet of packet trains sent will stress the network. The third experiment is to evaluate how long does it take for ABE to have the estimation.

5.1. Performance Metrics. The usefulness of bandwidth estimation algorithm for network-aware adaptation relies on the accuracy of the estimation, the convergence time, and intrusiveness. In order to measure the accurateness of the bandwidth estimation, ABE error is computed as the difference between the actual bandwidth AB measured using passive monitoring technique and the estimated AB. We also used relative error given below to evaluate the accurateness of the AB:

$$\text{Rel}_{\text{Error}} = \frac{\text{AB}_{\text{estimated}} - \text{AB}_{\text{actual}}}{\text{AB}_{\text{actual}}}, \quad (7)$$

where $\text{AB}_{\text{estimated}}$ is the available bandwidth estimates generated by ABE, and $\text{AB}_{\text{actual}}$ is the actual available bandwidth detected by WRM.

In the evaluation of ABE, one of the main interests is intrusiveness. The intrusiveness of bandwidth estimation expresses to what extent its heavy packets probing affects the existing network traffic. In other words, low intrusiveness is vital for decreasing the impact initiated by the probing traffic. We defined intrusiveness as follows:

$$\text{Intrusive}_{\text{BEM}(\%)} = \frac{T_{\text{ABE}}}{\text{AB}} \times 100, \quad (8)$$

where AB is the actual bandwidth available identified by WRM and T_{ABE} is the amount of traffics (in bits) sent by ABE during probing time.

Convergence time is the total estimation time used by ABE to complete an estimation.

5.2. Experimental Setup. We verify the measured available bandwidth by comparing it with WRM [32] and by measuring how ABE responds to induce changes in available bandwidth. Both ABE and WRM use the same machines.

It is essential to use test traffic that closely simulates traffic on real networks and reproduces its most critical characteristics. In our study we conducted two series of laboratory experiments using Nuttcp to generate and send different volume of contending traffic to the network. The dynamic traffic conditions that typify actual wireless links are being replicated under varying levels of traffic load from 0%, 25%, 50%, and 75%.

5.3. Testbed Topology. A laboratory testbed was set up in order to evaluate the ability and efficiency of the proposed ABE to compute the available bandwidth. The testbed topology is shown in Figure 5. It contains two PCs (PC1 and PC2)

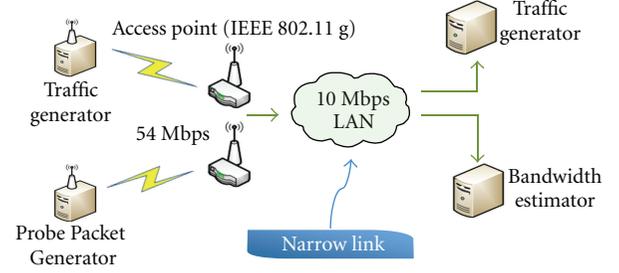


FIGURE 5: 10 Mbps single hop tight link.

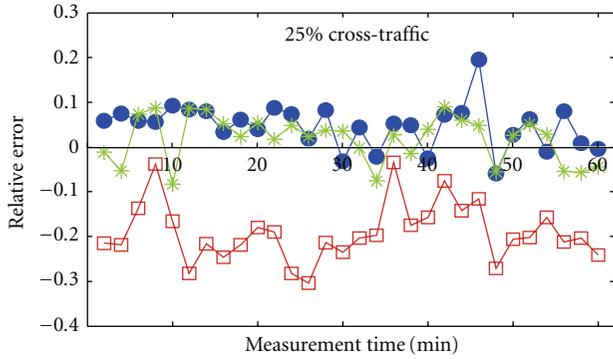
running Nuttcp traffic generator to produce controllable synthetic cross-traffic and two Dell Dimension PCs (PC3 and PC4) running under Ubuntu 7.08 which are used as the sender and the receiver to generate probe traffic. The stream which originated in PC5 is sent to PC6 and PC5 is wirelessly connected to the AP by means of a 54 Mbps WiFi segment and PC6 is directly connected using a 10BaseT network connection to the same AP.

