BANDWIDTH ESTIMATION IN WIRELESS LANS FOR MULTIMEDIA STREAMING SERVICES

[1]*Heung Ki Lee,*[1]*Varrian Hall,*[2]*Ki Hwan Yum,* [3]*Kyoung Ill Kim and* [1]*Eun Jung Kim*
[1] *Texas A&M University,* [2]*University of Texas at San Antonio,*
[3]*ETRI (Electronics and Telecommunication Research Institute)*

{hklee, cystar, ejkim}@cs.tamu.edu, yum@cs.utsa.edu, kki@etri.re.kr

ABSTRACT

The popularity of multimedia streaming services via wireless networks presents major challenges in the management of network bandwidth. One challenge is to quickly and precisely estimate the available bandwidth for the decision of streaming rates of layered and scalable multimedia services. Previous works based on wired networks are too burdensome to be applied to multimedia applications in wireless networks. In this paper, a new method, *IdleGap*, is suggested to estimate the available bandwidth of a wireless LAN based on the information from a low layer in the protocol stack.

We use a network simulation tool, NS-2, to evaluate our new method with various range of cross traffic and observation times. Our simulation results show that *IdleGap* accurately estimates the available bandwidth for all ranges of cross traffic (100Kbps ~ 1Mbps) with a very short observation time of 10 seconds.

1. INTRODUCTION

Since introduced commercially in 1995, multimedia streaming services have become one of the most promising Internet services currently available. In addition, Wireless Local Area Networks (WLANs) make multimedia streams commonplace, and terminals are diversifying into hand-held devices such as PDAs, laptops and audio/video players. These heterogeneous devices have different access patterns and mobility [1]. Most multimedia streams are hungry for stable network bandwidth, but a shared-medium WLAN may not support it. To meet their bandwidth requirements, rate scalability can be achieved by layered video representation [2,3]. However, there are still problems in

Figure 1. Stream Service Based on Set Top Box and 802.11

estimating the point in time to change the bit rate of the transmitted bit stream. Estimating the available network bandwidth in a WLAN is very challenging and crucial for multimedia streaming services.

Although there can be various wireless environments where multimedia services are provided, we mainly focus on the LAN/WAN shown in Figure 1. In this figure, an Internet-based Set Top Box (STB) is the interface between a wired network and a wireless network. Even though wired networks can provide high and stable bandwidths, fragile wireless networks may not support it. Therefore, for layered streaming services, it is very critical for the STB to know the available wireless network bandwidth.

In a wireless network, the IEEE 802.11 protocol in Distributed Co-ordination Function (DCF) mode, based on CSMA/CA algorithm, is becoming very popular. Previous works [4,7,8] based on the bandwidth estimation of wired environments are not applicable to wireless networks that use the DCF protocol. Multimedia streaming is a soft real-time service where each frame is delay-sensitive. Swiftness and availability is critical for real time system. During bandwidth deviations, the rate of the transmitted multimedia streams should change expeditiously. The accuracy of previous works, *Spruce*[4] and *ProbeGap*[8], is dependent on probing time and the volume of the packets for probing. *ProbeGap* produces good estimates at low cross traffic rates (2 Mbps cross traffic regardless of the cross traffic packet size); however, it significantly overestimates available bandwidth when the cross traffic is high (4 Mbps cross traffic generated with 300-byte packets) [8]. Influence by cross traffic on probe packet sequences causes probe packets in sequences to be split up or even lost**.**

Our contribution in this paper is two-fold. First, we suggest *IdleGap*, which is a bandwidth estimation tool for a real-time system in a wireless network. Second, our system is independent of cross traffic. We estimate the available bandwidth via the ratio of free time in the wireless links. To get the ratio of idle time in a wireless network, information from network management at the low layer is used. It provides us with an efficient and fast method for estimating the available bandwidth.

