Establishing How Many VoIP Calls a Wireless LAN Can Support Without Performance Degradation

Ángel Cuevas Rumín
Universidad Carlos III de Madrid
Department of Telematic Engineering
Ph.D Student
Madrid, Spain
acuevasr@inv.it.uc3m.es

Eur Ing Chris Guy
The University Of Reading
School of System Engineering
Head of School
Reading, UK
c.g.guy@reading.ac.uk

ABSTRACT

The use of Voice IP (VoIP) over wireless local area networks (WLAN) is a topic of great interest in the research arena. This paper attempts to answer the question: how many VoIP calls can be established over a WLAN 802.11b network before the call quality becomes unacceptable?

In order to answer the question we carried out a large number of simulations of both infrastructure and ad-hoc networks, with different background traffic conditions. In each case we measured the key performance criteria: packet delay, jitter and lost-packet rate. This paper presents some of the key results and provides a number of answers, depending on the precise conditions. It turns out that for infrastructure networks it is unlikely that more than 5 VoIP calls can be supported before some of the calls experience unacceptable performance limitations. For ad-hoc networks the situation is less clear, as it depends very much on the exact routing used. However the likely limit is similar.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication network**]: Network Architecture and DesignWireless communication

General Terms

Performance

Keywords

Ad-hoc, Infrastructure, WLAN, VoIP

1. INTRODUCTION

Two kinds of WLAN architectures or operation modes exist. Infrastructure mode uses a centralized coordination station, usually called an Access Point (AP), for the scheduling of transmisions. All traffic goes via the Access Point. An ad-hoc mode network works without this centralized element, therefore it needs a routing protocol to provide reliable end-to-end communications between users.

The most widely adopted architecture for infrastructure and ad-hoc wireless networks is based on the IEEE 802.11 standard [2], which was initially designed for data services. Therefore, it has not got ideal characteristics for real-time communications like video or voice. Much research has been carried out with the intention of extending the services offered from data to real time communications in wireless networks.

In particular, the protocol IEEE 802.11b was used in our simulations.

Voice over IP (VoIP) is a reality nowadays, every day more and more people use this system to phone around all the world. There are many common programmes which make it easy to use VoIP: Skype, MSN messenger, VoIPcheap, VoIP-buster, etc. They are more used every day because they offer a good quality and specially because they are cheap, even free in some cases. Hence, VoIP is starting to be a very widely used technology. The principal telephony service provider companies have realized this and most are now offering VoIP services, specially to enterprises. Many organisations are using WLANs as the first hop in there networking strategy, so it is important to investigate how VoIP over WLAN performs. An example realistic scenario is often the best way to investigate a problem: A university department which uses WLAN as its local area network wants to install VoIP. Then, all the calls between lecturers inside the department will use VoIP over WLAN. In this case, it would be interesting to know what are the performance limitations.

Trying to establish a clear objective in our research from the beginning, we established the principal objective as a question: How many VoIP calls can be established over a 802.11b WLAN?. To answer this we measured three quantities which are the most important parameters involved in real-time voice communications, assuming that raw data throughput is not a problem. The first parameter is packet delay, which is a measure of the time taken by a packet to travel from an originating node to a destination node. The VoIP maximum delay recommended by ITU-T G.114 is 150 miliseconds [1]. The second parameter is packet jitter, which is a measure of delay variability. For this purpose we measured the time between packet arrivals. We checked the time variability between arrivals, because for jitter the average is not important but the variance is. The last parameter is lost packet rate. This is a measure of how many packets are dropped in a particular conversation. The results are given

WMuNeP'06, October 6, 2006, Torremolinos, Malaga, Spain.

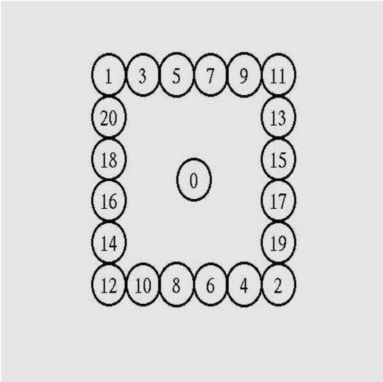

Figure 1: Infrastructure Mode Real Scenario

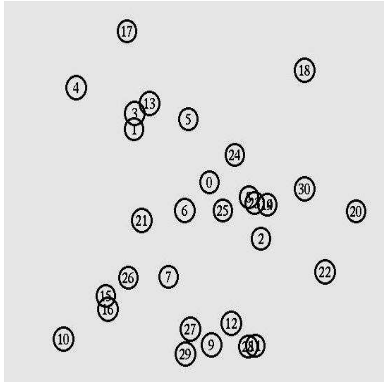

Figure 2: ad-hoc Mode Real Scenario

as a percentage, which will give some clue as to the impact on quality.