5.4. Experimental Results

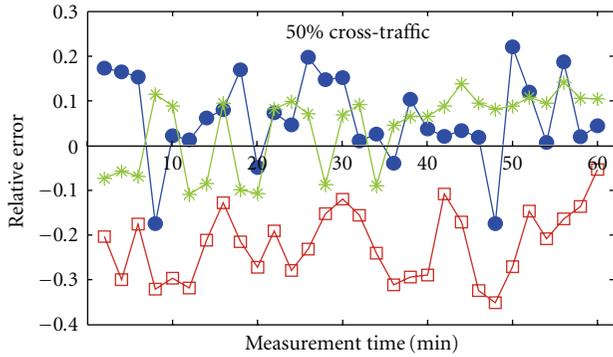
5.4.1. Accuracy. From Figure 6, we observe that as we decrease the packet pairs and trains to 15, the relative error are higher for 25%, 50%, and 75% cross traffic. Packet pairs and trains of 30 and 45 provide more accurate estimation. We realize that the estimation algorithm depends on the size of packet pairs and trains as what have been discovered by [33]. As the number of packets decreases, accuracy will also decrease.

5.4.2. Intrusiveness. The intrusiveness of ABE is assessed on its capacity of generating a result while transferring the minimum amount of additional traffic. The main aim of this experiment is to determine whether the total amount of probing bytes generated by ABE causes a considerable decline in the AB. Figure 7 shows the results of a 60-minute experiment, performed similarly with the experiment 1. As expected, sending 15 packet pairs and trains will make the ABE intrusiveness decreases. When we used 30 and 45 packet pairs and trains, the intrusiveness was between 5 to 17% for all traffic loads. This indicates that even if we use 30 or 45 packet pairs and trains, ABE can still be considered as nonintrusive because the average probing traffic rate does not encompass a substantial load on the network during the measurement process. This result is in agreement with Prasad et al.'s [34] finding which suggests that ABE is noninvasive.

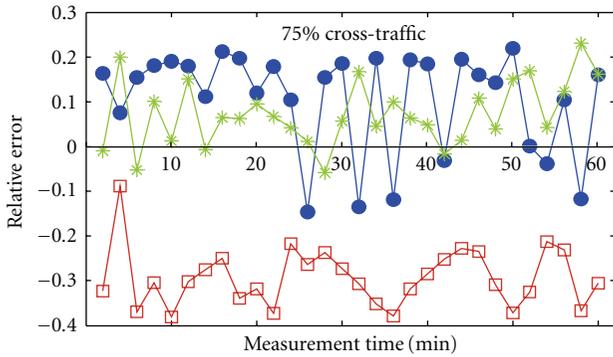
5.4.3. Convergence Time. Figure 8 depicts, respectively, the result of convergence time for each traffic load with packet pairs and trains of 15, 30, and 45. The results show that on average if we use 30 packet pairs and trains, we can effectively achieve the convergence time between 0.3 to 0.4 seconds. ABE requires a short convergence time due to the fact that WBest algorithm estimates the available bandwidth without using searching algorithms [4]. Furthermore, according to



(a)



(b)



Number of packet pairs and trains

- 15
- 30
- *— 45

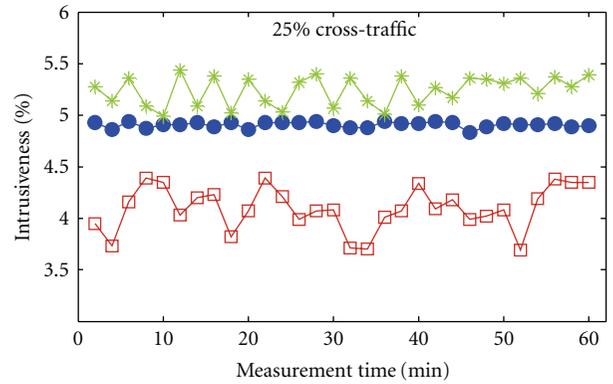
(c)

FIGURE 6: Available bandwidth estimation error for (a) 25%, (b) 50%, and (c) 75% cross-traffic.

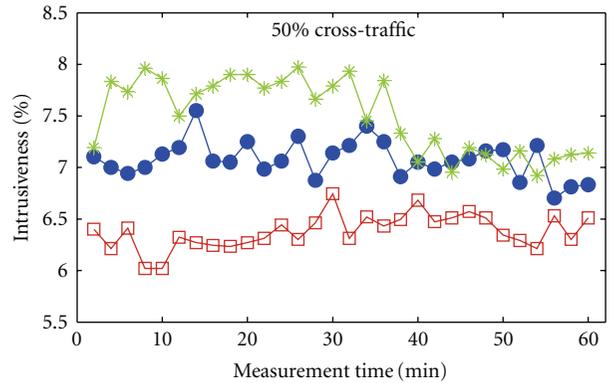
[4] this algorithm does not rely on delay measurements to measure the AB which in return will make it more robust and yield accurate results with lower convergence times and intrusiveness.

