Connection Sharing in an Ad Hoc Wireless Network among Collaborating Hosts

Maria Papadopouli
Department of Computer Science
Columbia University
maria@cs.columbia.edu

Henning Schulzrinne
Department of Computer Science
Columbia University
hgs@cs.columbia.edu

Abstract

We present an architecture that enables network connection sharing in an environment of mobile, wireless, collaborating hosts. Dual-homed hosts with wireless connection to the Internet share their connection with other hosts of the collaborating group by acting as temporal gateways. The system aims to increase data availability and quality of service while achieving load balancing across the gateways. We focus on collaborative applications that support scalable, multimedia, streaming data, such as layered video. We discuss the main components of the system, namely, admission control, gateway selection, measurement of the gateway traffic and the announcement policy. Finally, we present the simulation results to quantify the system performance.

1 Introduction

Light-weight laptops and PDAs, the deployment of wireless networks and services, and the popularity of the Internet make mobile computing attractive. These technologies will enable users to connect to the Internet anytime and anywhere. This anytime-anywhere access will trigger the design of new applications and enrich already existing ones. We would like to design a system that assists users who wish to collaborate by creating a wireless network on the spot, dynamically. By the term "on the spot" we mean instantaneously, with minimal overhead. The collaboration takes place through existing applications, such as teleconferencing, news on demand, electronic white boards, etc. We are particularly interested in these collaborative applications that support scalable, multimedia, streaming data, such as layered video.

Current wireless technologies vary in terms of bandwidth, latencies, frequencies and coverage. We can divide these into two main categories: those that provide a low bandwidth over a wide geographic area and those that provide a high-bandwidth service over a narrow geographic area. We speculate that no single wireless network technology can simultaneously provide low latency, high bandwidth, wide-area data service to a large number of mobile users. In order a mobile user to maintain connectivity to the Internet, the host may need to access different networks depending on the speed at which the user moves and the availability of base stations. Also, depending on a given task, the host may need to access different networks. For example, if she wants to retrieve

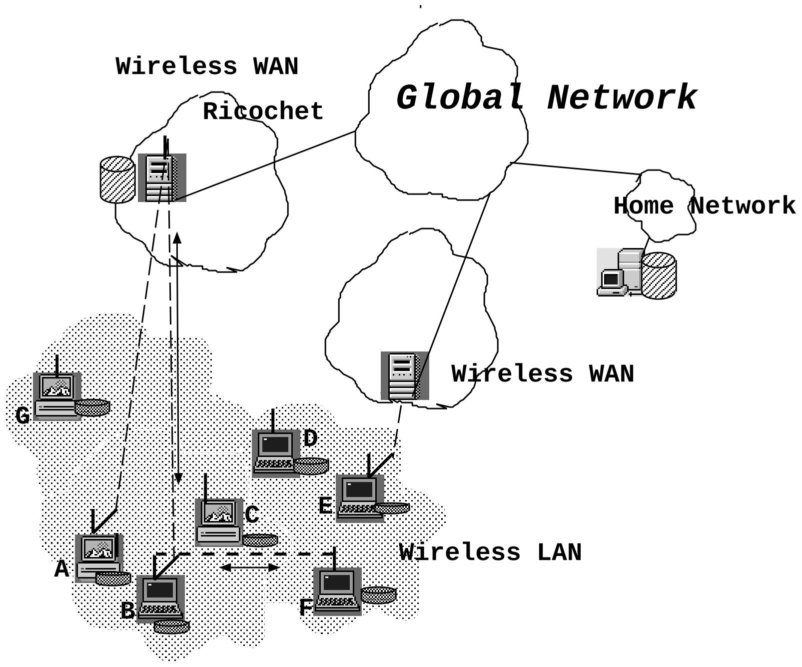

Figure 1: Description of the environment: Hosts share a wireless LAN. Some of them have wireless WAN connection to access the Internet.

information from devices in close proximity, the host will connect to the local area network they form. Or if she needs to teleconference with a colleague who is far away, the host may have to access the wide area network. We speculate that it would not be unusual for mobile users that need high data availability to keep two network interfaces in order to increase the data availability and connectivity to the network. As an example, consider a user with a wireless modem (e.g., Ricochet [1]) and a WaveLAN card [2] or an infrared interface. Or, instead of a Ricochet modem, the user may have a Bluetooth device [3] that enables the laptop to transmit and receive data via her mobile phone. Either via the Ricochet modem or via the mobile phone, she is able to access the global network. The user may communicate with others in close proximity via the WaveLAN or the infrared. However, it may not be possible to access the Internet via the WaveLAN interface. In Section 2, we mention briefly related work that considers a deployment of a combination of wireless networks of different technologies. Users with multiple network interfaces are moving in this networking environment.

Current wide area network wireless deployment is characterized by intermittent connectivity, low bit rates, and high end-to-end delays. These constraints provide a strong incentive to make better utilization of the user's local resources in order achieve better quality of service (QoS) and higher data availability. The characteristics of collaborative applications led us to the natural extension of data sharing to network connection sharing. The central idea is that collaborating hosts share their network connections in order to improve their service, increase the data availability and have potentially other benefits, as we describe in the following paragraphs.

Figure 1 illustrates the setup we consider in this paper: there is a group of hosts (A, B, C, D, E, F and G) in close proximity that are capable of communicating via a wireless LAN, such as a WaveLAN card. Some of them (A, B and E) have an additional network interface that provides them with access to the Internet via a wireless WAN connection. All the hosts can communicate with each other via the wireless LAN. It would be typical in the near future to support a wireless WAN connection of 100 kb/s [4], [1], [5] [1].

We envision this system to be applicable especially in cases where users meet in a conference or in a meeting (e.g., IETF meeting) or in a train and want to gain Internet access. We assume that the users are selfish. They decide to cooperate and share/lend their resources in order to facilitate a common need and potentially have other gains as we describe in the following paragraphs.

