УДК 004.7(045)

Mousa Taleb, MSc Student, Damascus University, Syria
Mhd. Aiman Al Akkad, Dr. Eng., Damascus University, Syria

OPTIMIZATION OF AVAILABLE BANDWIDTH EXPLOITATION IN COMPUTER NETWORKS

The importance of available bandwidth as a parameter used in optimizing computer networks performance is revealed. The parameter estimating and its optimal exploiting enable to overcome congestion and distribute the load in a balanced way. First the available bandwidth estimation methods and its results evaluation concerning precision and speed are introduced. Then the authors' point of view in finding a tool for estimating available bandwidth which has been called PathFinder with hybrid characteristics of previous methods is presented. Then we implemented and evaluated this tool, and the results reached prove its quality concerning speed and precision relaying on a simple and quick algorithm. Finally we emphasized our methods advantages and the importance of using it in data exchange applications through computer networks especially in the Internet.

Key words: available bandwidth, bandwidth estimation, performance evaluation, congestion, load balancing.

Introduction

Recently, attention was greatly drawn to *available bandwidth* estimation within a certain computer network path, as a result of its use in many computer networks applications, as an essential component in network management tools and platforms, which monitors large network systems, which provide us with information about the current usage of network resources. It can also be used to monitor and verify the level of service for users and service providers with accurate information to manage the agreements with each other. In definition *capacity* is the maximum rate at which packets can be transmitted by a link, and *available bandwidth* is a link's unused capacity. As available bandwidth has a significant impact on the performance of many applications that run over computer networks, therefore many researchers studied the possibility of measuring it, and disseminated tools for this. Recently many tools appeared for the estimation process and for performance evaluation of these tools in terms of reliability, accuracy and speed. We observed differences in the results presented in the assessment process of the available bandwidth estimation tools, as in [1] and [2], where the authors agreed that the Pathload tool [3] is accurate, while disagreement was about the accuracy of PathChirp [4]. Also, the author in [5] say that the Spruce tool is more accurate than Pathload, while according to the researcher in [2] Pathload is more accurate than Spruce. When we look at the path of a specific network, one of the basic things we can ask about is the amount of data that can be moved through it between the sender and the receiver without suffering a loss at any part of it, or to be subject to delay which can affect the quality of service. If we were network administrators, or have authority to enter and get some information about the status of some of its links, this is good, but mostly this is not available to us. Therefore, we turn to tools that enable us to know and measure what we want (available bandwidth). Of these tools *Spruce* [5], *PathChirp* [4], *DietTOPP* [6], *Pathload* [3]. All the presented tools so far rely on one of two principles, either use the principle of sending fixed size packets with variable intervals, or sending variable size packets with fixed time intervals. Studies have shown that the tools that use the probing train gives an accuracy better than those that use a pair of probing packets but it requires a longer probing time [7]. To this date no body provided a tool which acquired researchers agreement, i.e. a reliable tool for estimating the available bandwidth and of good accuracy and acceptable speed, and transparent to the network, serving many of the applications that may need a prior knowledge of available bandwidth values.

Available bandwidth estimation

All the presented tools so far rely on one of two principles, either using fixed size packets with variable time interval, or variable size packets with fixed time interval:

– Variable size packets

A single probing train consists of N packets differ in size in a linear manner as shown in figure 1, i. e. if the packet i has size P_i, the size of $i + 1$ packet is $P_{i+1} = P_i + \Delta P$. The size of first sent packet is P_1.
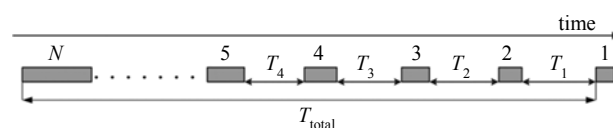

Fig. 1. Probing train of N packets, with varying packets size and time interval

So the linear increase in the size of packets, enables the receiver to determine which packets have been received and which one was lost on the way. So the receiver can locate any packet in the probing train when knowing the size of that packet. The relationship between the packets *order* and *size* within the probing train is shown in figure 2.

– Variable time interval

Here the time interval between the probing train packets is also not fixed, but decreases linearly. In other words, if the time interval at packet i is T_i, and this inter-

val at the following packet $i + 1$ is $T_{i+1} = T_i - \Delta T$ according the following:

$$T_1 > T_2 > T_3 > \ldots > T_{N-2} > T_{N-1}. \qquad (1)$$

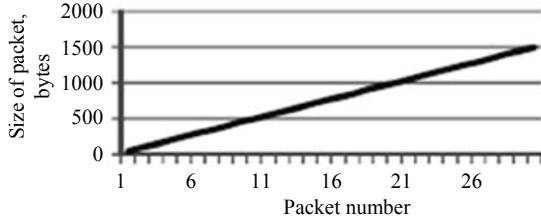Figure 3 explains the relationship of time interval between probing packets and the order of these packets.

