Throughput Bounds for HTTP Services Over WiFi Networks

M. F. Caetano†, P. Vieira††, J. L. Bordim†† and P. S. Barreto††

†Department of Electrical Engineering, ††Department of Computer Science
University of Brasilia, Brasilia-DF, Brazil.

Summary
Access networks based on cooper cables are costly to build and maintain. For this reason, last-mile access networks based on wireless technologies are gaining considerable attention. Among the wireless technologies being employed, the IEEE802.11 (WiFi) is commonplace. In such scenarios it is important to have well defined mechanisms to better evaluate the underline traffic characteristics and overall system performance. Such understanding can help network designers to better estimate the resources needed to provide basic services with a reasonable level of quality (QoS). This task, however, has been shown to be non-trivial. The main contribution of this work is to present techniques than can be applied to estimate the throughput and the access pattern for fundamental services in the context of a WiFi based networks.

Key words: Wireless networks, IEEE 802.11, Throughput Estimation and Traffic Analysis

1. Introduction

Access networks based on wireless technologies are gradually being considered as an affordable option to cooper cables based networks. Indeed, access networks based on wireless technologies have a number of advantages, such as reduced cost to build, maintain, and most important, reduced setup time. These characteristics have fostered a number of initiatives aiming to provide Internet access at places of public interest, such as touristical and historical sites. Indeed, projects in this direction, such as *cyber-city* projects, have been gaining considerable attention in recent years [1].

For obvious reasons, the service providers attempt to provide access to as many people as possible while maintaining a reasonable quality level for the delivered services. When applying wireless technologies for last-mile access, it is important to have well defined mechanisms to evaluate traffic characteristics and means to foresee its performance. This understanding can help network designers to better estimate the resources needed to provide basic services with a reasonable level of quality. To help on this task, a number of works have been focusing on mechanism to estimate the maximum throughput of a wireless setting, including traffic analysis and modeling. Traffic analysis and modeling attempt to estimate the minimum communication channel capacity necessary to provide services to a given number of users over a communication channel. This task must be accomplished while respecting the applications characteristics. The outcomes of these studies have shown that the answers to these questions are non-trivial, particularly due to the difficulty to model the underline channel [3,4,8,11,13,14,16].

In the seminal work of Gupta *et. al.* [8], the authors have shown that the per node capacity of a WiFi network decreases with increasing network size. Biachi *et. al.* [3] presented a Markov-chain model (a.k.a. Bianchi's Model) which allows to estimate the collision probability on saturated environments. The work in [10] extends the model presented in [3] by showing that the throughput of any flow is bounded by the one with the smallest transmission rate. It was also shown that the aggregate throughput is bounded by the reciprocal of the harmonic mean of the transmission rates [10]. Yeo *et. al.* [16] presented an in-depth analysis of the packet error impact on the capacity of a WiFi network. The work also presents a scheme to model the transmission failure probability, the saturation throughput as well as its relation to the packet error probability.

More recently, a simplified model of the work proposed by Bianchi *et. al.* [3] was presented in [14]. This simplified model was then enhanced in [13], which takes into account the channel conditions over a certain period of time in order to compute the collision probability of a future event. Bruno *et. al.* [4] proposed the formulation of Markov chain models to compute the steady-state distributions of the number of active TCP and UDP connections in the network. They developed a comprehensive analysis of the throughput performance of long-lived upstream and downstream TCP connections competing with finite-load UDP upstream flows in a WiFi setting.

In this work we propose mechanisms to estimate the necessary throughput of a WiFi network in order to

Manuscript received July 5, 2012
Manuscript revised July 20, 2012

support a given service. The scenario considered here is similar to the one shown in Figure 1, where a number of

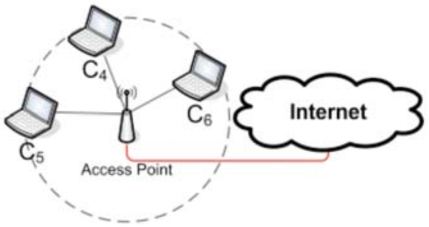

Fig. 1 Basic Service Set in a IEEE802.11b/g network

users are accessing the Internet via an access point. To estimate the throughput in such scenarios, we begin by estimating the available throughput of the shared wireless channel. Next, the user application characteristics are taking into account as well as the added overhead of the TCP/IP layers. With these results, we show that is possible to estimate the minimum bandwidth necessary to provide a reasonable quality of service for all the users, which are sharing a common channel. For that purpose, in this work we estimate the available throughput on wireless settings using the CSMA-CA (Carrier Sense Multiple Access with Collision Avoidance) protocol. We then go on to estimate the characteristics of an essential application. In this work we have selected the HTTP (*Hypertext Transfer Protocol*). More precisely, this work presents results that can help estimate a minimum, and yet acceptable, throughput to support basic services for a group of users on a wireless communication network. The obtained results are expected to help system administration to better estimate the user's needs when deploying WiFi networks.