The motivations for connection sharing are[2]

- Utilization of the temporarily idle connections.
- Exploit the statistical multiplexing for bursty traffic.
- Reduction in the transmission of replicated data that belong to "shared" (collaborative) applications.

Let us discuss further these motivations with some examples. When a user is connected to the network, there are periods when the connection is idle such as when the user is reading a page that was downloaded earlier. Very often, when a user does not use her connection temporarily (i.e., idle connection), she does not disconnect from the network. While her connection is "idle", a member in the ad hoc network may use this mobile device as a *gateway* to the global network. Consider another example, in which the group members (as in Figure 1) videoconference with some other colleagues over the Internet, or view the news from a server. It is unnecessary to transmit the data as multiple streams with the same content. Instead it is sufficient for one of the hosts with access to the Internet to receive the stream via its wireless WAN connection and multicast it to the rest via the wireless LAN. This host acts as a *gateway* temporarily. Throughout this paper, the term "gateway" denotes any host that acts *temporarily* as a gateway for other hosts in the group. It must have access to the Internet and an wireless LAN interface. The other hosts either do not have a connection to the Internet or have a connection which is saturated at that instant. Also, we use the term "gateway connection" to refer to the wireless WAN connection of the gateway.

Hence, in this environment regular users with wireless WAN connection act as gateways temporarily, unlike traditional networks where the routers do not "walk away". Another difference between this environment and traditional networks is the lack of mechanisms for directing flows to different routers based on criteria such as bandwidth availability. Note that there is no fundamental change in the connection sharing problem if instead of wireless WAN connections, we have wired WAN connections, such as ADSL lines (or cable modem), provided that the dynamic nature of the setup still remains the same.

---

[1] Currently, there are wireless WAN modems of (approximately) 45 Kb/s. A shared 2 Mb/s wireless LAN is typical, (e.g., [2]).

[2] In this work we focus on the first two motivations. We discuss the third one in Section 5.

Under the connection sharing mechanism, the gains for the users with no wireless WAN connection are obvious. However, even the users with a wireless WAN connection can potentially benefit. As we describe in greater detail in Section 5, when users are receiving same data, (e.g., as participants in a multicast discussion), the connection sharing results in a better QoS. The bandwidth requirement for the transmission of all the layers of a multimedia object[3] is usually much higher than the capacity of a single wireless WAN connection. However, if they collaborate and use the aggregate bandwidth of (some of) their connections for the layered multimedia transmission, the video quality can be increased dramatically.

For other cases of sharing a connection with a user without one, the owner may receive financial benefits through a renting or rewarding mechanism. A user may lend a part of her connection, depending on the bandwidth availability and the bandwidth requirements of the flow that need to be served. So, pricing issues may have an important effect on the system operation. A variety of different pricing arrangements[4] depending on the setup and the users' relation (i.e., degrees of collaboration) are possible and make connection sharing desirable, despite the cost and power consumption requirements of keeping them active. The relatively high power consumption when transmitting data may constrain the deployment of connection sharing. On the other hand, the power consumption for wireless modems is decreasing and the number of electrical outlets in places where we expect the system to be used (such as conference rooms, trains, airports) is increasing. Note, also, that wide-area wireless generally is more expensive (subscription fees, at least) than the local-area one (e.g., infrared, Bluetooth, WaveLAN, etc.). Thus, if someone, while moving, uses the network infrequently, "leasing" a temporary gateway is more efficient.

In this work, we concentrate on the basic components of the architecture and study its performance. The contribution of this paper is the design of a novel system that provides dynamic resource sharing among collaborating hosts. The four main components of the system are: admission control at the gateways, a mechanism that assists hosts to select a gateway while ensuring load (i.e., traffic) balancing across the gateways, traffic measurement mechanism at each gateway, and the gateway availability announcement mechanism. We present performance measurements of the system through simulation results. Specifically, we consider a time snapshot in which a fixed number of gateways provide their wireless WAN connection to serve hosts in the wireless LAN that request access. The requests correspond to various services and generate control and data traffic in the wireless LAN and at the gateway connections. We measure the bandwidth utilization (and the gains from statistical multiplexing) and the packet dropping rates at each gateway connection. We found that the bandwidth utilization varies from 21% to 82% and the dropping rates from 0% to 10% depending on the traffic model characteristics. The traffic overhead due to the control messages exchanged in order to enable the sharing is very low. It contributes around 0.9 kb/s to 1.8 kb/s to the wireless LAN (compared to the wireless LAN bandwidth capacity that ranges from 1.2 Mb/s to 6 Mb/s depending on the technology). Finally, the selection mechanism that the hosts use to choose a gateway achieves load balancing across the gateways. The load balancing metric, as defined in Eq.1, ranges from 1.7% to 4.6% (with 0% "perfect" load balancing).

The remainder of this paper is organized as follows. In Section 2 we discuss related work. Section

---

[3] The design of tools for video conferencing services, conference controller and QoS control mechanism is the focus of papers such as [6], [7]. L. Wu *et al* [8] and S. Floyd *et al* [9] investigate the layered video transmission.

[4] An example of such pricing arrangement would be a "bandwidth co-op" scheme.

3 gives an overview of the connection sharing system. Section 3.1 describes the measurement of the gateway traffic and the announcement policy. In Section 3.2 we present the gateway selection mechanism that ensures load balancing across the gateways. Section 3.3 discusses the admission control policy at the gateways and Section 3.4 evaluates the protocol overhead. Section 4 presents simulation results. Finally, in Section 5 we summarize our conclusions and discuss directions for future work.