Fig. 2. Packet size change for probing train of 30 packets, $\Delta P = 50$ bytes.
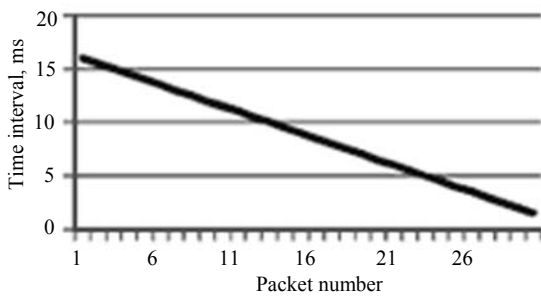

Fig. 3. Relationship between probing train of 30 packets and time intervals, $\Delta T = 0,5$ ms

The receiver should have a prior knowledge about the time intervals between the probing train packets when they were sent, and should register arrival time of each packet. The receiver can locate any received packet through its size, so the duty of the receiver is to compare time intervals between probing packets when sending them (already available), and when received. While sending the probing packets at a rate equal or less than the available bandwidth value $R_{send} \le Abw$, the time intervals between consecutive received probing packets are equal to those between consecutive sent probing packets, if we neglect the delay experienced by probing packets while passing through the network (delay suffered by all packets and it is the time they need to cross the network path), i. e.:

$$T_{1,snd} = T_{1,rcv}; \; T_{2,snd} = T_{2,rcv}; \; \ldots; \; T_{N-1,snd} = T_{N-1,rcv}; \; (2)$$

When the probing packets *transmission rate* exceeds available bandwidth, the relationship (2) becomes not correct and the probing packets have to stand in the waiting queue, leading to an increase in time intervals between the consecutive probing packets when received. Based on the above, the receiver determines the available bandwidth at the packet, among consecutive probing packets, where the time interval starts to increase in comparison with what it had from previous values. The above can be summarized by the following equations:

$$T_{i,snd} = T_{i,rcv}; \; R_i \le ABw;$$
$$T_{i,snd} < T_{i,rcv}; \; R_i > ABw; \qquad (3)$$

We can calculate the instantaneous transmission rate at packet i as follows: $R_i = \dfrac{P_i}{T_i}$

While the instantaneous probing rate of the next packet is:

$$R_{i+1} = \frac{P_{i+1}}{T_{i-1}} = \frac{P_i + \Delta P}{T_i - \Delta T}.$$

At package i the instantaneous probing rate, relative to primitive values, is as follows:

$$R_i = \frac{P_1 + (i+1)\Delta P}{T_1 - (i-1)\Delta T}. \qquad (4)$$

We note that the instantaneous probing rate is a function of packet order in the probing train. Where the values of each of the following parameters: P_1, T_1, ΔP and ΔT are known and specified in advance. Figure 4 shows the change in the probing rate.
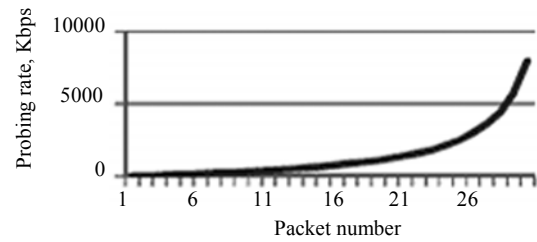

Fig. 4. Relationship between instantaneous probing rate and probing packets order, $N = 30$ packets, $\Delta T = 0.5$ ms, $\Delta P = 50$ bytes

From the above it is clear that the instantaneous probing rate changes with time within what we call the possible or *available probing range*. Available probing range is defined by two limit values: the lower probing rate R_{min}, and the higher probing rate R_{max}, which can be calculated by the following equations:

$$R_{min} = \frac{P_1}{T_1}; \; R_{max} = \frac{P_N}{T_N} = \frac{P_1 + (N-1)\Delta P}{T_1 - (N-1)\Delta T}, \qquad (5)$$

where N is the number of packets forming the probing train.

The length of the probing train is calculated by combining the time intervals between all the packets as follows:

$$T_{total} = \sum_{i=1}^{N} T_i = T_1 + (T_1 - \Delta T) + (T_1 - 2\Delta T) + \ldots$$
$$+ (T_1 - (N-1)\Delta T). \qquad (6)$$

Designing the PathFind tool

Combining accuracy and speed in the estimation process is an essential requirement for approving

the tool. For this, we tried to look for a method that can combine between both requirements accuracy and speed, with the fulfillment of the following constraints:

a. *Excluding synchronization*: between sender and receiver, that is, all the information needed by the receiver can be found in received probing packets, without the need to reconnect with the sender.

b. *Eliminating repetition*: since it has a bad effect on the estimation process continuation time, and the tool should have an acceptable transparency for the network.

c. *Accuracy of estimation*: accuracy is relative, and we think that accuracy of the tool should be high at low available bandwidth values rather than at high values (problems of packets delay and loss increase when available bandwidth is small). We will see later that PathFind gives importance for low values of available bandwidth, which distinguishes it from other tools.