The remaining of this document is organized as follows. Section 2 presents theoretical models, which will be used to estimate the maximum channel throughput. The impact of the related protocols is also discussed in this section. In Section 3, the HTTP protocol in analyzed. In this section we look into the user requirements when using the HTTP protocol. From the results obtained in the previous sections, we go on to estimate the maximum number of users that can be supported in a shared channel while maintaining a reasonable quality of service that are expected to meet the user needs. Section 4 draws some consideration about this work and points out directions for further investigation.

2. WiFi Networks: Throughput Estimation

In order to estimate the maximum throughput of a communication channel, it is usually assumed a saturated environment where the number of collisions and overall channel conditions are expected to degenerate. The mathematical models used in this work are aimed to such scenarios. In order to simplify the discussions, the following assumptions are used in this work:

- A MAC layer dialog is assumed to be comprised of DATA and ACK frames with frame spacing as defined in the standard [9];
- Fragmentation is not considered;
- The underline channel is error free and noiseless;
- Each TCP Data packet transmitted receives a corresponding TCP ACK.
- On a collision event, the packet is corrupted and discarded by the receiver.

2.1 Background

As can be seen in Fig. 2, as the user's data traverses the TCP/IP stack, from the application layer to the physical layer, a number of headers are added. Suppose that an application layer *message* consists of n_{msg} bytes. When the message is passed to the transport layer, it receives an appropriate header. In what follows the TCP (*Transport Control Protocol*) will be considered. The TCP header comprises of 20 bytes (without the options section). The segment is then passed to the network layer, where additional 20 bytes are added. The datagram is then passed to the data link layer, which is sub-divided into LLC (*Logical Link Control*) and MAC (*medium access control*) sub-layers. According to the IEEE802.11 standard, the LLC adds 8 bytes (3 for the LLC and 5 for SNAP). At the MAC sub-layer, additional 34 bytes are added, and then the frame is passed to the physical layer [10]. Thus, the size of a frame can be expressed as $n_{frm} = n_{msg} + n_{hrd}$, where $n_{hrd} = 82\ bytes$.

At the physical layer, a PLPC header and preamble are added. The standard defines two different types of preamble: the *short* and the *long* preamble. In this work we consider the long preamble, which comprises of 144 bits in contrast with the short preamble that comprises of 96 bits. The header of the PLPC has 48 bits. The TCP protocol is a connection-oriented protocol, which works by acknowledging the transmitted segments. Each TCP ACK can be viewed as a segment at which the $n_{msg} = 0$.

To simplify the discussion, in what follows we consider that each TCP segment receives back its corresponding ACK. In order to obtain the maximum throughput, we consider that the network layer passes to the data link layer a datagram of 1500 bytes (including 20 bytes for the TCP and 20 bytes for the IP). Therefore, the $n_{msg} = 1460\ bytes$. Similarly, the TCP ACK (n_{Ack}), consists of 82 bytes as the $n_{msg} = 0$. At the physical layer, the PLPC *preamble* and the PLPC header are sent at a data rate of 1 Mbps. At this data rate, it takes 192μsec to transmit the 48 bits from the PCPC header plus the 144 bits from the preamble. Then,

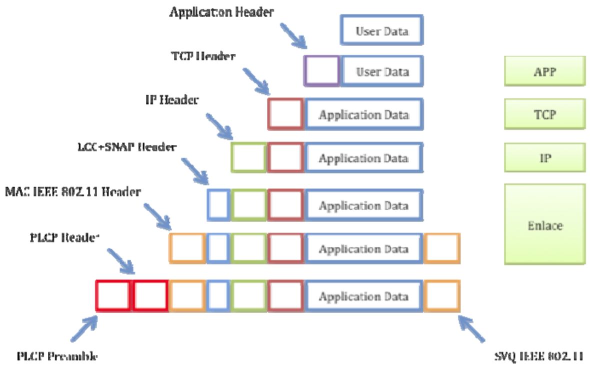

Fig. 2 Data encapsulation process

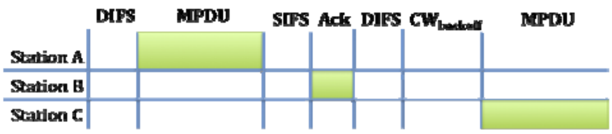

Fig. 3 Transmission sequence for IEEE802.11b/g networks

the physical layer time can be represented as $T_{phy} = T_{PLPC} + T_{preamble}$.