2. THE SIMULATED SCENARIOS

2.1 VoIP Traffic

First of all, we needed to decide on the VoIP traffic mechanism because there are many differents types. Our choice was to use a 64 Kbps half-duplex UDP flow, which means that only the node which established the call (caller) sends traffic at first. Then, after a random time, this node stops and the node which received the call (callee) starts to send traffic after a configurable blank time space. The time each node spends sending traffic was randomised. Usually, the node which sent traffic changed several times during the simulation.
To generate a 64 Kbps flow, 64 bytes packets were sent every 8 miliseconds. This value, 8 miliseconds, is important for the jitter, because since we measure the time difference between arrivals, if the environment is not saturated and the calls are working properly the jitter value should be oscillating around 8 miliseconds.

2.2 Scenarios Simulated

Five scenarios were simulated. Three of them under an infrastructure mode architecture and two using an ad-hoc mode network. Some of them are ideal scenarios by which we mean that nodes in these simulations were very close together, something that may not happen in a real scenario. The reason to simulate an ideal scenario was to obtain a baseline maximum number of calls value and which we could then compare with some more realistic scenarios.
The simulated scenarios were:

- Infrastructure mode ideal scenario. AP packet queue length: 50 packets. 20 nodes plus AP.
- Infrastructure mode ideal scenario. AP packet queue length: 100 packets. 20 nodes plus AP.
- Infrastructure real scenario. 20 nodes plus AP (Figure 1).
- Ad-hoc mode ideal scenario. 31 nodes.
- Ad-hoc mode real scenario. 31 nodes (Figure 2).

The key point in the infrastructure mode scenarios is the AP. When more conversations are added the AP has to schedule more packet transmisions, hence its packet queue will lengthen. These packets which are waiting to be forwarded will be delayed. The AP queue has a limited capacity and when the queue is full, if a new one is received it will be dropped.
In the ad-hoc case, a routing protocol has to establish routes among nodes in order to establish end to end user conversations. In this case, there will normally be intermediate nodes in a call. Depending on which nodes establish a VoIP call there will be more or less hops needed for a packet to arrive at the destination node. It is possible that some node in a conversation path will become saturated, because it could be a node involved in many conversations. In this case, all the conversations which use that node will be affected, their packets will be delayed and, if the node packet queue is full, some packets will be dropped.
This is the normal behavior in an ad-hoc network. However, in our simulation there is a special ad-hoc scenario which presents ideal conditions (nodes very close). In this scenario there will not be any intermediate nodes in the conversation path, because the nodes are very close to each other, and only one hop is needed for a packet to reach its destination. Therefore, if there is high delay it is because the environment is saturated and many collisions are happening in the link.

2.3 Simulation Features

The simulation time used was 95 seconds in the four first scenarios, and more than 30 seconds in the last one. This time was enough to obtain the desired results.
In the infrastructure mode scenarios, to obtain results it is enough to analyze one call to understand what is happening in that scenario, even if more calls are running at the same time. Since there is a centralized control element, the AP, and there is no quality of service (QoS) mechanism running, all packets receive the same treatment. Therefore, when the AP queue is saturated, packets from all conversations are dropped on a random basis, and packets from all conversations are delayed in the same way. Knowing this, if one conversation starts to present bad results in delay, or dropped packets appear for that conversation, the rest of conversations will also present the same behaviour and bad quality, and it is enough to check only one conversation.