The rest of the paper is organized as follows. Section 2 shows the related work in estimating bandwidth and discusses the Cross layer. In Section 3, our new method,
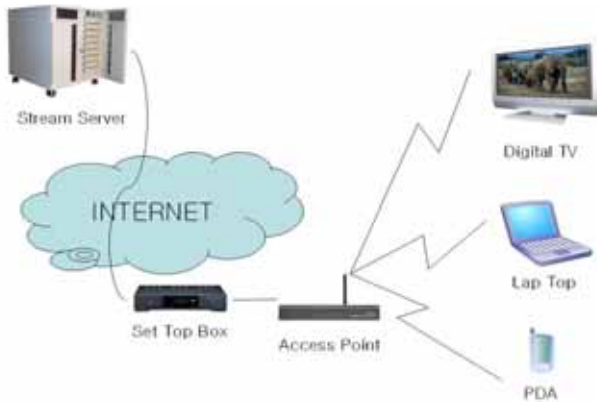
IdleGap, is proposed and known challenges in bandwidth estimation are addressed. After presenting the results of our method and other tools in Section 4, we conclude this paper in Section 5.

2. RELATED WORK

2.1 Estimation of Bandwidth in Broadband Networks

Since the introduction of Cprobe [12], a method for estimating bandwidth using Internet Control Message Protocol (ICMP) packet trains, many tools have been suggested. *Spruce* [4] and *IGI* [5] use the interval of consecutive probe packets, since the interval or gap between probe packets is increased in heavy cross traffic. *Topp* [6] and *Pathload* [7] are based on the rate of incoming packets. The comparison of the incoming rate from the sender side to the outgoing rate at the receiver side reveals the incoming rate to be less than or equal to the available bandwidth of the probing link. In *Probegap*[8], the link's idle time is the milestone for bandwidth estimation of a wireless network.

2.2 Cross Layer Feedback

For efficient mobile device communication and interaction, cross layer feedback is performed by a mobile device accessing its own protocol stack layers that contain information from transmitted packets. Cross layer feedback allows interaction between a layer and any other layer in the protocol stack. Packet information retrieval across the protocol stack layers (cross layer) provides very useful information about mobile devices in the wireless network. Several studies [9,10,11,12] revealed interaction among other layers for improving the system. In [9], the central bandwidth manager controls client traffic by updating the Defer time within the MAC layer. [10] suggested a 802.11 management method that processes the captured frame to get the available bandwidth. For a QoS-sensitve application, a different priority at the MAC layer may be assigned based on the applications [11].

3. IDLEGAP USING NETWORK ALLOCATION VECTOR

3.1 Background

Bandwidth estimation is a prerequisite problem for real-time applications in wireless networks. There are two factors making this problem unique. First of all, unlike wired networks, traditional FIFO is not used to schedule bandwidth among connections in wireless networks. To avoid collisions in wireless networks, nodes are arranged in a distributed manner. This arrangement causes bandwidth estimation methods in wired networks using intervals [4,5] or rates [6,7] inapplicable for bandwidth estimation in wireless networks. Secondly, probing time for the available bandwidth should be short for time-sensitive multimedia streaming services. [8,10] suggested that idle time of a link in a wireless network can be a major milestone for estimating the available bandwidth as follows:

Let *C* be the capacity of the wireless network[1]. *Idle_rate* indicates the rate at which the link is idle. Then the available bandwidth (*AB*) can be obtained by the following product:

$$AB = C \times Idle\_rate \qquad (1)$$

However, previous methods [8,10] using this formula cause too much overhead to be used in a real-time system for the estimation of the available bandwidth. In [8], too much time elapsed probing the link and analyzing probing data, and results showed multiple incorrect estimated values in heavy traffic. [10] utilizes too much time in order to capture whole packets in the network and get node information from captured packets. For real-time applications such as multimedia streams, it is difficult to use these methods; therefore, in Section 3.2.2 we introduce an efficient method to calculate the *Idle_rate*.

