ICWMMN2006 Proceedings

WLAN Emulation on StarBED

Răzvan Beuran†‡, Ken-ichi Chinen‡†, Khin Thida Latt‡, Toshiyuki Miyachi‡†, Junya Nakata‡†, Lan Tien Nguyen‡, Yoichi Shinoda‡†, Yasuo Tan‡†, Satoshi Uda‡†, Saber Zrelli‡

† National Institute of Information and Communications Technology,
Hokuriku Research Center, 2-12 Asahidai, Nomi, Ishikawa, 923-1211, Japan
‡ Japan Advanced Institute of Science and Technology, 1-1 Asahidai, Nomi, Ishikawa, 923-1292, Japan
E-mail: razvan@nict.go.jp

Abstract

In this paper we present the Wireless LAN (WLAN) emulator that we design to run on StarBED, the large scale network experiment environment of the Hokuriku Research Centre in Ishikawa, Japan. As WLANs start to spread on an ever wider scale, the need to investigate application deployment on WLANs has become stringent. We develop an emulation environment since it allows using in experiments the real applications that will be deployed in practice. In addition, it offers full control of the network conditions in which the applications will be tested. Our approach is user-oriented in the sense that the input of the system is a user-level scenario representation. This is subsequently converted into a quality degradation description that represents the effect the network will have on application traffic. Finally, the quality degradation description is transformed into an emulator configuration that is effectively used in the emulation environment to enforce network quality degradation on the application traffic.

Keywords: WLAN emulation, application performance assessment, StarBED, convergence.

1 Introduction

Wireless LANs (WLANs) are being more and more widely deployed at present as the number of mobile users is increasing steadily. Given the different environments in which WLANs are used, the types of applications that need to run on these networks vary. Satisfactory solutions for simultaneously running applications with different traffic characteristics have not been found yet even for the traditional fixed networks. Given the inherent properties of wireless networks, the situation becomes even more challenging in this case. According to the survey we did in [1], difficulties are met at the moment for several reasons:

i. WLAN QoS parameters (bandwidth, packet loss, delay & jitter) have a high variability in real-world environments, and this has a significant effect on application performance;

ii. Existing WLAN QoS mechanisms are only of limited use for managing contention when applications with different QoS requirements, such as Voice over IP (VoIP) calls and TCP-based data traffic, share the same communication channel;

iii. Multimedia applications require timely servicing of their traffic; this is a challenging task in WLANs, even when using QoS enforcement, since most currently-implemented QoS mechanisms focus on bandwidth provisioning;

iv. Roaming between access points, a typical WLAN event, introduces communication gaps that may even be of the order of seconds, an unacceptable situation for real-time applications.

Nevertheless, users of WLANs, in either regular or special environments, such as emergency or disaster conditions, undoubtedly require that applications run at a satisfactory performance level. The emulation system we design is a tool that makes it possible to thoroughly analyse application performance in a wide range of controllable network conditions. By correlating an objective assessment of the User-Perceived Quality (UPQ) for the applications under study with the corresponding network conditions one can determine the reasons of application performance degradation, and investigate the mechanisms needed to ensure satisfactory performance.

The WLAN emulator that we started developing as part of our research on application performance on wireless LANs makes use of StarBED [2]. StarBED is the large scale network experiment environment of the National Institute of Information and Communications Technology (NICT) Hokuriku Research Centre in Ishikawa, Japan. This experiment environment is a cluster-based testbed currently employing about 700 PCs. The use of the custom-designed configuration language SpringOS makes it possible to define complex experiments on StarBED in a straightforward manner.

A survey of existing WLAN-related real-world and simulation testbeds is available in [3]. General characteristics of WLAN emulators are summarized in [4]. Our design addresses these requirements through a scenario-driven two-stage approach inspired by [5] and [6]. The novelty of our work consists in the quality degradation view we take, the emphasis we lay on emulation realism, as well as the use of a large-scale testbed.

The paper is structured as follows. First we present more details about the StarBED network experiment environment. Then we discuss the architecture we propose for WLAN emulation. Following that we show a typical test setup and some illustrative results for our study of application performance on WLAN, using VoIP as an example. The paper ends with a section of conclusions and future work.

2 StarBED

The core of StarBED consists of a cluster of about 700 nodes that are actually standard PCs, with redundant full connectivity by means of an experiment switch cluster (see Fig.1). The connectivity and specific switch configurations are used to produce a target network topology using Virtual LAN (VLAN) technology. On the cluster nodes we can use popular end-user operating systems such as Windows or Linux, software router systems based on the Unix operating system, or even wired-network emulators such as "dummynet" [7]. There are additional empty locations in the core network where users of the experiment environment can plug in their own devices to form an appropriate target network. These devices may be products under test, commercial routers, measurement equipment, etc.