2.2 WiFi Networks: Single-User Maximum Throughput

This subsection is devoted to estimate the maximum throughput for a single-user in an IEEE802.11 network. To this end, we begin looking at the frame space intervals and control packets involved in a IEEE802.11 transmission sequence. According to the IEEE802.11, the *frame spacing* is defined as follows:

- Time Slot (*TS*) = 20μsec;
- Short Inter Frame Space (*SIFS*) = 10μsec;
- DCF Inter Frame Space (*DIFS*) = 2 * *TS* + *SIFS*.

Fig. 3 depicts the frames spacing during the transmission of a MAC PDU (MPDU). The figure also shows the DIFS, SIFS, ACK and CW (*Contention Window*). The IEEE802.11 ACK is fixed in 14 bytes. Thus, to complete one IEEE802.11 transmission, without the use of Request to Send and Clear to Send (RTS/CTS) frames, can be expressed as in the Eq. 1:

$$T_{802_{su}}(n_{msg}, rate) = DIFS + T_{Data} + SIFS + T_{Ack} + T_{Bo_{su}} \quad (1)$$

In Eq. 1, T_{Data} and T_{Ack} represent the time taken to transmit the IEEE802.11 Data and Ack frames, respectively. The $T_{Bo_{su}}$ is the backoff time. Note that, in fact, T_{Data} and T_{Ack} include the synchronization time and subject to the channel data rate. Eq. 2 and Eq. 3 allow to compute the time to send T_{Data} and T_{Ack} respectively.

$$T_{Data} = T_{phy} + \frac{n_{frm}}{rate} \quad (2)$$

$$T_{Ack} = T_{phy} + \frac{Ack}{rate} \quad (3)$$

Eq. 4 shows the average backoff time for a single user environment, where the $CW_{\min}$ represents the minimum contention window.

$$T_{Bo_{su}} = \frac{CW_{\min} * TS}{2} \quad (4)$$

Recall that a TCP transaction requires two transmissions, one to send the datagram and another to receive the corresponding TCP ACK. Thus, when viewed at the IEEE802.11 MAC layer, time to complete a TCP transaction can be expressed as in Eq. 5 below:

$$T_{TCP_{su}}(n_{msg}, rate) = T_{802_{su}}(n_{msg}, rate) + T_{802_{su}}(0, rate) \quad (5)$$

Eq. 5 computes the time to send the TCP Data and its corresponding TCP ACK. The maximum throughput depends on the amount of data transmitted in a T_{TCPsu} time. Thus, the maximum throughput for a single station in an IEEE802.11 network is can be expressed as follows:

$$T_{put_{su}}(n_{msg}, rate) = \frac{n_{msg}}{T_{TCPsu}(n_{msg}, rate)} \quad (6)$$

This sub-section discussed transmission characteristics involved in the case of a single-user environment. The next sub-section extends the above results to the multi-user environment.

2.3 WiFi Networks: Multi-User Environments

As the number of users increases in a BSS cell, so does the number of collisions and subsequent retransmissions, which in turn, reflects negatively on the throughput. Let p_e denote packet collision probability and CW_{min} denote the maximum contention windows (*i.e.*, the first interval). Then, the mean contention window ($CW_{backoff}$) depends directly on the number of collisions, as it doubles on each packet collision. In [16] and latter in [13], it was shown that the mean $CW_{backoff}$ is given by:

$$CW_{backoff} = \frac{1 - p_e - p_e - (2p_e)^m}{1 - 2p_e} * \frac{CW_{\min}}{2} \quad (7)$$