3.2 IdleGap

3.2.1 Network Allocation Vector

When two nodes in a wireless network share the same Access Point (AP) but cannot hear each other, one node will not be able to know whether the other node is already using the shared resource, i.e., the wireless channel. For addressing this hidden node problem, each node uses the Network Allocation Vector (NAV) that shows how long other nodes allocate the link in the IEEE 802.11 DCF MAC protocol. Even though a node is located at a place where it cannot reach other active nodes, the node can know whether another node is already using the wireless network by checking its NAV. In Figure 2, when the sender sends RTS (Request To Send) to the receiver (AP), *Other-1* node that is reachable from sender updates its NAV. However, *Other-2* node does not update NAV, because it is not reachable from sender. When the receiver sends CTS (Clear To Send), *Other-2* node updates its NAV. The idle time in the wireless network can then be estimated from the NAV information.
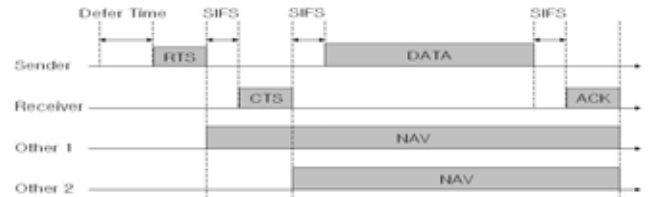

Figure 2. IEEE 802.11 DCF MAC Protocol

3.2.2 Estimation of Wireless Link Idle Rate

All nodes in a WLAN share the same resource; i.e., a wireless channel. If a node in a WLAN is utilizing the resource, the additional node(s) should await the release of the wireless channel. During a transmission in a WLAN, a

---

1. It can be changed by the negotiated data rate between a wireless node and the access point.

node can be one of the following: sender, receiver or on-looker. If a node transmits data to another node, it is a sender. A node is a receiver if receiving data. Finally, when a node does not join the transmission, it is an onlooker.

The busy time of the link can be estimated by adding up all the transactions of nodes in the network as depicted in Equation (2). Here T_l is the busy time of link l and $TT(i, j)$ indicates the transaction time between nodes i and j.

$$T_l = \frac{1}{2} \times \sum_{i=1}^{n} \sum_{j=1}^{n} (TT(i, j)). \qquad (2)$$

Unfortunately, we can not know all the transaction times from the nodes in the network. In addition, obtaining the transaction information can increase network traffic, hence affecting current traffic on the network. Therefore, we propose to obtain all the necessary information from one node in the network as follows.

The transaction time of node i can be obtained via the sum of the sending and receiving time to/from node i ($TT(i, j) = ST_i + RT_i$,where ST_i is the sending time from node i to j and RT_i is the receiving time from node j to i). For the transaction time between other nodes, we can get the on-looking time from the NAV in node i that is updated in other node transactions ($TT(i, j) = OT_i$, where OT_i is the on-looking time at node i). Therefore, we can estimate the busy time T_l in any node i in the network as shown in Equation (3):

$$T_l = ST_i + RT_i + OT_i. \qquad (3)$$

We can then obtain *Idle_rate* using the busy time:

   Idle_rate = 1 – busy time/total elapsed time.

3.2.3 System Model

We propose to add an *Idle-Module* in the MAC layer of a node in the network. This module obtains the busy time (T_l) from (a) and (b) in Figure 3. The transaction time of a node can be obtained through accessing outgoing and incoming packets ($ST_i + RT_i$) between the Network layer and the Link and MAC Layer (shown in (b)). Idle-Module also gets the on-looking time (OT_i) from the NAV (shown in (a)). The updating process of the NAV triggers the *Idle-Module* to update its value. An application can access the *Idle-Module* to get the idle rate (1 – busy time/total elapsed time). Then applying the idle rate and link capacity C to Equation (1) above, the estimated bandwidth of the link can be calculated with minimal effort. We call this method *IdleGap*.