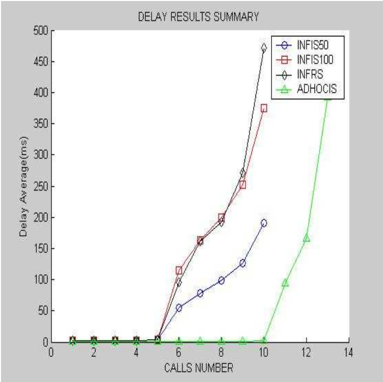

Figure 3: Average packet delay

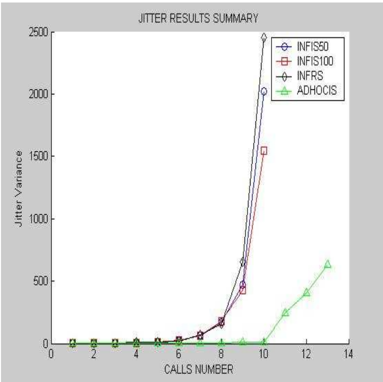

Figure 4: Variance packet jitter

In the fourth scenario, also, it is enough to obtain results for only one conversation to understand the situation. As it is an ideal ad-hoc scenario where all the nodes were very close, the routing protocol establishes one hop path between all the nodes. In other words, one packet needs only one hop to arrive at the destination node. Hence all the conversations suffer the same environment. So if packets from an individual conversation show high delay or are dropped, it is because the link is saturated. It is straightforward to conclude that all the conversations must have the same problems. Hence, one conversation is enough to understand what is happening in a particular simulation.
In the ad-hoc real scenario we had to obtain data for every VoIP call running, because in this case every call has a different path and different number of hops. It is possible that one intermediate node in a conversation is saturated and this conversation will present bad results, whilst other conversations running at the same time do not cross that node, and will be carried successfully.

3. RESULTS

The simulation results are reported in several figures. Figure 3 shows average packet delay vs number of calls running over a scenario. Figure 4 shows jitter variance vs number of calls. Figure 5 shows lost packet rate vs number of calls. These graphs show results for all scenarios except the ad-hoc real scenario.

The first scenario shows an abrupt change when the sixth call is added. The lost packet rate changes from 0% to 10%. This is an unacceptable value, because 10% of the information in a phone call cannot be lost, because the conversation would become not understandable. Also the delay results indicate bad quality, because the average delay is over 50 miliseconds. This means that the delay of many packets is over this value, whereas in the previous cases it has a very low value. The jitter's variance starts to increase at this point (sixth call) but not by too much, but it is a warning showing that something is working worse than in the previous cases. If even more calls are added, all the parameters get worse.

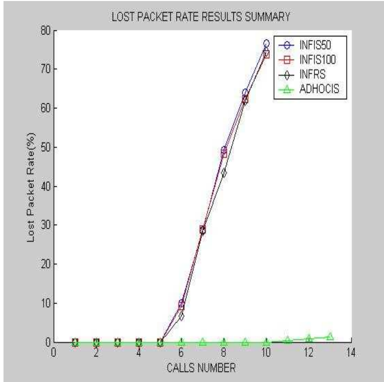

Figure 5: Lost Packet Rate

The second scenario results are very similar. We simulated this scenario because in the previous one we obtained an unacceptable lost packet rate when the sixth call was added. The AP drops packets when its queue is full. Trying to reduce this effect we doubled the AP's queue length. We expected to reduce the LPR whilst paying with a higher packet delay. Results obtained shows that when the sixth call is added the average delay is higher than in the first scenario, more or less the double, which means around 100 miliseconds of average delay. This is an unacceptable value because for much of the time the packet delay will be over 150 miliseconds to obtain an average of 100 miliseconds. However, a lower value of dropped packets does not appear. The result is an LPR of 9% instead of 10% with a 50 packet queue in the AP. The commentary for the jitter's variance is also the same than in the previous case. With the sixth call it starts to increase. Again, very bad results are obtained if 6 or more calls are running over this scenario.

The third scenario is a real infrastructure mode scenario in contrast to the previous ideal scenarios. In this case the nodes are located around the AP, as will occurs in many real cases. However, results in this case are quite similar to the previous ideal results. Again, when the sixth call is added the call quality becomes unacceptable. Specifically, the average packet delay becomes around 100 miliseconds. The lost packet rate is slightly lower than in the previous cases, around 7% but this is still too much information being lost. The variance in jitter again presents the same bahviour than in the previous cases. When we introduce 7, 8, 9 or 10 calls as in scenario 1 and 2 cases the results are really bad.