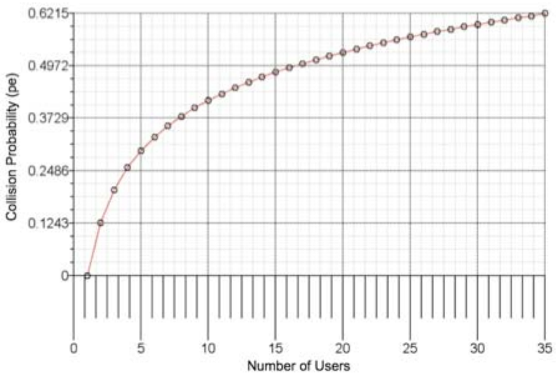

Fig. 4 Collision Probability with $p_e(CW_{min} = 32)$

In order to transmit a packet, the transmitting station has to wait until its $CW_{backoff}$ is count down to zero. Thus, collisions can only occur when two stations have reached zero at the same time. The probability that a packet sent by station **A** will collide with that of station **B** is given by $1/CW_{backoff}$. The results in [6, 13, 16] have shown that the collision probability can be computed as in Eq. 8.

$$p_e = 1 - \left(\frac{1}{CW_{backoff}}\right)^{n_{usr}-1} \quad (8)$$

Fig. 4 shows the collision probability as a function on the number of transmitting stations computed with Eq. 8. In the figure, the CW value was set to 32, which reflect the minimum contention window for IEEE802.11b networks. As it can be seem the collision probability is over 60% with 35 stations.

In [16], it shows that at in saturation environments, most transmissions are preceded by a minimum backoff of CW_{min} . When n stations uniformly choose a time in $[0, CW_{min}]$, the separation between choices has the mean $CW_{min}/(n_{usr}+1)$, where n_{usr} represents the number of users trying to communicate. It should be noted that the stations that pick the earliest slot breaks the channel silence. Let $T_{Bo_{mu}}$ represent earliest slot. The $T_{Bo_{mu}}$ can be expressed as in Eq. 9 below:

$$T_{Bo_{mu}} = \frac{CW_{\min} * TS}{n_{usr}+1} \quad (9)$$

Thus, the overall transmission time ($T_{TCP_{mu}}$) when a number of users are in the same BSS, can be computed by replacing $T_{Bo_{su}}$ (in Eq. 4) to $T_{Bo_{mu}}$ (Eq. 9) in Eq. 1, as follows:

$$T_{TCP_{mu}}(n_{msg}, rate) = T_{802_{mu}}(n_{msg}, rate) + T_{802_{mu}}(0, rate) \quad (10)$$

Yeo *et. al.* [16], have shown that a better throughput estimation, when considering multiple users, can be obtained by multiplying a *factor*, represented here by p_{factor}, which is defined below in Eq. 11.

$$p_{factor} = \frac{2*(1-p_e)}{2-p_e} \quad (11)$$

Thus, the maximum throughput for a multi-station scenario can be estimated using Eq. 12, shown below.

$$Tput_{mu}(n_{msg}, rate, n_{usr}) = \frac{n_{msg}}{T_{TCP_{mu}}(n_{msg}, rate, n_{usr})} * P_{factor} \quad (12)$$

Fig. 5 shows the maximum throughput for the TCP protocol using the above equations in a IEEE 802.11b networks. In the figure, the transmission rates of 1, 2, 5.5 and 11 Mbps are shown. As expected, the maximum theoretical TCP throughput reached its maximum value when $n_{usr} < 3$, operating at 11 Mbps. In this case, the maximum throughput is about 4.7 Mbps. Fig. 6 shows that the relation between packet sizes and maximum throughput. The figure shows packet sizes varying from 256, 512, 1024 and 1460 bytes. The best results have been obtained for larger packet sizes. With packets of 256 bytes, the throughput is below 1.4 Mbps.

2.4 Maximum Throughput Estimation

The overhead introduced to the transmission process in an IEEE802.11 environment is independent on the data size being transmitted. Indeed, the values for SIFS, DIFS, TS, T_{PLPC}, CW_{min} and IEEE802.11 ACK are fixed. In a WiFi based network, the time spent to transmit a symbol equals to $8/rate$, where $rate$ is the transmission data rate of the underline channel, when 8 bits per symbol are considered.

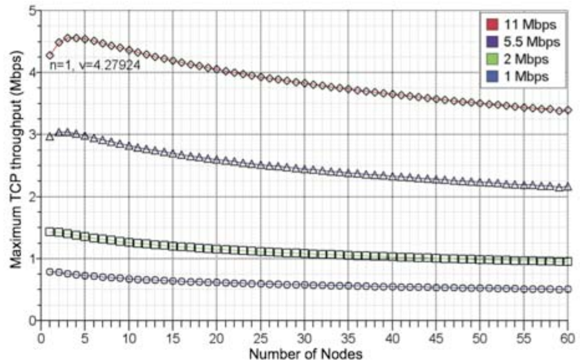

Fig. 5 Maximum TCP throughput for a IEEE802.11b network with a varying transmission rate.