4. EXPERIMENTAL RESULTS

To verify the performance of our *IdleGap* method, network simulations were conducted using NS-2. As shown
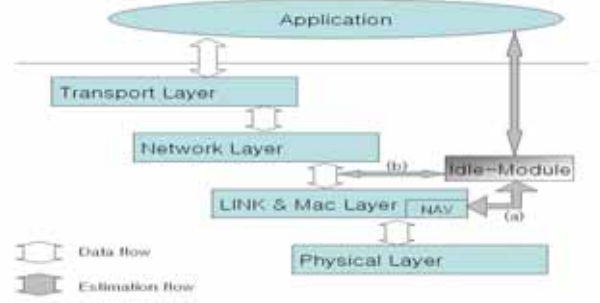

Figure 3. Architecture of Idle-Module

in Figure 4, there are seven nodes including three wired nodes, three wireless nodes and an AP. In the wired network, the capacity of the link was set to 10Mbps, while the capacity in wireless network was set to 1Mbps.

In Figure 4, communication in the simulation via the AP involves three connections: Wired Node 1 to Wireless Node 2, Wired Node 2 to Wireless Node 1, and Wired Node 3 to Wireless Node 3. Wired Nodes 1 and 2 generate the cross traffic, while the algorithm generates timestamps from packets received by Wireless Node 3 via packets sent from Wired Node 3 to estimate the available bandwidth. We compare *IdleGap* with *ProbeGap* [8] and *Spruce* [4], which provides more accurate estimation than other previous works.
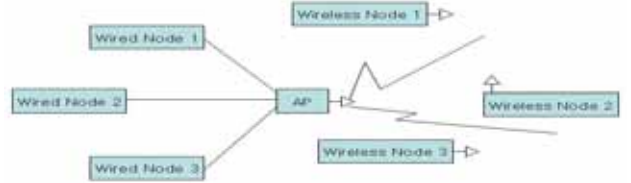

Figure 4. Simulation Environment

4.1. Experiment with Increasing Cross Traffic

Figure 5 shows the estimated available bandwidth value for each algorithm. The capacity of the wireless network in our simulation is 1 Mbps. Probing time for each algorithm is 1000 seconds and 200 probing packets are allowed. In light cross traffic, *ProbeGap* produces bandwidth estimates reflective of measured available bandwidth values. However, it shows multiple transition points over 200Kbps cross traffic. In the original *Spruce* paper, the intra-pair gap is set to the transmission time of the narrow link [4]. This causes the underestimation of the available bandwidth for the link. Therefore, the intra-pair-gap was calibrated to reflect the available 1.0 Mbps with no cross traffic. Even after the calibration, *Spruce* overestimates the bandwidth severely with more than 0.5Mbps cross traffic. The reason is due to high drop rates with heavy cross traffic. Thus, the estimated bandwidth value becomes polluted. This could cause the overestimation of the available bandwidth.

The *IdleGap*, which uses NAV to estimate bandwidth, shows the closest match to the real bandwidth. Note that after 0.6Mbps cross traffic, saturation occurs due to the overhead of the wireless network such as defer time and RTS/CTS.

4.2. Experiment with Different Observation Times

In this experiment, we vary the observation time to estimate the available bandwidth. Since we focus on the effect of observation period, the cross traffic is set to 10Kbps, where all three schemes are able to estimate the bandwidth accurately as shown in Figure 5. *ProbeGap* and *Spruce* send the probes at intervals of 5 seconds [8]. Figure 6 shows the estimated values of the available bandwidth for *ProbeGap, Spruce* and *IdleGap* between observation periods of 10 and 500 seconds. Until 250 seconds, *ProbeGap* and *Spruce* record values not reflective of measured available bandwidth. After 250 seconds, *ProbeGap* and *Spruce* values are near the measured bandwidth values. However, *IdleGap* generates values reflective of measured bandwidth for all periods. Therefore, we can conclude that *IdleGap* provides accurate estimations with short observation times.
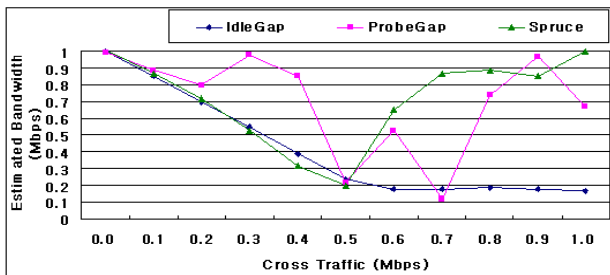