2 Related Work

There has been work on the deployment of a combination of wireless networks of different technologies. For example, M. Stemm and R.H. Katz [10] considered a hierarchy of network interfaces that included combination of wireless network interfaces, spanning in-room, in-building, campus, metropolitan and regional cell sizes. Their main objective was to enable a user to roam among multiple wireless networks in a manner that was transparent to applications and reduce the handoff disruption. They were focused on performance issues for vertical handoffs, i.e., handoffs between base stations that were using different wireless network technologies. The MosquitoNet project [11] addressed the multiple connectivity management on mobile hosts, i.e., the need to support multiple packet delivery methods simultaneously and the use of multiple network devices for both availability and efficiency reasons. Multiple interfaces were not available at any point in time, just the “best” interface that is selected according to a specific policy. Goals similar to those of Stanford’s MosquitoNet, InfoPad and Daidalus project (e.g., [12]), were also discussed in [13]. While these groups focused more on mobile IP implementations, J. Inouye *et al* [13] were dealing more with dynamic reconfiguration policies.

There is a large amount of work focused on routing protocols to support mobility and some on ad hoc mobile networks [14], [15], [16], [17], and [18]. S. Lee and A. Campbell [19] presented a signaling system for supporting quality of service in mobile ad hoc networks. It has been designed to support the delivery of adaptive real-time services and includes fast session reservation, restoration and adaptation algorithms between source/destination pairs.

However, to the best of our knowledge, there is no paper in the wireless environment we describe that allows collaborating hosts to share their WWAN connections to increase the data availability and QoS while guarantee a load balancing across the gateways. Under this network connection sharing framework, we would like to exploit further the nature of collaborative applications that support scalable, multimedia, streaming data, such as layered video.

3 Overview of the Connection Sharing

Before proceeding with the overview of the connection sharing system, let us state our assumptions for the setup:

- In general, laptops and base stations can operate in the system as gateways. In this setup, we assume that all the gateways are laptops with wireless WAN connections of the same

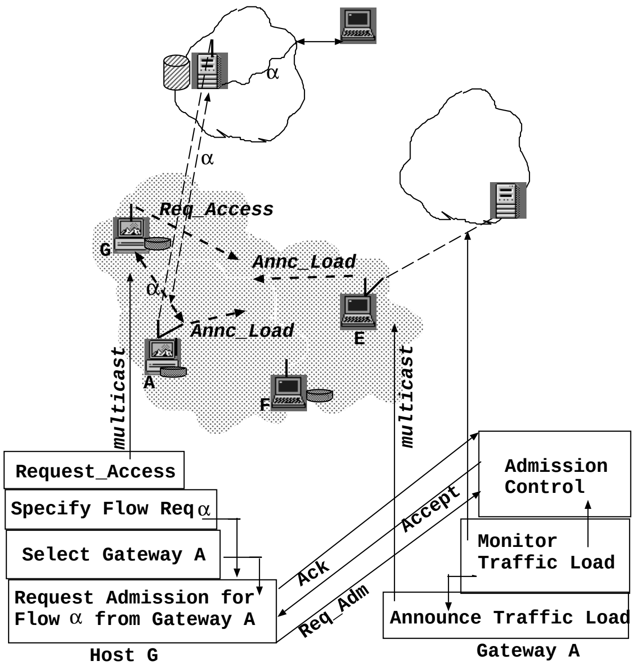

Figure 2: Overview of the communication protocol that enables network connection sharing.

bandwidth. The system is not restricted to support only hosts with the same bandwidth capabilities or to support only laptops, but we assume this for simplicity of exposition.

- A host may leave the multicast group or stop acting as a gateway without prior notification.
- The system treats all the gateways uniformly. There is no "bias" or restriction factor for the selection of a gateway other than the traffic load on that gateway[5]. As we describe in Section 3.2 we aim at a sharing which guarantees load balancing across the gateways.
- There is a routing protocol that enables the communication in the wireless LAN.

As we mention in Section 1 and illustrate in Figures 1 and 2, hosts create a wireless LAN dynamically and communicate in order to collaborate and share their network connections. They communicate by sending messages and listening to a well known multicast address. They multicast requests for accessing the Internet. The gateways multicast announcements of their measured traffic load and availability. In order a gateway to share its wireless WAN connection with a host, for a given flow, the gateway needs to decide if there are sufficient resources. For that, an admission control takes place[6]. We would like to emphasize that the need for admission control depends on the resource sharing/renting characteristics. For network connection sharing without any guarantees, the host

---

[5] A possible extension would be to generalize the selection policy to enable the user to define preferences or restrictions in the choice of the gateway.

[6] Note that the control messages of the admission control are unicast via the wireless LAN.

may operate temporarily as gateway in a best-effort fashion. In such case no admission control is required. However, if there is a pricing mechanism that charges the user who "rents" the resource, then some form of admission control is needed.

Figure 2 illustrates an overview of the communication protocol that takes place among the group members and enables the connection sharing. For example, host G requests an access over the Internet for flow α with peak rate r_α. It queries for available gateway by sending a Request_Access multicast message. As we describe in Section 3.1, the gateways (e.g., hosts A and E) announce the measured traffic on their WWAN link. G waits for time T_c to collect the gateway announcements and selects a gateway. Let us assume that it selects gateway A. Then, it sends directly to A a Request_Admission message to share its wireless WAN connection. In this unicast message, it includes the peak rate of the flow, r_α. Upon receiving a Request_Admission, the gateway decides to accept or refuse to serve the flow. It sends the decision to G and host G sends back an acknowledgement.

We would like to discuss in more detail the main components of the system:

1. Traffic load estimation of a gateway, i.e., the bandwidth utilization of the wireless WAN connection over a sampling period (in Section 3.1).
2. The gateway policy for announcing traffic load (in Section 3.1).
3. The criteria the hosts use to select a gateway (in Section 3.2).
4. The admission control mechanism at the gateway (in Section 3.3).

There are some additional architectural issues closely related to the security mechanism and the pricing arrangement for realizing the network connection sharing or leasing. In this work, we concentrate on the basic components of the architecture. The security and pricing issues are topics of future work.

3.1 Measurement and Announcement of Gateway Traffic

Each gateway is capable of estimating periodically the load of the wireless WAN. It computes an average load every sampling period S (few hundrends of ms). The most recent estimated average load is the value of the traffic load that a gateway announces.