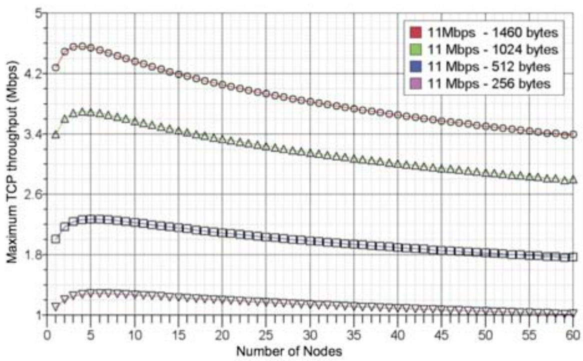

Fig. 6 Maximum TCP throughput for different size of n_{msg} in a IEEE802.11b network.

In what follows, the values expressed in Table 1 are used to compute the maximum theoretical throughput ($Tput_{mu}$) for different data rates. In an IEEE802.11b network, the possible data rates are 1, 2, 5.5, and 11 Mbps. Fig. 5 shows the maximum throughput for a TCP connection on such networks for a varying number of users. It can be seen in the figure that less than 50% of the channel is effectively used. As mentioned before, the maximum throughput is obtained when the data rate is set to its maximum value allowed by the system. Fig. 6 shown the maximum throughput for different values of n_{msg}, which are set to 256, 512, 1024 and 1460 bytes.

In the IEEE 802.11g, the physical layer adds 6 bits at the end of each frame. The physical layer divides the frames into blocks of 216 bits. Here, each symbol takes $216/54\ Mbps = 4\ \mu sec$ to be transmitted. The minimum contention window for IEEE 802.11b/g networks is drawn from [0, 15] while in IEEE 802.11b it is drawn from [0, 31]. Fig. 7 and Fig. 8 show, respectively, the maximum throughput for different transmission data rate and packet sizes for IEEE 802.11b/g networks.

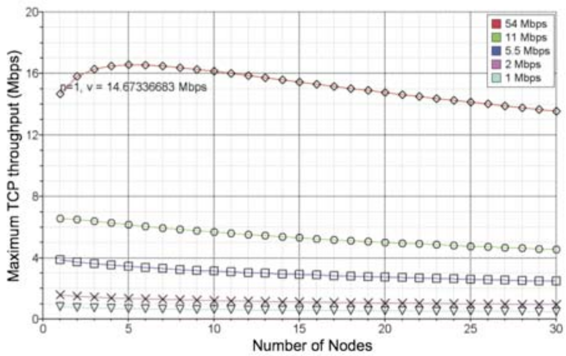

Fig. 7 Maximum TCP throughput for a IEEE802.11 b/g network with a varying transmission rate.

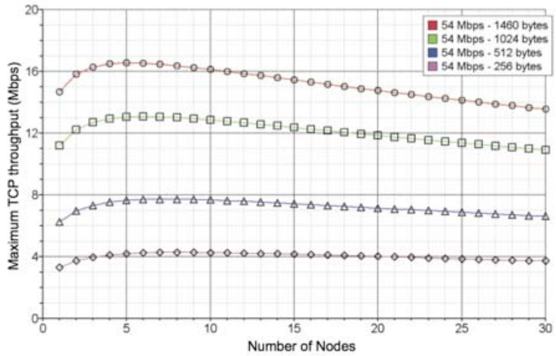

Fig. 8 Maximum TCP throughput for different size of n_{msg} in IEEE802.11 b/g network.

Table 1 summarizes the parameters used in IEEE 802.11b/g networks.

Table 1: Parameters for IEEE 802.11 b/g networks.

TS	20 µsec
MAC	272 bits
DIFS	50 µsec
SIFS	10 µsec
Ack	112 bits
CWmin 802.11b	31
CWmin 802.11g	15
CWmax	1023
Rate (802.11b)	1, 2, 5.5 and 11 Mbps
Rate (802.11g)	12, 18, 24, 36, 48 and 54 Mbps