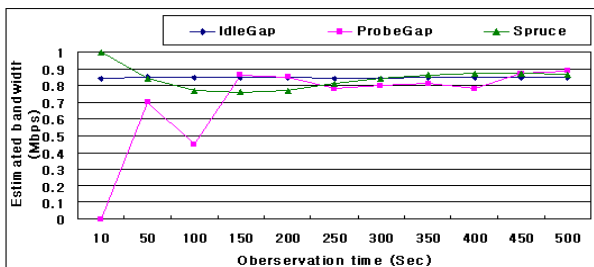
Figure 5. Estimated Bandwidth With Cross Traffic

Figure 6. Estimated Bandwidth With Different Observation Times

5. CONCLUDING REMARKS

The most challenging aspect of multimedia streaming services is the adaptive bit rate of each multimedia stream according to the network status; therefore, in this paper we focused on a method to estimate the available bandwidth of a wireless link. The method must have the following characteristics: (a) it should be applicable to real-time applications such as multimedia streaming services; (b) be simple and effective in estimating the available bandwidth and (c) incur low overhead.

We have presented a new bandwidth estimation method, *IdleGap*, which can efficiently calculate the available bandwidth using the information collected from one node in a wireless network. *IdleGap* is simple and does not incur extra network overhead. The simulation result shows that *IdleGap* outperforms the other probing and bandwidth estimation methods, *ProbeGap* and *Spruce*.

REFERENCES

[1] T. Henderson, D. Kotz, and I. Abyzov, "The Challenging Usage of a Mature Campus-wide Wireless Network," in Mobicom, 2004.

[2] J. Shin, J. Kim, and C.-C. J. Kuo, "Quality-of-Service Mapping Mechanism for Packet Video in Differentiated Service Network," IEEE Trans. Multimedia, vol. 3, no. 2, pp. 219-231, June 2001.

[3] D. Quaglia and J. C. de Martin, "Delivery of MPEG Video Streams with Constant Perceptual Quality of Service," in Proc. International Conference Multimedia and Exhibition, Lausanne, Switzerland, IEEE, vol.2, pp. 85-88, Aug. 2002.

[4] J. Strauss, D. Katabi, and F. Kaashoek, "A Measurement Study of Available Bandwidth Estimation Tools," in IMC, 2003.

[5] N. Hu and P. Steenkiste, "Evaluation and Characterization of Available Bandwidth Techniques," in IEEE JSAC Special Issue in Internet and WWW Measurement, Mapping, and Modeling, 2003.

[6] B. Melander, M. Bjorkman, and P. Gunningberg, "A New End-to-End Probing and Analysis Method for Estimating Bandwidth Bottlenecks," In Global Internet Symposium, 2000.

[7] M. Jain and C. Dovrolis. "Pathload: A Measurement Tool for End-to-End Available Bandwidth." in Passive and Active Measurements, Fort Collins, Co, March, 2002.

[8] K. Lakshiminarayanan, V. N. Padmanabhan, and J. Padhye, "Bandwidth Estimation in Broadband Access Networks," in IMC, 2004.

[9] Samarth H. Shah, Kai Chen, and Klara. Nahrstedt, "Dynamic Bandwidth Management in Single-Hop Ad Hoc Wireless Networks", in Mobile Networks and Applications, Oct, 2005.

[10] M. Davis, "A Wireless Traffic Probe for Radio Resource Management and QoS Provisioning in IEEE 802.11 WLANs," in Proc. of the 7th ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems, 2004.

[11] G. Xylomenos and G. C. Polyzos, "Quality of Service Support over Multi-Service Wireless Internet Links," in Wireless Internet Links, Computer Network 37, 2001.

[12] R. L. Carter and M. E. Crovella, "Dynamic Server Selection Using Bandwidth Probing in Wide-Area Networks," Technical Report TR-96-007, Boston University Computer Science Department, 1996.