There are two possible announcement policies for the gateway traffic load:

- **Policy I**: The gateway periodically (every T_a sec) multicasts its traffic load to the group.
- **Policy R**: The gateway multicasts its estimated traffic load (that corresponds to the last sampling) *only* in response to Request_Access message.

The purpose of announcing the traffic load is to let the hosts know about the available gateways and select the appropriate gateway to share its connection to the Internet. As we discuss further in Section 3.2, the selection assists in load balancing the traffic across the gateways. We need to emphasize that the selection of the gateway is *not* an admission control mechanism. It only indicates the gateway the querier should contact for admission control. In the future, we plan to include pricing information in these announcement messages as part of a pricing mechanism. This would enable a "leasing" or "bandwidth co-op" scheme for the network connection sharing.

3.2 Gateway Selection Mechanism

As we mentioned a host may request access to the Internet for a specific service. For that it selects a gateway, by listening the multicast announcements of the gateways of their estimated traffic load. The selection has to ensure some load balancing requirements across the gateways. In this work, the load balancing criteria is the reduction of the maximum difference in the average load over a time period (snapshot), τ, across the gateways. More specifically, the load balancing metric we consider is

$$\sigma \;=\; \frac{\max_i \{L_i(\tau)\;\} - \min_i \{L_i(\tau)\}}{b} * 100\% \tag{1}$$

where $L_i(\tau)$ is the average traffic measured at the gateway i over the time interval τ and b is the bandwidth capacity of the gateway connection (i.e., bandwidth of wireless WAN link).

We assume no knowledge of the arrivals of future request or their duration. The problem of minimizing σ is a hard one, due to its on-line nature and the burstiness of traffic. We suggest a *greedy* algorithm and show (through simulation results) that we can achieve a fairly balanced system for different types of traffic. A host that requests access to the Internet chooses the *least loaded gateway*, based on the traffic load value included in the most recent announcement that the gateways have sent.

The low operational cost, its simplicity and its good performance make the greedy algorithm an attractive choice for the system. We investigate its performance through simulations for a variety of traffic models such as the the Exponential and the Pareto distributions. For both Exponential and Pareto distributions the greedy algorithm performs well: σ ranges from 1.7% to 4.6% (as defined in Eq.1). Section 4 presents the traffic models and the results in detail.

Lastly, we would like to comment that, in general, Eq.1 is not a representative metric for load balancing, since it does not capture the potential skewness of the load across the gateways. It has been used mostly to express a "fairness" criteria. However, in cases that its value is very small, as it is in our simulations, it also ensures that the system is load balanced.

3.3 Admission Control Mechanism

The gateway may provide some service guarantees to ensure that sufficient resources are available to serve the flows. For that, the system applies admission control. The criteria to choose an admission control mechanism are:

- low complexity, easy implementation and low operational cost

- high bandwidth utilization
- designed for adaptive, real-time applications that can tolerate variance in packet delays and some packet loss.

Notice that due to the dynamic nature of the system where gateways may "walk away", strict QoS guarantees cannot be made. An admission control with strict QoS guarantees does not match with the characteristics of this system. The admission control algorithm we choose for the system is the "Measured Sum" algorithm [20] [7]. The "Measured Sum" algorithm has low operational cost, promises high bandwidth utilization and does not make strict guarantees.

In the Measured Sum algorithm, each gateway uses measurement to estimate the load of existing traffic and it admits the new flow requested by a host if

$$\hat{v} + r_\alpha \leq u * \beta \tag{2}$$

where u is a user-defined utilization target, β the bandwidth capabilities of the gateway, $\hat{v}$ the measured load of existing traffic, and r_α the rate requested by flow α.

As mentioned in Section 3, each gateway is capable of estimating periodically the load of the connection (a point-to-point link) via which it accesses the Internet. Specifically, it computes an average load every sampling period S. At the end of a measurement window T_m, the gateway uses the highest average from the just ended T_m as the load estimate for the next T_m window. When a new flow is admitted to the network, the estimate is increased by the parameters of the new request to reflect the worst-cast expectations, and then restart the measurement window. If a newly computed average is above the estimate, the estimate is immediately raised to the new average. At the end of every T_m, the estimate is adjusted to the actual load measured in the previous T_m. As expected, a smaller S gives higher maximal averages, resulting in a more conservative admission control algorithm. A larger T_m keeps longer measurement history, again resulting in a more conservative admission control algorithm (as we illustrate through simulations in Section 4, Table 4).

If a flow is admitted, it is served by that gateway until its completion or termination due to the gateway or host leave. In the case, a flow is rejected, the querier merely drops it, as opposed to queue it and retry later. That is, the system performs in a "drop call loss" fashion rather than "drop call retry".

In Section 4, we run simulations to investigate the performance of "Measured Sum" in this system.

3.4 Connection Sharing Protocol Overhead

The overhead of the protocol is due to the control messages that are exchanged for the resource sharing. It includes traffic announcements (in respect to I or R policy), the request for access (i.e.,

---

[7] Sugih Jamin *et al* [20] discussed several measurement-based admission control algorithms. Our paper has benefited from this work.

Request_Access) and the admission control (i.e., Request_Admission, Accept/Reject, and Ack) messages. Note that these messages contribute *only* to the traffic of the wireless LAN.