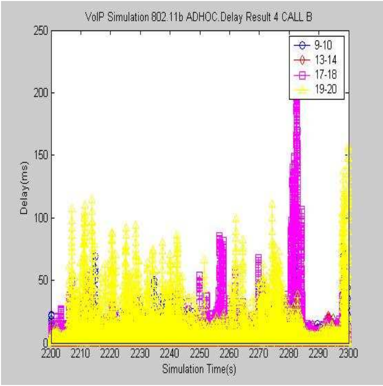

Figure 6: ad-hoc real scenario with 4 calls

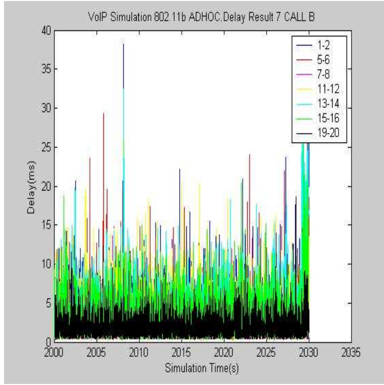

Figure 7: ad-hoc real scenario with 7 calls

Results for the ad-hoc mode ideal scenario are reported in figures 3, 4 and 5. In this case the key step occurs when the eleventh call is added. Now, delay and jitter are the parameters which show unacceptable values. The average packet delay is around 100 miliseconds, and as we explained in some previous cases this is a bad value because, if the average is 100 ms, it is because many packets have a delay over 150 miliseconds which is not acceptable. The jitter's variance presents an abrupt jump between the tenth and eleventh calls. A high variability in the jitter shows bad quality in the communication, in this case very bad quality. The lost packet rate changes from 0% to 0.3%, which is an important change. Since it is an ideal scenario and packets need only one hop to arrive the destination node, if there are dropped packets it is because in the originating node queue there are 50 packets waiting to be sent, and a new one is generated and dropped because the queue is full. Therefore, this low lost packet rate explains why the delay is so high. A quick conclusion is that eleven or more calls cannot be established in an ad-hoc scenario with ideal conditions.

In the ad-hoc real scenario we cannot present a summary result, because the results in every simulation were different. It depends on which calls are running in the scenario and which nodes are involved in these calls. To explain this, figures 6 and 7, which are measuring the instantaneous delay of each packet in the scenario, are shown. In figure 7, seven calls are running properly over the scenario. It is easy to check that packet delay is always under 40 miliseconds and most of the time under 20 miliseconds, which is an acceptable value. However figure 6 shows the results with just 4 VoIP calls running with poor values for the delay, with a packet delay value over 50 miliseconds much of the time in 2 calls, and one of them with peaks which reach more than 150 miliseconds. If the jitter graphs were shown we could see a much more higher variability in the 4 calls simulation. Finally, in the 4 calls simulation there were dropped packets in one call, whereas there were not dropped packets in the simulation with 7 calls.

Other scenarios were simulated, and we found scenarios with 4 calls with good parameters' values and scenarios with 7 calls with unacceptable quality in the conversations.
We did not find a simulation with 8 calls working properly. We simulated many times with different combinations of calls, but no simulation presented good results.
These results are a very good example to explain that in a real ad-hoc scenario it is impossible to obtain a general result, because it very much depends which nodes establish a conversation and which nodes are involved in the path of each conversation, whether a particular conversation will have an acceptable or not acceptable quality
The last commentary in this real ad-hoc scenario is that we found one case when even one call could not be established, because the routing protocol did not find a path between the caller and the callee. Checking the trace output file we saw that all the packets sent from the originating node were dropped because the originating node did not know the following hop.

4. STATISTICAL ANALYSIS

We tried to find if there are some pattern in the delay and jitter statistical distribution. In this way, by only glancing at a distribution from a particular VoIP call we should be able to recognize if the quality of that conversation is good or not.
Following this objective we obtained histograms with packet percentage versus time, for packet delay and jitter as we can see in figure 8 and figure 9.

Figure 8 shows that when we have a simulation with good parameters, the distribution which describes this situation is an exponential distribution in packet delay, because for the most part the packets have a low delay. Obviously the exponential does not start at time = 0, because it is impossible for a packet to experience a delay of 0 seconds. The minimum will be around 1 or 2 miliseconds. If we add more calls and consider the first bad case, 6 calls simulation, the distribution has a peak at low values of delay, which includes about 20% of the packets, but there is an important percentage of packets with high delay values, between 100 and 200 miliseconds. The shape of this distribution is something like a U. If more calls are added, the peak in the low values disappears, and more packets are in the high values, at this moment a distribution similar to a normal distribution can be seen, as the last graph in figure 8 shows. Obviously, this distribution is centered around a high packet delay value.
A different evolution occurs in packet jitter, figure 9. With 4 calls, there is a normal distribution centered around 8 miliseconds, with short and symmetrical queues, which is