6. Conclusion

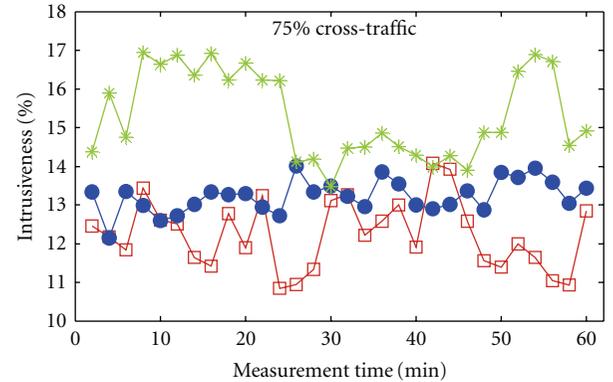
This paper proposes the NAPeL architecture for wireless campus e-learning system that includes an end-to-end



(a)



(b)



Number of packet pairs and trains

- 15
- 30
- *— 45

(c)

FIGURE 7: Intrusiveness for (a) 25%, (b) 50%, and (c) 75% cross-traffic.

available bandwidth estimation module called ABE. It allows bandwidth estimator algorithm to be executed separately from applications, tracks the network bandwidth, and then sends this information to web applications. We have conducted several experiments to estimate the effectiveness of available bandwidth based on accuracy, intrusiveness, and convergence time by investigating how it reacts when the cross traffic is present on the single hop narrow links. Based

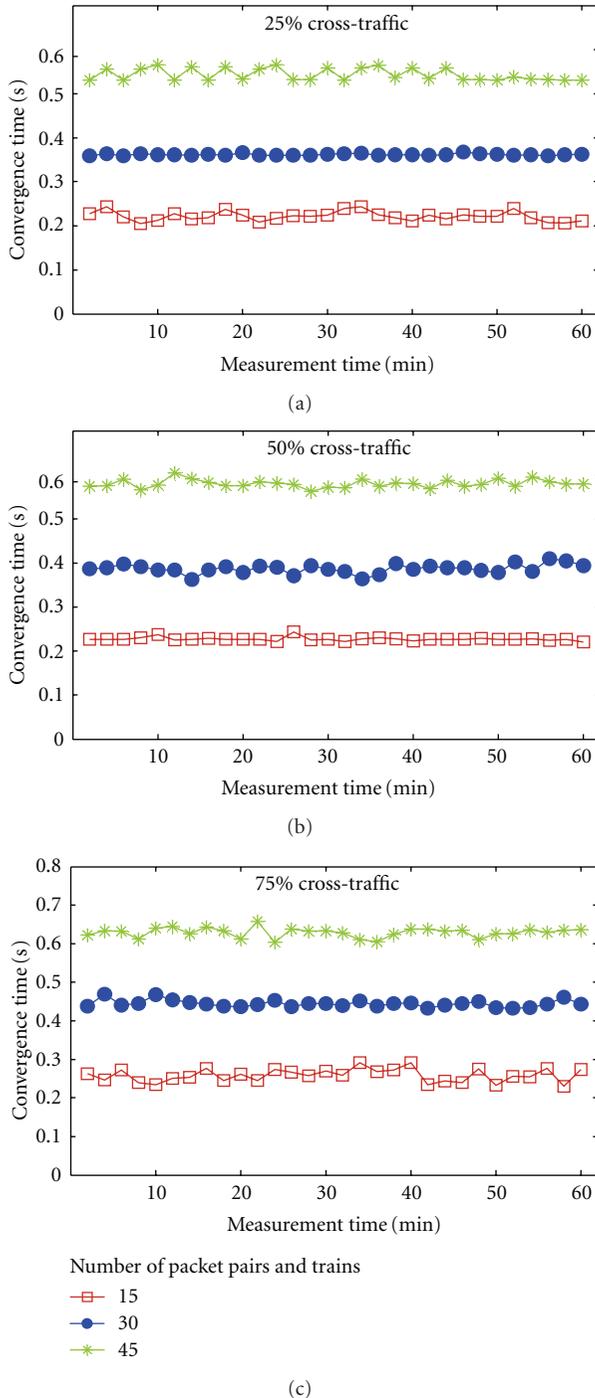


FIGURE 8: Convergence time for (a) 25%, (b) 50%, and (c) 75% cross-traffic.

on the experiments, the estimations are very close to the passive measurement with low probe packets generation and probing time.

Future research activity will aim at providing adapted content not only according to the network resources constraints but also device capabilities limitations. The limitation of mobile devices such as processing power, screen size, small memory, and storage should be considered when

developing and implementing network-aware applications so that appropriate content can be delivered and presented in any heterogeneous devices.