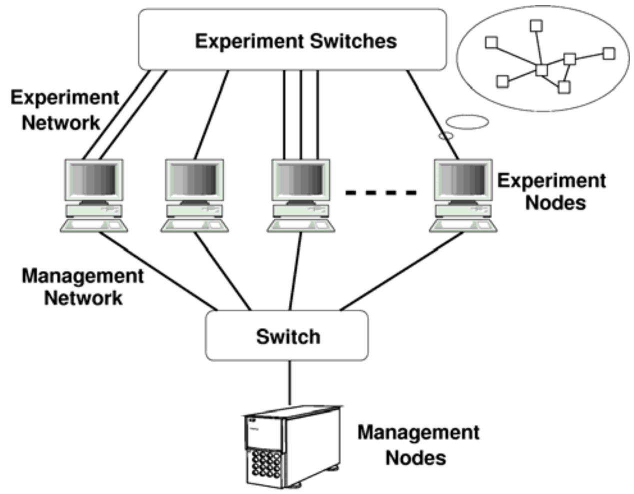

Figure 1: StarBED topology

In addition to the core experiment network, there is a dedicated management network that controls and monitors node and switch activity. Nodes can be loaded with appropriate software, controlled, and monitored using the management network, without affecting a running emulation experiment. The experiment switch cluster has a backdoor connection that can be used to monitor network activity, or to create connections so as to link the core network with the external world. A large emulation environment using standard PCs nodes, such as StarBED, behaves very similar to a real network, since it uses real nodes similar to those from real-world environments.

Given the large scale of the StarBED network experiment environment, we developed a software management system named SpringOS that assists users in making experiments. When using SpringOS, users must describe the experiment by a configuration file, and SpringOS runs the experiment following that configuration file. The functions of SpringOS are: uploading the appropriate operating system and software to the nodes, configuring switches to build the target topology, and driving the execution of the experiment scenario.

There are many experiments that have been performed on StarBED. This environment is also an appropriate platform for our WLAN-related experiments for several reasons: (i) use of real nodes in a large scale setup, that makes it possible to realistically emulate large WLAN environments; (ii) flexibility of the test environment, that allows to easily use different hardware configurations; (iii) a powerful management system that enables the control, quick reconfiguration, and even concurrent use of the facility for independent experiments.

3 Emulator architecture

The scenario-driven architecture we propose here has two stages. In the first stage, from a real-world scenario representation, we create a network quality degradation (ΔQ) description which corresponds to the real-world events (see Fig.2). The ΔQ description represents the varying effects of the network on application traffic. Subsequently this description is converted into an emulator configuration which is used during the effective emulation process to replicate the user scenario. This permits to study the effects of the scenario on the application under test.

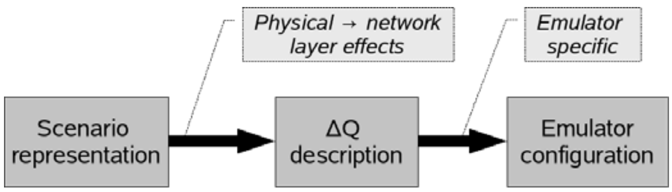

Figure 2: Two-stage scenario-driven emulation

As an example, assume that the scenario representation describes how, from an initial condition, a mobile node moves with respect to an access point. As a consequence of the motion, the received radio signal strength will change. This is the WLAN physical layer effect of the movement, and it is quantified by means of the received signal power, *Pr*. We model the dependency of *Pr* on the distance between transmitter and receiver using the log-distance path loss model [8].

The variation of *Pr* causes quality degradation at the data-link layer, as follows. The weaker signal first induces higher packet loss. Then there is a delay as the WLAN adapters change the channel encoding and rate, according to a mechanism such as the Auto-Rate Fallback (ARF) [9]. At the end the available bandwidth diminishes. These effects are modelled by taking into account typical receive sensitivities of WLAN adapters, as well as the behaviour of ARF.

The first stage of our approach will therefore transform a sequence of real-world events at moments T_0, T_1, ..., T_n, into a sequence of ΔQ descriptors ΔQ_0, ΔQ_1, ..., ΔQ_n, where ΔQ_i

represents the state of the network (available bandwidth, packet loss rate, delay and jitter) at moment T_i, with i=0, ..., n.