3. HTTP Protocol Profile

As shown in the previous sections, the throughput of a WiFi based network depends on several factors, among them, the number of collisions which is directly associated with the number of users competing for the channel. There are a number of other factors that may impact on the throughput as well, such as multi-path fading and noise. In the context of this work, we are concerned on the number of stations competing for the channel and their transmission pattern to obtain a fair share of the available bandwidth in order to support a given service. Such understanding is important to help network designers and administrators to identify the overall system capacity and as well as it limitations. Identifying the application limits and constraints, in terms of jitter and delay for instance, becomes crucial to determine the infrastructure requisites. Among the available applications, the HTTP one of the most used protocol on the Internet. The HTTP serves as the start point to other applications as well, such as file search, downloads (FTP), reading emails, e-commerce, etc. Due to its importance, even when other applications fail, the HTTP is likely to be preserved. For the reasons stated above, in this work we consider the HTTP as the basic service to be delivered in a WiFi BSS cell. In what follows, we will discuss the HTTP protocol and the its limits in order to identify the maximum number of users that may be co-located in a BSS cell and yet respecting the QoS requisites to support the HTTP protocol.

The HTTP is based on a client-server infrastructure. The HTTP defines a set of messages, which are used for client-server communication. The HTTP messages can be classified in two classes: (*i*) messages, which are used by the client to issue requests to the server, and (*ii*) messages replies sent by the server. According to the application needs, different types of messages and parameters can be exchanged by the client and the server.

To estimate the necessary bandwidth to support HTTP applications, it is necessary to known the mean size of an HTTP message. According to [2], the mean size of an Internet web page has increased more than 230% over the last 5 years. Also, the number of objects, such as images, present on a page, has double in the same period. In the work done in [12], where more than 340 million packets were analyzed, it was found that the average Internet packet size is about 402.7 bytes. Thus, one may consider that, when a user requests a page, it will be divided into packets of roughly 400 bytes. That is, $n_{msg}/402.7$ packets. For the sake of simplicity, in what follows we consider the mean packet size on the Internet to be 512 bytes.

Besides the average message size, it is also necessary to know the average waiting time to request and retrieve an object. According to [2], the mean waiting time to retrieve a Web page has decreased from 2.8 sec, in 2006, to 2.33 sec in 2008. Also, in [15], it has been shown that users tolerate a waiting time of about 2 to 3 seconds to receive the requested information. The same report stated that in 2001 this value was around 10 seconds. In [2], it was also shown that the average Web page size is about 312 Kbytes. Thus, with a download tolerance of 3 seconds, a bandwidth of 104 Kbytes/sec (or 0.832 Mbps) would be necessary to meet the user expectancy. It is worth mentioning that the HTTP *Timeout*, which is defined by the application, has a recommended value of 30 seconds. Fig. 9 shows the maximum throughput for the HTTP protocol, per user, as function of the waiting time. Eq. 13 formalizes the above discussion.

$$Tput_{HTTP}(T_{tolerance}) = \frac{\frac{n_{HTTP}}{n_{pkg}}}{T_{tolerance}} * n_{pkg} \qquad (13)$$

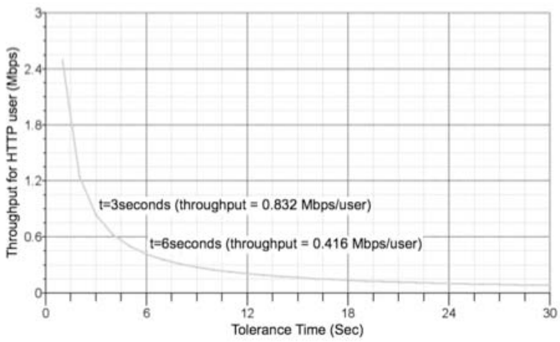

Fig. 9 HTTP throughput for different values of $T_{Tolerance}$.

3.1 HTTP User Access Pattern

In the previous section, we have seen that the each user demands for at least 104 Kbps to have a reasonable quality of service to browse the Web. Suppose that a given BSS consists of n_{usr} users, each of them being active for $1/l$ of the time, where $0 \leq l \leq 1$. In this case, *what would be the maximum number of active users that can be supported by the system?* The section shows a simple way to answer this question. Using equation Eq. 11, it is possible to verify the maximum TCP throughput when a number of users are accessing the channel concurrently. As can be observed, the number of users that are able to browse the Web concurrently is subject to the variables shown in Eq. 14.

$$n_{HTTP_{usr}} = \frac{Tput_{mu}(n_{msg}, rate, n_{usr})}{Tput_{HTTP}(T_{Tolerance})} \qquad (14)$$

As an example, suppose that we have a 54 Mbps channel, a data packet of 512 bytes, 9 concurrent users with a tolerance of 104 Kbps (see Fig. 8). In this case we have $Tput_{mu}(512, 54, 9)/104$, which is ≈ 7.5 Mbps, enough to support 9 concurrent users.