Finally, it is interesting to notice that the network-aware adaptation approach described in this paper could further strengthen the advantages and benefits of network-aware application over wireless campus environment in terms of the utilization of mobile application, multimedia content, and network resources. One of the main reasons of deploying this approach is to improve users' productivity by optimizing the utilization of resources. Our work here is towards providing the wireless campus communities to access multimedia content anywhere, anytime, and on any device.

References

- [1] B. D. Noble, M. Satyanarayanan, D. Narayanan, J. E. Tilton, J. Flinn, and K. R. Walker, "Agile application-aware adaptation for mobility," in *Proceedings of the 16th ACM Symposium on Operating Systems and Principles*, pp. 276–287, 1997.
- [2] O. Angin, A. T. Campbell, M. E. Kounavis, and R. R. F. Liao, "The mobiware toolkit: programmable support for adaptive mobile networking," *IEEE Personal Communications*, vol. 5, no. 4, pp. 32–43, 1998.
- [3] M. Kim and B. Noble, "Mobile network estimation," in *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking*, pp. 298–309, July 2001.
- [4] M. Li, M. Claypool, and R. Kinicki, "WBest: a bandwidth estimation tool for IEEE 802.11 wireless networks," in *Proceedings of the 33rd IEEE Conference on Local Computer Networks (LCN '08)*, pp. 374–381, October 2008.
- [5] H. Zhao, E. Garcia-Palacios, J. Wei, and Y. Xi, "Accurate available bandwidth estimation in IEEE 802.11-based ad hoc networks," *Computer Communications*, vol. 32, no. 6, pp. 1050–1057, 2009.
- [6] S. Tursunova, K. Inoyatov, and Y. T. Kim, "Cognitive passive estimation of available bandwidth (cPEAB) in overlapped IEEE 802.11 WiFi WLANs," in *Proceedings of the 12th IEEE/IFIP Network Operations and Management Symposium (NOMS '10)*, pp. 448–454, April 2010.
- [7] T. Nadeem, "Network-aware applications in wireless networks," in *Proceedings of the 2nd Workshop on Research Directions in Situational-aware Self-managed Proactive Computing in Wireless Adhoc Networks*, 2009.
- [8] K. Lee, "Adaptive network support for mobile multimedia," in *Proceedings of the 1st Annual International Conference on Mobile Computing and Networking*, pp. 62–74, November 1995.
- [9] A. Johnsson, B. Melander, and M. Björkman, "Bandwidth measurement in wireless networks," in *Proceedings of the 4th Annual Mediterranean Ad Hoc Networking Workshop*, 2005.
- [10] C. Dovrolis, P. Ramanathan, and D. Moore, "Packet-dispersion techniques and a capacity-estimation methodology," *IEEE/ACM Transactions on Networking*, vol. 12, no. 6, pp. 963–977, 2004.
- [11] M. Li, M. Claypool, and R. Kinicki, "Modeling and simulating packet dispersion in wireless 802.11 networks," Tech. Rep. WPI-CS-TR-06-03, Computer Science Department at Worcester Polytechnic Institute, 2006.
- [12] M. Jain and C. Dovrolis, "End-to-end available bandwidth: measurement methodology, dynamics, and relation with TCP