The second stage of our approach is emulator specific. In this stage the generic ΔQ description calculated previously is converted into an emulator configuration. Initially the conversion target is the "dummynet" network emulator running on StarBED. However by decoupling the two conceptually-independent stages we make it possible that later on other emulators will be used as well, emulators that are more accurate than "dummynet" and that have a richer set of features.

An essential component of our approach is the emphasis we lay on emulation realism. Existing WLAN emulators often use models that are too simple, hence do not adequately reflect real network conditions as experienced by users, and only reproduce a simplified theoretical behaviour of WLANs. However the edge effects that do occur in reality are generally overlooked (e.g., between rate changes, as the WLAN cards automatically adapt operation rate to signal reception conditions using the ARF mechanism). Nevertheless these effects have evident consequences at application level (see the next section for an example). Moreover, in order to achieve an even higher degree of realism we propose to calibrate our system by combining observations and traffic traces of real WLANs with our analytical model of the WLAN environment, so that the ΔQ description that we produce from the scenario representation accurately describes the observed network behaviour.

4 Application study

Our main research interest lies in the area of application performance on WLANs. The two directions of our research are application performance analysis and assurance. Due to its specific real-time requirements, VoIP has been selected as the network application we started performing experiments with. We already have extensive experience concerning VoIP performance on wired networks [10]. Using the emulator presented here we extend our research to the WLAN environment. A typical experiment setup is shown in Fig.3.

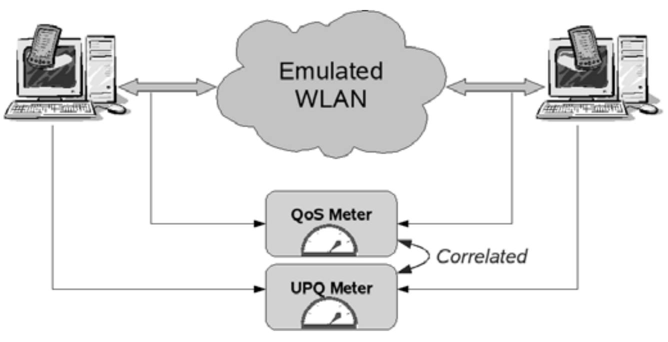

Figure 3: WLAN emulation for application performance assessment

The end nodes in Fig.3 are usual PCs that represent the mobile nodes in our emulated environment. We can compute the QoS parameters of the emulated WLAN by means of the "QoS Meter" block that uses sniffed traffic traces. Simultaneously we measure application-level performance with the "UPQ Meter", by employing application-specific performance metrics. Correlating these two measures makes it possible to determine objectively the relationship that exists between application performance and network quality degradation. For VoIP we use the ITU-T E-model [11] to predict voice communication quality based on the ΔQ descriptors. The output of the E-model is the R-value, which is on a scale from 0 to 100. Its interpretation is the following: an R-value higher than 80 indicates good quality, values between 80 and 50 signify acceptable quality, and R-values below 50 indicate unacceptable quality.

To illustrate our approach, consider a scenario with a node that from an initial distance of 10 m with respect to an access point moves on a perpendicular direction with a speed of 0.5 m/s for a duration of 30 s. This is a scenario fragment representative of a user moving in a building while making a VoIP over WLAN phone call. For simplicity reasons we considered in this scenario that the VoIP call destination is a fixed node located in the same LAN.

We used for the log-distance path loss model the following parameters: α = 5.5 (difficult reception conditions) and $Pr0$ = -20 dB. We obtained the variation of the power of the received signal, Pr, which is shown in Fig.4 plotted versus time. We modelled the WLAN effects of this variation in terms of packet loss, delay and bandwidth by taking into account realistically the consequences the Pr variation has on the quality degradation ΔQ. This includes packet loss increase when Pr falls under the receive sensitivity thresholds for the different operating rates, and ARF side-effects such as the delay associated with rate variation.

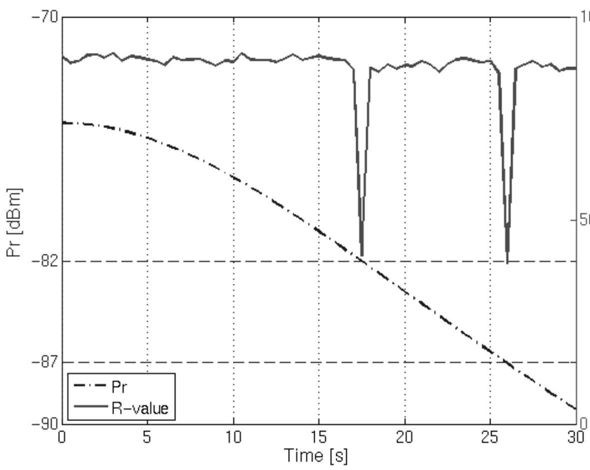

Figure 4: Received signal power, Pr, and the E-model output, R-value, versus time