It is important to note that in the context of Web objects, the concurrency at which users are sending requests is also related to the time each user will spend consuming the requested object. While reading a Web page contents, the unused bandwidth may be used by another station to request its desired objects. According to [5, 7], the average time spent reading the contents of a Web page, before issuing a new request (that is, a new page or following one of the links), is about 48 seconds. Thus, the channel is used for only 3 seconds with the reminder 45 seconds is spent by the user to read its contents. That is, the channel is used for (3/48)*100 = 6,25% of the time. With the help of equation Eq. 15, it is possible to compute that probability that a number of users will be accessing the channel concurrently.

$$\Pr(x) = 1 - \sum_{j=0}^{n_{usr}} (C(n_{usr}, j) * p^{j} * (1-p))^{(n_{usr}-j)} \quad (15)$$

In Eq. 15, x represents the number of concurrent users from a population of n_{usr} users. This value depends on the probability that there will be j users, $1 \leq j \leq n_{usr}$, using the channel and $n_{usr} - j$ users not using the channel at a give time. The variable p is the probability that a station is active and $1-p$ is the probability that the station is not active. Thus, Eq. 15 allows the system administrator to define the number of users that can be active at a given time within the BSS. Also, as there may be other external factors that can degrade the signal quality and consequently the available bandwidth. Thus, it is a good practice to define a security margin. In other words, the system could be designed to operate with at most 75% of its capacity, allowing 25% to accommodate other external factors that have not been accounted for. Note that, however, the security margin may change from place to place, depending on the external conditions. The percentage of packet disruption could be a way to estimate the security margins. In this work we do not address such issues. In what follows, we give an example of how the above results can be used to estimate the maximum number of users that can be supported with a reasonable QoS on a BSS cell.

As shown previously, in an IEEE 802.11b/g network at 54 Mbps, it is possible to accommodate up to 9 concurrent users as shown in Figure 8. Now, suppose that each user is likely to be active for 6.25% (i.e. p=0.0625) of the time when accessing web contents. Thus, with, Eq. 15 shows that it would be possible to accommodate 125 users in the same BSS. Fig. 10 shows the relation between the number of users and the percentage of the active time. That is, the number of users accessing the Web contents. Clearly, these are optimistic results, as interference and multipath fading, among other aspects that might impact on the throughput are not considered here. On the other hand, a site survey can be helpful in understanding the channel conditions. With such information at hand, one can adjust the parameters in the equations above to take such aspects into account.

4. Conclusion and Future Works

The main contribution of this work was to provide means for network designers and system administrator to estimate the maximum number of users that can be supported in a WiFi based network when the users are consuming basic services. In this work we have focused on widely used application protocol, the HTTP protocol. Our results allow estimating the number of users that can share the same BSS cell when the users are consuming Web content (Web pages). For that purpose, we first show how to estimate collisions in a wireless environment and also analysed the characteristics and requirements of the HTTP protocol. The results obtained in this work are optimistic in the sense that we do not consider transmission errors, noise, multipath fading and other channel issues. Nevertheless, the results can be extended to include other factors that may impact on the network performance. This is an important point that shall be explored in future works.

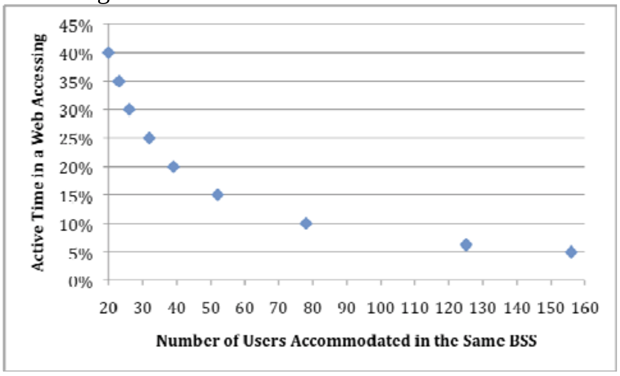

Fig. 10 The number of users accommodated in the same BSS as a function of its active time accessing Web contents.

Acknowledgments

This work was partially supported by the Fundação de Apoio a Pesquisa do Distrito Federal (FAP-DF) – DF and Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

References

[1] A. Arjona, C. Westphal, J. Manner, A. Ylä-Jääski and S. Takala, “Can The Current Generation of Wireless Mesh Networks Compete With Cellular Voice?,” *Computer Communication*, vol. 31, pp. 1564-1578, May 2008.