Let us notate as B_{proto}^{P}, where $P \in \{I, R\}$, the average overhead in bandwidth, as n_g the average number of gateways that participate in the system (simultaneously), pkt the packet size[8], b the bandwidth of the gateway connection, and f the aggregate (i.e., generated from *all* the participants) flow interarrival time.

$$B_{proto}^{P} = B_{reqacc} + B_{adm} + B_{annc}^{P} \tag{3}$$

where $B_{reqacc} = \frac{1*pkt}{f}$ and $B_{adm} = \frac{3*pkt}{f}$. The overhead of the announcement policy is:

$$B_{annc}^{P} = \begin{cases} \frac{n_g*pkt}{f} & \text{if P=R} \\ \frac{n_g*pkt}{T_a} & \text{if P=I} \end{cases}$$

As it was expected, the difference in the overhead depends on the interval values and the aggregate flow interarrival time. Note also that the flow across the gateways will not saturate the wireless LAN bandwidth network as long as $n_g \leq \frac{B-B_{proto}}{b}$. From this, we can compute the maximum number of gateways in the group , n_g^{max},

$$n_g \leq \frac{B - \frac{4pkt}{f}}{b + \frac{pkt}{f}} \Rightarrow n_g^{max} = \lfloor \frac{B - \frac{4pkt}{f}}{b + \frac{pkt}{f}} \rfloor \; if \; P = R \tag{4}$$

$$n_g \leq \frac{B - \frac{4*pkt}{T_a}}{b + \frac{pkt}{T_a}} \Rightarrow n_g^{max} = \lfloor \frac{B - \frac{4*pkt}{T_a}}{b + \frac{pkt}{T_a}} \rfloor \; if \; P = I \tag{5}$$

From Eqs. 5 and 4, given a typical range of values of B, b, f and T_a, we see that $\frac{B}{b}$ is the dominant term in determining the value of n_g^{max}.

4 Simulation Results

We consider a time snapshot in which a fixed number of gateways operate. Hosts request access to the Internet from the gateways. The requests correspond to various services and generate data traffic, i.e., flows, in the wireless LAN and at the gateway connections. We use the ns-2 simulator [21] to quantify the performance of the system. The performance measurements include the bandwidth utilization and the packet dropping rates at each gateway connection, the protocol overhead and the load balancing characteristics across the gateways. The simulation is parametrized on: the source flow traffic model ($TrafM$), the bandwidth and total link delay of the wireless WAN connection (BW_{WWAN}, $TDelay_{WLAN}$, respectively), the bandwidth and total link delay of the wireless LAN (BW_{WLAN}, $TDelay_{WLAN}$, respectively), the simulation time ($SimT$), the measurement time ($MeasT$), the number of gateways (n_g), the aggregate (i.e., generated from *all*

---

[8] For simplicity of exposition, we assume control and data packets of equal size (i.e., 100 bytes).

the participants) flow interarrival time ($Fint$), the aimed bandwidth utilization of the gateway connection ($Utilp$), the interval size (T_m) and sampling period (S) for measuring the traffic at the gateway connection.

First we describe the simulation parameters and the motivations for their values. The hosts generate *homogeneous* data traffic, i.e., of the same type (CBR, Pareto or Exponential) and flow interarrival time. Our main focus is on Pareto and Exponential data traffic, since they more accurately simulate the actual measured traffic. We, also, run a few tests on CBR data traffic. W. Willinger *et al* [22] modeled measured Ethernet LAN traffic[9] with well known ON/OFF source models, such as Pareto. In [23], they showed that network traffic often exhibits long-range dependence (LRD), with the implications that congested periods can be quite long and a slight increase in the number of active connections can result in large increase in the packet loss rate. Each Pareto traffic does not itself generate LRD. But, the aggregation of Pareto traffic results in LRD.

- Exponential ($TrafM$ = E): ON/OFF model with exponentially distributed ON and OFF times. During each ON period, an exponentially distributed random number of packets are generated at fixed rate p packet/s, with an average OFF time, $IdleT$ (ms), and an average ON time $BurstT$ (ms).
- Pareto distribution ($TrafM$ = P): during each ON period of the Pareto flow, packets are generated at peak rate p packet/s, an average burst $BurstT$ (ms) and an average idle time $IdleT$ (ms). According to [22], the shape parameter of the Pareto distributed OFF and ON times covers the interval $(1, 2)$. The shape-parameter-estimate of the OFF period stays mostly below 1.5.

  In our simulations, the shape parameter for both the ON and OFF periods is 1.2.

Emerging, third generation networks, investigated in Europe under the umbrella term UMTS (Universal Mobile Telephone Services) [4] aim at supporting user bit rates of up to 144 kb/s with wide mobility and coverage and upto 2 Mb/s with local mobility and coverage. This is said to be the 3rd generation wireless system. Currently, there is the 2.5(!) generation which is the result of evolution of the 2nd generation to 100 kb/s data capability. We simulate the wireless WAN connection as a link of bandwidth of 100 kb/s and total link delay in the range of 100 ms or 165 ms [24] and [5]. The total link delay ($TDelay$) is the sum of MAC delay (driver), link layer delay and propagation delay.

The wireless LAN in our testbed is RadioLAN [25] or WaveLAN [2]. We run some actual tests to estimate their bandwidth capabilities. The tests involve two laptops, each with a PCMCIA card, indoors, placed in a distance varying from $2-30$ meters. These two hosts are the only participants of the wireless LAN. The measurements include ftp transfer and bandwidth estimation of a link using *pathchar* [26] and *hop_speed* [27]. The highest value of the RadioLAN link (i.e., between the two laptops) capacity measured was 5.8 Mb/s, running the ftp transfer test[10]. The *hop_speed* estimates the bandwidth to be 4.8 Mb/s. We repeat the tests for the WaveLAN. Using *hop_speed*,

---

[9] This data set includes traffic due to applications, such as ftp, e-mail, WWW and Mbone [22].

[10] We repeat 10 ftp transfers of a large, mpeg-1 file of 33.5 MB from one host to another. The 5.8 Mb/s corresponds to the average bit rate of these tests.

we found the bandwidth of the link to be 1.2 Mb/s. In our simulations, the wireless LAN has a bandwidth of 2 Mb/s, link delay of 64 us and CSMA/CA Mac layer.

We assume no failures or disconnection occur during the snapshot of the test. The group size and the number of gateways remain fixed during that period, i.e., there are no changes due to gateway arrivals or leaves. We experiment with n_g values of 3 and 10. In all the simulations, the announcement policy used is R. Also, in all the simulations, the aggregate flow interarrival time follows an exponential distribution. The aimed bandwidth utilization of the gateway connection is 95%. Throughout the tests, the packet size is fixed at 100 bytes and the buffer size at the gateway connection is fixed at 160 packets.