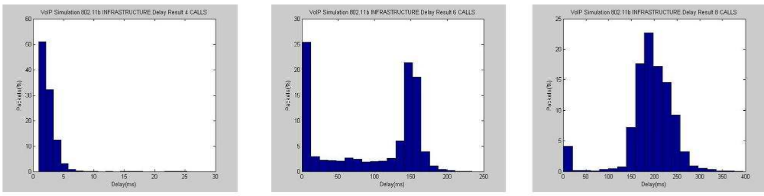

Figure 8: Delay Distribution Evolution. From left to right: Packet Delay Histogram 4 calls, Packet Delay Histogram 6 calls and Packet Delay Histogram 8 calls. All histograms are obtained from infrastructure mode real scenario

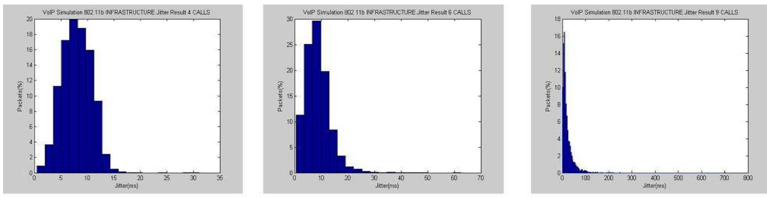

Figure 9: Jitter Distribution Evolution. From left to right: Packet Jitter Histogram 4 calls, Packet Jitter Histogram 6 calls and Packet Jitter Histogram 9 calls. All histograms are obtained from infrastructure mode real scenario

the expected result when the jitter presents a good quality value. In the case with 6 calls, when the conversations start to become poor quality, the queue on the right side is longer and the distribution presents a non-symmetrical shape. With 9 calls, the aspect of the distribution is like an exponential distribution starting in low values and with a very long right queue. Since the important factor for the jitter is the variance, obviously, the highest variance when there are most calls, and the 4 calls graph is the one which exhibits the lowest variance.

With this statistical analysis an interesting pattern model can be established. Looking at one delay or jitter packet distribution for one conversation we will be able to know what is happening approximately, and if that conversation will have good or bad quality by comparing the distribution with one of the previous explained patterns.

5. CONCLUSIONS

In this paper we have evaluated the performance of VoIP calls over 802.11b WLANs, in infrastructure and ad-hoc mode architectures. We have proposed and discussed several scenarios with differents features.

We measured different parameters, involved in every real-time communication, such as packet delay, jitter and lost packet rate, and on the basis of them we have determined the maximum number of VoIP calls which can be supported by a WLAN.

A statistical analysis is presented, to enable us to look for comparison patterns. In this way just by making a comparison with a given distribution we can know, approximately, what is happening in that conversation.

The most important result obtained in all infrastructure mode scenarios simulated is that 5 VoIP calls can be established. When the sixth call is added, in the three infrastructure mode cases, the lost packet rate was not 0% anymore and the average delay increases abruptly, in one case up to 50 miliseconds, in the other two up to 100 miliseconds. Jitter's variance presented a slight increase. Worse results were obtained if more calls were added.

A similar conclusion could be made in the ad-hoc ideal scenario, when the nodes are very close together. In this case 10 calls can be established at the same time without problems. When the eleventh call was added, average delay and jitter variance grows until they reach unacceptable values. At this point a lost packet rate that was not 0% appeared. It was a low rate of 0.3% but with an important meaning because it expalins why the delay is so high.

There are no conclusions with values in an ad-hoc real scenario. The reason is because in this case every simulation made presented a different behavior. In a real ad-hoc scenario it becomes very important which nodes are establishing a conversation and which are the intermediate nodes. Whether the calls can be supported is a function of node distribution and which nodes are involved in the calls. We showed that a network scenario with 7 calls could be established, whereas a different situation with 4 calls had bad quality. We did not find any simulation with 8 calls running properly over this real scenario.

Finally the statistical analysis made shows that the evolution in packet delay goes from an exponential distribution, when there are good performance features in a scenario, towards a normal distribution centering around a high delay value, when bad quality is expected. In packet jitter the evolution goes from a normal distribution centering around a known value towards an exponential distribution.

6. ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for insightful ideas and suggestions that made it possible to write a better article. Also we want to thank IMPROVISA project (TSI2005-07384-C03-027) for its support.

7. REFERENCES

[1] N. G. Corporation. Implementing the voip network. a technical briefing series on voip and converged networks. 3:Page 6, August 2005.

[2] IEEE 802.11-1999 Edition (ISO/IEC 8802-11: 1999). Standards for Information Technology – Telecommunications and Information Exchange between Systems – Local and Metropolitan Area Network – Specific Requirements– Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.