- throughput,” *IEEE/ACM Transactions on Networking*, vol. 11, no. 4, pp. 537–549, 2003.
- [13] B. Melander, M. Bjorkman, and P. Gunningberg, “A new end-to-end probing and analysis method for estimating bandwidth bottlenecks,” in *Proceedings of IEEE GLOBECOM—Global Internet Symposium*, pp. 415–420, 2000.
- [14] B. Melander, M. Bjorkman, and P. Gunningberg, “Regression-based available bandwidth measurements,” in *Proceedings of the International Symposium on Performance Evaluation of Computer and Telecommunications Systems*, 2002.
- [15] V. Ribeiro, R. Riedi, R. Baraniuk, J. Navratil, and L. Cottrell, “PathChirp: efficient available bandwidth estimation for network paths,” in *Proceedings of the Passive and Active Measurements Workshop*, 2003.
- [16] N. Hu and P. Steenkiste, “Evaluation and characterization of available bandwidth probing techniques,” *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 6, pp. 879–894, 2003.
- [17] J. Strauss, D. Katabi, and F. Kaashoek, “A measurement study of available bandwidth estimation tools,” in *Proceedings of the ACM SIGCOMM Internet Measurement Conference (IMC ’03)*, pp. 39–44, October 2003.
- [18] V. Ribeiro, M. Coates, R. Riedi, S. Sarvotham, B. Hendricks, and R. Baraniuk, “Multifractal cross-traffic estimation,” in *Proceedings of the ITC Conference on IP Traffic, Modeling and Management*, 2000.
- [19] K. Lakshminarayanan, V. N. Padmanabhan, and J. Padhye, “Bandwidth estimation in broadband access networks,” in *Proceedings of the ACM SIGCOMM Internet Measurement Conference (IMC ’04)*, pp. 314–321, October 2004.
- [20] D. Gupta, D. Wu, P. Mohapatra, and C. N. Chuah, “Experimental comparison of bandwidth estimation tools for wireless mesh networks,” in *Proceedings of the 28th IEEE Conference on Computer Communications (INFOCOM ’09)*, pp. 2891–2895, April 2009.
- [21] T. Sun, L. J. Chen, G. Yang, M. Y. Sanadidi, and M. Gerla, “SenProbe: path capacity estimation in wireless sensor networks,” in *Proceedings of the 3rd International Workshop on Measurement, Modelling, and Performance Analysis of Wireless Sensor Networks*, 2005.
- [22] L. J. Chen, T. Sun, G. Yang, M. Y. Sanadidi, and M. Gerla, “AdHoc probe: path capacity probing in wireless ad hoc networks,” in *Proceedings of the 1st International Conference on Wireless Internet (WICON ’05)*, pp. 156–163, July 2005.
- [23] L. J. Chen, T. Sun, G. Yang, M. Y. Sanadidi, and M. Gerla, “AdHoc Probe: end-to-end capacity probing in wireless ad hoc networks,” *Wireless Networks*, vol. 15, no. 1, pp. 111–126, 2009.
- [24] A. Amamra and K. M. Hou, “Bandwidth estimation in broadband access networks,” in *Proceedings of the 10th International Conference on Computer Modeling and Simulation*, pp. 46–51, 2008.
- [25] J. Bolliger and T. Gross, “A framework-based approach to the development of network-aware applications,” *IEEE Transactions on Software Engineering*, vol. 24, no. 5, pp. 376–390, 1998.
- [26] J. Bates, D. Halls, and J. Bacon, “A framework to support mobile users of multimedia applications,” *Mobile Networks and Applications*, vol. 1, no. 4, pp. 409–419, 1996.
- [27] B. L. Tierney, D. Gunter, J. Lee, M. Stoufer, and J. B. Evans, “Enabling network-aware applications,” in *Proceedings of the 10th IEEE International Symposium on High Performance Distributed Computing*, pp. 281–288, August 2001.
- [28] A. Fox, S. D. Gribble, E. A. Brewer, and E. Amir, “Adapting to network and client variability via on-demand dynamic distillation,” in *Proceedings of the 7th International Conference of Architectural Support for Programming Languages and Operating Systems*, pp. 160–173, 1996.
- [29] B. Badrinath, A. Fox, L. Kleinrock, G. Popek, P. Reiher, and M. Satyanarayanan, “Conceptual framework for network and client adaptation,” *Mobile Networks and Applications*, vol. 5, no. 4, pp. 221–231, 2000.
- [30] D. M. Batista, N. L. S. da Fonseca, F. K. Miyazawa, and F. Granelli, “Self-adjustment of resource allocation for grid applications,” *Computer Networks*, vol. 52, no. 9, pp. 1762–1781, 2008.
- [31] J. Cao, K. M. McNeill, D. Zhang, and J. F. Nunamaker, “An overview of network-aware applications for mobile multimedia delivery,” in *Proceedings of the Hawaii International Conference on System Sciences*, vol. 9, pp. 4663–4672, January 2004.
- [32] M. Davis, “A wireless traffic probe* for radio resource management and QoS provisioning in IEEE 802.11 WLANs,” in *Proceedings of the 7th ACM Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, pp. 234–243, October 2004.
- [33] M. Bredel and M. Fidler, “A measurement study of bandwidth estimation in IEEE 802.11g wireless LANs using the DCF,” in *Proceedings of the Networking Ad Hoc and Sensor Networks, Wireless Networks, Next Generation Internet*, vol. 4982 of *Lecture Notes in Computer Science*, pp. 314–325, 2008.
- [34] R. Prasad, C. Dovrolis, M. Murray, and K. Claffy, “Bandwidth estimation: metrics, measurement techniques, and tools,” *IEEE Network*, vol. 17, no. 6, pp. 27–35, 2003.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