The measurement time ($MeasT$) indicates when we start measuring the link utilization and the dropping rates. As recommended in [20] Pareto ON/OFF sources require a longer warmup period and a longer simulation time for the LRD effect to be seen, thus we run them (if not otherwise specified) for $SimT = 18000$ s and $MeasT = 10000$ s. The Exponential sources run for $SimT = 3000$ s and $MeasT = 1500$ s.

CBR Traffic

The CBR has rate 64 kb/s. The snapshot of the test is [1500 s, 3000 s]. The aggregate flow interarrival mean is 600 ms. The holding time of the flows follow Pareto distribution with mean of 300 s and shape parameter 2.5. Table 1 presents the measurements on the bandwidth utilization of the wireless WAN. In both the cases the packet dropping rate in the gateway link is 0%.

Num Gateways	Utilization (%)
10	$63.9, 63.4, 63.3, 63, 63.2, 63.7, 63.2, 63.4, 63, 63.6$
3	$64, 63.4, 63.4$

Table 1: **CBR** traffic (bitrate=64 kb/s), $f_{int} = 0.6$ s, $HoldT_P^{mean}$ =300 s, Gateway ($TDelay_{WWAN} = 165$ ms, $BW_{WWAN} = 100$ k/bs), $pkt = 100$ bytes, $Utilp$=95%

We obtain the confidence interval [28] for the average bandwidth utilization of each gateway, the packet dropping rates and the load balancing metric σ (as defined in Eq. 1), for a system with Pareto and Exponential traffic. The wireless LAN consists of 6 hosts, 3 of them act as gateways. Each gateway connection has a bandwidth capacity of 100 kb/s and the total link delay is 165 ms. The measuring interval time is T_m =3 s and the sampling period is $S = 400$ ms. We repeat 64 times the simulation of the Pareto case, and 100 times the Exponential case, each time with a different seed. The two cases differ only in the data traffic model and the flow holding time; In the Exponential case the generated flows follow an Exponential distribution with peak rate of 64 kb/s, average bursty time of 312 ms and idle time of 325 ms. The holding time follows an Exponential distribution, $HoldT_E$ with mean equal to $HoldT_E^{mean}$ =300 s. In the Pareto case, the traffic follows a Pareto distribution with peak rate of 64 kb/s, shape parameter of 1.2, mean bursty time equal to 312 ms and mean idle time of 325 ms. The holding time for Pareto traffic, $HoldT_P$ follows a Pareto distribution with an average of $HoldT_P^{mean}$ =300 s and shape parameter equal to 2.5 [29], [30].

Before proceeding with the exposition of our results, let us first show how we measure the load balancing metric σ (as defined in Eq.1) in the simulations: At the end of each test, we compute the average utilization of each gateway connection over $[MeasT,\ SimT]$, $L_i([MeasT,\ SimT])$, for $i = 1, 2, 3$. From that, we find the maximum difference in the traffic across the 3 gateways and compute σ with respect to Eq.1. We repeat the tests 64 times for the Pareto case and 100 times for the Exponential case, each time with different seed. From these values, we compute the confidence interval for the load balancing metric.

Pareto Traffic Case

In Table 2, we illustrate, for each gateway, the packet dropping rate and link bandwidth utilization. We run simulations for aggregate flow interarrival mean ($Fint$) of 600 ms or 6 s. The 99% confidence interval for the load balancing of the system, σ (as defined in Eq.1), is $[1.63\%, 2.52\%]$ when $Fint$ is 600 ms and $[3.4\%, 4.6\%]$ when $Fint$ is 6 s.

The packet dropping rate is very high (e.g., around 10% when $Fint$ =0.6 s). As already mentioned, the queue at each gateway connection is 160 packets or 16KB. We conjecture that [11], in our simulations, $\mathcal{P}_q$, where

$$\mathcal{P}_q = \frac{k_1}{\beta_q^{\alpha-1}} \tag{6}$$

shows how the packet losses behave on a queue of size β_q. In Eq.6, k_1 is a constant and α is the shape parameter of the Pareto traffic (equal to 1.2). By increasing the queue size (β_q) by 32% (i.e., β_q is equal to 512KB), the packet losses are cut in half to 5%.

Confidence Interval for Pareto Traffic (99%)			
Aggregate Fint	Gateway 1	Gateway 2	Gateway 3
	Link Utilization (%)	Link Utilization (%)	Link Utilization (%)
0.6 s	$[81.14, 82.08]$	$[80.81, 81.62]$	$[80.49, 81.24]$
6 s	$[64.7, 65.8]$	$[63.3, 64.7]$	$[62.6, 64.0]$
	Pkt. Dropping Rate (%)	Pkt. Dropping Rate (%)	Pkt. Dropping Rate (%)
0.6 s	$[9.62, 10.23]$	$[9.64, 10.29]$	$[9.53, 10.12]$
6 s	$[3.7, 4.1]$	$[3.4, 3.9]$	$[3.3, 3.7]$

Table 2: **Pareto** Traffic(64 kb/s, $IdleT = 325$ ms, $BurstT = 315$ ms), **Gateway** ($TDelay_{WWAN} =$ 165 ms, $BW_{WWAN} = 100$ kb/s), $pkt = 100$ bytes, $Utilp = 0.95$

Exponential Traffic Case

In Table 3 we illustrate, for each gateway, the packet dropping rate and link bandwidth utilization. The aggregate flow interarrival mean is 600 ms. The 99% confidence interval of the load balancing of the system, σ (as defined in Eq.1), is $[1.71\%, 2.12\%]$.

Therefore, in both the Pareto and Exponential case, the greedy algorithm performs well: The σ ranges from 1.7% to 4.6% (with 0% "perfect" load balancing).

---

[11] After a discussion that we had with Prof. Predrag Jelenkovic.