Using the ΔQ computed above as the input of the ITU-T E-model we calculated the R-value and predicted the UPQ for the VoIP communication (see Fig.4). Note the effect that rate changes due to the ARF mechanism have on VoIP UPQ. During the rate transition periods VoIP quality drops below the acceptability threshold of 50 on the R-value scale. These effects would not be revealed by our emulation environment if we would take only bandwidth changes into account, as most WLAN emulators do. This is because the available bandwidth in our scenario is 2 Mb/s in the worst case, largely sufficient for an 80 kb/s VoIP stream. The effects that we evidence are caused by the increased packet loss and delay variation that occur in reality when WLAN adapters oscillate between operating rates.

The results shown in Fig.4 represent the predicted behaviour of a VoIP system in a scenario as the one we emulated. They constitute the basis for the experiments using the emulator environment that we described, and will be further validated through real-world tests.

5 Conclusions

The approach we propose for WLAN emulation allows transforming a user-meaningful real-world representation of a WLAN environment (termed "scenario representation") into a network quality degradation description (termed "ΔQ description"). The ΔQ description is sufficient to subsequently configure an emulator and effectively reproduce an environment that corresponds accurately at network level to the emulated scenario.

We illustrated the practical use of our approach by emulating a simple real-world scenario, for which we determined the induced network quality degradation due to node mobility and rate adaptation. Then we quantified the influence of this quality degradation on VoIP UPQ in an objective manner using the ITU-T recommendation G.107 concerning expected user satisfaction for VoIP communication. We were thus able to quantify the effects the real-world scenario would have on application performance.

As we will be able to run more complex scenarios, the use of the large scale network experiment environment at StarBED will make it possible to run experiments involving up to 700 real nodes, or even up to 10 times more nodes by using virtual machine technology.

In a second phase of our research we will investigate application performance assurance. We intend to use the same setup to study techniques of application performance assurance in WLAN environments. The framework and the specific techniques that we plan to develop will allow the creation of ad-hoc WLANs under critical conditions, and the assurance of quality guarantees even under such circumstances for high-priority users (e.g., public safety teams, hospitals, etc.).

References

[1] Beuran R. "VoIP over Wireless LAN Survey", *research report, IS-RR-2006-005*, Japan Advanced Institute of Science and Technology (JAIST), Ishikawa, Japan, (April 2006).

[2] Miyachi T., Chinen K. and Shinoda Y. "Automatic Configuration and Execution of Internet Experiments on an Actual Node-based Testbed", *Tridentcom2005*, Trento, Italy, pp. 274-282, (February 2005).

[3] Kropff M., Krop T., Hollick M., Mogre P.S. and Steinmetz R. "A Survey of Real-World and Emulation Testbeds for Mobile Ad hoc Networks", *Proc. of TridentCom2006*, Barcelona, Spain, (March 2006).

[4] Kojo M., Gurtov A., Manner J., Sarolahti P., Alanko T. and Raatikainen K. "Seawind: a Wireless Network Emulator", *Proc. of 11th Conference on Measuring, Modelling and Evaluation of Computer and Communication Systems (MMB)*, Aachen, Germany, (September 2001).

[5] Bateman M., Allison C. and Ruddle A. "A Scenario Driven Emulator for Wireless, Fixed and Ad Hoc networks", *Proc. of PGNet2003*, Liverpool, U.K., pp. 273-278, (June 2003).

[6] Perennou T., Conchon E., Dairaine L. and Diazet M. "Two-Stage Wireless Network Emulation", *Proc. of WCC2004*, Toulouse, France, (August 2004).

[7] Rizzo L. "Dummynet FreeBSD network emulator", *http://info.iet.unipi.it/~luigi/ip_dummynet.*

[8] Rappaport T. S. "Wireless Communications: Principles and Practice", *Prentice Hall PTR*, 2nd edition, (2002).

[9] Kamerman A. and Monteban L. "WaveLAN-II: A High-performance wireless LAN for the unlicensed band", *Bell Lab Technical Journal*, pp. 118-133, (summer 1997).

[10] Beuran R. and Ivanovici M. "User-Perceived Quality Assessment for VoIP Applications", *technical report, CERN-OPEN-2004-007*, Geneva, Switzerland, (January 2004).

[11] ITU-T Recommendation G.107. "The E-model, a computational model for use in transmission planning", *ITU-T*, (March 2005).