[2] “Average Web Page Size Triples Since 2003,” http://www.websiteoptimization.com/speed/tweak/average-web-page/, July 10, 2010.

[3] G. Bianchi, “Performance Analysis of the IEEE 802.11 Distributed Coordination Function,” *IEEE Journal on Selected Areas in Communications*, vol. 18, pp. 535–547, March 2000.

[4] R. Bruno, M. Conti, E. Gregori, “Throughput Analysis and Measurements in IEEE 802.11 WLANs with TCP and UDP Traffic Flows,” *IEEE Transactions on Mobile Computing*, vol. 7, pp. 171, February 2008.

[5] “Cetic Painel IBOPE // NetRatings,” http://www.cetic.br/usuarios/ibope/index.htm, July 10, 2010.

[6] F. Filali. “Link-layer Fragmentation and Retransmission Impact on TCP Performance in 802.11-based Networks,” in *Proc. of 7th IFIP/IEEE International Conference on Mobile and Wireless Communications Networks*, September 19-21, 2005.

[7] “Global Index Chart Web usage Data,” http://en-us.nielsen.com/content/nielsen/en_us/product_families/nielsen_netratings.html, July 10, 2010.

[8] P. Gupta, P. Kumar, “The capacity of wireless networks,” *IEEE Transactions on Information Theory*, vol. 46, pp. 338–404, March 2000.

[9] IEEE Std. 802.11-1999, “Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, Reference number ISO/IEC 8802-11:1999(E),” IEEE Std 802.11, 1999 edition IEEE (1999).

[10] J. Jun, P. Peddabachagari, M. Sichitiu, “Theoretical Maximum Throughput of IEEE 802.11 and its Applications,” in Proc of the Second IEEE International Symposium on Network Computing and Applications, April 16-18, pp. 249, 2003.

[11] A. Kumar, E. Altman, D. Miorandi, M. Goyal, “New Insights From a Fixed-Point Analysis of Single Cell IEEE 802.11 WLANS,” *IEEE/ACM Transactions on Networking*, vol. 15, pp. 588–601, June 2007.

[12] “NLANR - Measurement and Network Analysis,” http://pma.nlanr.net/Datacube/, July 10, 2010.

[13] T. Sakurai, H.L. Vu, “Collision Probability in Saturated IEEE 802.11 Networks,” in *Proc. of Australian Telecommunication Networks and Applications Conference*, pp. 21–25, December, 2006.

[14] Y.C. Tay, K.C. Chua, “A capacity analysis for the IEEE 802.11 MAC Protocol,” *Wireless Networks*, vol. 7, pp. 159–171, March 2001.

[15] “The Psychology of Web Performance,” http://www.websiteoptimization.com/speed/tweak/psychology-web-performance/, July 10, 2010.

[16] J. Yeo, A. Agrawala, “Capacity and Variability of the IEEE 802.11 MAC Protocol,” University of Maryland, Department of Computer Science, Technical Report UMIACS-TR-2003-45, March 31, 2003.

Marcos Fagundes Caetano received the B.Sc. degree in Computer Science from the Federal University of Santa Catarina in 2005 and M.Sc. degree in Computer Science from the University of Brasilia in 2008, respectively. Currently he is a Ph.D. student at the Department of Electrical Engineering at the University of Brasilia. His interest includes mobile computing, collaborative computing, distributed systems, opportunistic spectrum allocation, MANETs and routing protocols.

Paulo Vieira received the B.S. degree in Computer Science from University of Brasilia in 2008. Currently hi is working in a industry as a java Developer.

Jacir Luiz Bordim received BSc in Computer Science from University of Passo Fundo (1994) , MSc degree in Computer Science from Nagoya Institute of Technology (2000) and Ph.D. degree in Information Science by Japan Advanced Institute Of Science And Technology (2003). He worked as a researcher at ATR-Japan from 2003 to 2005. Since 2005 he is an Assistant Professor with the Department of Computer Science at University of Brasilia. Dr. Bordim has published extensively in many international conferences and journals, with over 50 peer-reviews papers published. His interest includes mobile computing, collaborative computing, trust computing, distributed systems, opportunistic spectrum allocation, MAC and routing protocols.

Priscila Solis Barreto graduated in Computer Engineering at the Universidade Francisco Marroquin (1991), has a master in Electrical Engineering and Computer Science at the Universidade Federal de Goiás (2000) and has a PhD in Electrical Engineering at the Universidade de Brasília (2007). Currently she is professor at the Department of Computer Science, University of Brasília.