Confidence Interval for Exponential Traffic (99%)		
Gateway 1	Gateway 2	Gateway 3
Link Utilization (%) $[66.68, 69.13]$	Link Utilization (%) $[65.85, 68.04]$	Link Utilization (%) $[65.32, 67.63]$
Dropping Pkt. Rate (%) 0	Dropping Pkt. Rate (%) $3 * 10^{-3}$	Dropping Pkt. Rate (%) $2 * 10^{-3}$

Table 3: **Exponential** traffic (peak=64 kb/s), Gateway ($TDelay_{WWAN} = 165$ ms, $BW_{WWAN} =$ 100 kb/s), pkt=100 bytes, $Fint = 0.6$ s

The results in Tables 2 and 3 indicate that the admission control aggressively schedule the Pareto flows, which results to higher bandwidth utilization, in the cost of higher packet dropping rates (LRD effect). In some tests, the dropping rate is around 10%, which is an "unacceptable" level for many services. In the case of Exponential flows, keeping the same $Fint$, the bandwidth utilization is lower than in Pareto case with, also, lower packet losses.

Performance over Time Intervals (T_m (s), S (ms))			
(T_m,S)	Link Utilization (%)	Load Balancing (%)	Dropping Pkt. Rates (%)
(60,400)	$[30.74, 31.43]$	$[2.29, 3.52]$	$[0.0676, 0.12]$
(30,400)	$[36.49, 37.28]$	$[2.61, 3.76]$	$[0.19, 0.28]$
(3,400)	$[80.81, 81.62]$	$[1.63, 2.52]$	$[9.64, 10.29]$

Table 4: **Pareto** traffic(peak=64 kb/s, $IdleT = 325$ ms, $BurstT = 315$ ms), Gateway ($TDelay_{WWAN} = 165$ ms, $BW_{WWAN} = 100$ kb/s), $pkt = 100$ bytes, $HoldT_P^{mean} = 300$ s, $Fint$ =600 ms

Table 4 shows the packet dropping rates and link utilization for a range of values of T_m. Table 4, also, includes the load balancing measurements of the system. As we expect, by increasing the interval T_m and keeping the sampling period fixed, the admission control policy becomes more conservative, since the gateway uses now a longer time period for its traffic measurement. As we describe in Section 3.3, the gateway estimates its load as the maximum over the averages (that it computes for each sample during that period). For larger T_m the system estimates a higher utilization and therefore becomes more conservative. As expected, it results in lower packet dropping rates.

Let us compute the protocol overhead in the wireless LAN for n_g of 3 and 10. As before, the announcement policy is R. As it appears in Eq.3, the overhead depends on the *aggregate* flow interarrival mean. For an aggregate flow interarrival mean of 6 s, and packet size of 100 bytes, the traffic overhead contributes 0.93 kb/s to the wireless LAN. Similarly, for a system with 10 gateways, the protocol overhead is 1.86 kb/s [12].

---

[12] For an aggregate flow interarrival with mean equal to 600 ms, the protocol overhead increases by exactly a factor

5 Future Work and Conclusions

In summary, in this paper we present a framework that enables collaborating, mobile hosts to share their network connections in order to increase their QoS and data availability. We discuss the basic components of the system and illustrate their performance through simulation results. The connection sharing across the hosts is characterized by a bandwidth utilization that varies from $21\% - 82\%$ and a packet dropping rate from $0\% - 10\%$ depending on the system parameters (as we described in Section 4). The gateway selection mechanism, in both cases, achieves load (i.e., traffic) balancing across the gateways. The greedy algorithm performs well: the load balancing metric, as defined in Eq.1, ranges from 1.7 to 4.6% (with 0% "perfect" load balancing).

As mentioned in Section 1, the reduction in the transmission of replicated data is a motivation for the connection sharing. As we illustrate in the following scenario, this results in better utilization of the bandwidth of the wireless WAN connections. In Figure 1, users need to teleconference with colleagues over the Internet. We assume the support of layered multimedia data sources. User A joins the multicast discussion. Due to her low bit wireless WAN connection, he cannot receive more than one layer of the video stream. So, she listens to the first channel that transmits the first layer of the video (e.g., S_1). Later, user E joins the discussion. Similarly, E can also afford only one layer of video. However, instead of listening to the first channel that corresponds to the first layer of the video, she listens to the second one (as soon she becomes aware of A). Due to the dependencies across the layers, most scalable compression schemes require the receiver to decode all the layers. Therefore, they forward to each other the layer just received by the multicast. Both A and E decode the two layers and the video quality increases substantially. This idea can be extended as more users (e.g., host B) join the multicast discussion. Notice that if both A and E broadcast the layer S_1 and S_2 respectively, all the remaining hosts in the ad hoc network will be able to receive both the two layers. A similar scenario takes place in case of channels with different source.

We have implemented a prototype that operates as a multicast application. It runs as a multicast application and it is user-oriented. The users issue requests for Internet access by sending multicast messages to the ad hoc group. The users with available network connection can view the requests and may explicitly select the ones to respond to (as described in Section 3). We are in the process of extending it using a more generic approach that requires less interaction from the user. We consider possible extension of the IGMP-v3 and ICMP router advertisements. Currently most of the multicast routers support IGMP-v2 [31]. IGMP-v3 provides the additional capability of joining a multicast group for a specific source. In this paper, we describe a slightly more general case, in which join takes place in per "channel" or multimedia layer basis. In addition, an extension of the protocols will be required to support multiple routers servicing simultaneously in a LAN (i.e., listening to different multicast groups).

Another feature that we plan to include is a pricing protocol that will enable "leasing" or "bandwidth co-op" schemes and in the same time provide some protection from malicious participants. In addition, we would like to investigate ways in which the local wireless LAN configures itself efficiently (e.g., with low overhead).

---

of 10.

6 Acknowledgements

We would like to thank Wenyu Jiang for the helpful discussions and his bandwidth measurements. Prof. Predrag Jelenkovic provided insights into traffic modeling.

References

[1] Metricom, "The Ricochet wireless network overview," 1999.

[2] WaveLAN, "WaveLAN library." http://www.wavelan.com/support/library.html, 1998. Lucent Technologies.

[3] Bluetooth. http://www.bluetooth.com, 1999.

[4] S. G. Glisic and P. A. Leppanen, *Wireless Communications TDMA versus CDMA*. Kluwer Academic Publishers, 1997.

[5] K. Lai, M. Roussopoulos, D. Tang, X. Zhao, and M. Baker, "Experiences with a mobile testbed," in *Proceedings of the Second International Conference on Worldwide Computing and its Applications*, Mar. 1998.

[6] D. Sisalem, H. Schulzrinne, and C. Sieckmeyer, "The network video terminal," in *HPDC Focus Workshop on Multimedia and Collaborative Environments (Fifth IEEE International Symposium on High Performance Distributed Computing)*, (Syracuse, New York), IEEE Computer Society, Aug. 1996.

[7] H.-P. Dommel and J. Garcia-Luna-Aceves, "Network support for turn-taking in multimedia collaboration," in *Proc. of IS&T/SPIE Symposium on Electronic Imaging: Multimedia Computing and Networking*, (San Jose, California), Feb. 1997.

[8] L. Wu, R. Sharma, and B. Smith, "Thin streams: An architecture for multicasting layered video," in *Proc. International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV)*, (St. Louis, Missouri), May 1997.

[9] S. Floyd, V. Jacobson, C.-G. Liu, S. McCanne, and L. Zhang, "Reliable multicast framework for light-weight sessions and application level framing," in *SIGCOMM Symposium on Communications Architectures and Protocols*, (Cambridge, Massachusetts), pp. –, Sept. 1995.

[10] M. Stemm and R. H. Katz, "Vertical handoffs in wireless overlay networks," *ACM Mobile Networking (MONET)*, 1998. Special Issue on Mobile Networking in the Internet.

[11] X. Zhao and M. Baker, "Flexible connectivity management for mobile hosts," technical report, Stanford University, Sept. 1997.

[12] H. Balakrishnan, S. Seshan, M. Stemm, and R. H. Katz, "Analyzing stability in wide-area network performance," *SIGMETRICS Conference on Measurement and Modeling of Computer Systems*, June 1997.

[13] J. Inouye, J. Binkley, and J. Walpole, "Dynamic network reconfiguration support for mobile computers," in *MOBICOM* (ACM, ed.), 1997.

[14] J. Broch, D. Johnson, and D. Maltz, "The dynamic source routing protocol for mobile ad hoc networks," Internet Draft, Internet Engineering Task Force, Jan. 1999. Work in progress.

[15] S. Corson and V. Park, "Temporally-ordered routing algorithm (TORA) version 1 functional specification," Internet Draft, Internet Engineering Task Force, Dec. 1997. Work in progress.

[16] J. Broch, D. Maltz, D. Johnson, Y.-C. Hu, and J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols," in *Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking (Mobicom'98)*, (Dallas, Texas), Oct. 1998.

[17] Z. Haas and M. Pearlman, "The zone routing protocol (ZRP) for ad hoc networks," Internet Draft, Internet Engineering Task Force, Nov. 1997. Work in progress.

[18] S. Corson and J. Macker, "Mobile ad hoc networking (MANET): routing protocol performance issues and evaluation considerations," Internet Draft, Internet Engineering Task Force, Oct. 1998. Work in progress.

[19] S. Lee and A. Campbell, "INSIGNIA," Internet Draft, Internet Engineering Task Force, Dec. 1998. Work in progress.

[20] S. Jamin, S. J. Shenker, and P. B. Danzig, "Comparison of measurement-based admission control algorithms for controlled-load service," in *Proceedings of the Conference on Computer Communications (IEEE Infocom)*, (Kobe, Japan), Apr. 1997.

[21] K. Fall and K. Varadhan, "ns: Notes and documentation," tech. rep., Berkeley University, 1998. Technical Report.

[22] W. Willinger, M. S. Taqqu, R. Sherman, and D. V. Wilson, "Self-similarity through high-variability: Statistical analysis of ethernet lan traffic at the source level," *ACM Computer Communication Review*, vol. 25, pp. 100–113, Oct. 1995.

[23] V. Paxson and S. Floyd, "Wide-area traffic: the failure of Poisson modeling," in *SIGCOMM Symposium on Communications Architectures and Protocols*, (London, United Kingdom), pp. 257–268, ACM, Aug. 1994.

[24] S. Cheshire and M. Baker, "Experiences with a wireless network in MosquitoNet," in *IEEE Micro*, Feb. 1996.

[25] RadioLAN. http://www.radiolan.com/, 1998.

[26] V. Jacobson, "pathchar – a tool to infer characteristics of Internet paths," in *Mathematical Sciences Research Institute (MSRI) Math Awareness Week (Mathematics and the Internet)*, Apr. 1997. software at ftp://ftp.ee.lbl.gov/pathchar/.

[27] W. Jiang, "Detecting and Measuring Asymmetric Links in an IP Network," Tech. Rep. CUCS-009-99, Columbia University, 1999.

[28] W. Mendenhall, D. Wackerly, and R. Scheaffer, *Mathematical Statistics with Applications.* Duxbury Press, 1994.

[29] D. E. Duffy, A. A. McIntosh, M. Rosenstein, and W. Willinger, "Statistical Analysis of CCSN/SS7 Traffic Data from Working CCS Subnetworks," *IEEE Journal on Selected Areas in Communications*, vol. 12, pp. 544–551, Apr. 1994.

[30] S. Resnick, "Heavy tail modeling and teletraffic data," Technical Report TR 1134, Cornell University, 1995.

[31] W. Fenner, "Internet group management protocol, version 2," Internet Draft, Internet Engineering Task Force, Nov. 1997. Work in progress.