We will rely in the design of our tool on the combination of the two principles mentioned in section 2, in which we change both of probing packets size and time intervals between these packets. The tool uses a single probing train (i.e., repetition canceling), whose length is defined based on the desired precision. The *PathFind* tool using UDP protocol sends a single probing train consisting of N packets differ in their size from each other linearly, and time intervals between the probing train packets is not constant but decreases linearly as well. All what we have to do is determining the appropriate values of $R_{\min}$ and $R_{\max}$ for the required estimation process. Initially, identifying the values of $R_{\min}$ and $R_{\max}$ formed a challenge and was time consuming, until we reached the idea which answers the following question: Who will benefit from this tool?

It is difficult to define a measurable range, a tool can measure, which satisfies all tastes. Since the majority of data exchange currently between users is made via the *internet*, so we will focus on the available bandwidth values compatible with the internet. Actually in most countries the bandwidth value doesn't exceed 7.2 Mbps (e. g. provided in Syria by the service provider "*Syriatel*"). Therefore, we assume that the tool *PathFind* can estimate the available bandwidth within [0, 8] Mbps, with the focus on low values of the available bandwidth. As we mentioned the size of packets in a probing train built by *PathFind*, increases in a linear

way. The largest packet size is 1500 Bytes (taking into account UDP protocol overhead added to the original packet size), while the smallest packet size being sent is determined based on the number of sent packets N, and the amount of size change ΔP. Number of packets used in the probing train, determines the number of levels of available bandwidth which can be estimated, and so this determines the level of required precision. If the number of packets is $N = 30$ packets (additional packet is sent after a specified time interval when sending probe packets stops, in order to obtain the last value which can be estimated), and packet size difference between two consequent packets is $\Delta P = 50$ bytes, the values of packets sizes which form the probing train are the following: 50, 100, 150, …, 1500 bytes, i. e. $P_1 = 50$ bytes. In order to determine the time interval between successive packets, as we said earlier, the time interval between the probing packets decreases linearly with the successive sending of packets. If the time interval of the first packet is $T_1 = 16$ ms, and the time interval of the last packet is $T_{N-1} = 1.5$ ms, taking into account that the number of packets *is* $N = 30$ and the amount of decrease in time is $\Delta T = 0.5$ ms. From all what preceded and basing on equations (1, 2, 3, 4), we get the following parameters:

$P_1 = 50$ bytes, $T_1 = 16$ ms, $\Delta P = 50$ bytes, $\Delta T = 0.5$ ms. Substituting in equation (5) we calculate the minimum and maximum transmission rates:

$$R_{\min} = \frac{P_1}{T_1}; = \frac{50 \cdot 8}{16 \text{ ms}} = 25 \text{ Kbps};$$

$$R_{\max} = \frac{P_N}{T_{N-1}} = \frac{1500 \cdot 8}{1.5 \text{ ms}} = 8 \text{ Mbps}.$$

Substituting in equation (6) we get: $T_{total} = 262.5$ ms.

Practical Experiments and results:

We used *C#* in achieving the send and receive algorithms, and relied on *Wireshark* software to get precisely the time intervals defined between probing packets. To carry out the task of available bandwidth estimation, we designed an experimental network within a controllable environment as shown in figure 5, where the use of four computers in order to generate, send and receive each of the cross-traffic and probing packets, as well as two routers and two switches.
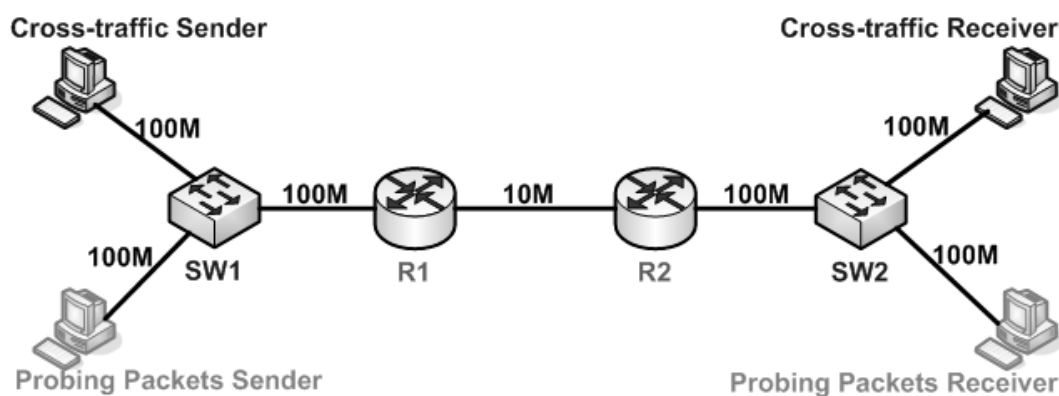

Fig. 5. Test network composed of 2 routers, 2 switches, cross-traffic generator, cross-traffic receiver, probing packets sender, and probing packets receiver

In the process of estimation, we measure the available bandwidth for the link with a capacity of 10 Mbps, located between the two routers. The *cross-traffic* generator generates cross-traffic packets at a steady rate. Estimation process is repeated several times. The objective of repetition is to get accurate values by calculating the average of the values that have been achieved. Values of cross-traffic that have been adopted are: 2 Mbps, 3 Mbps, 4 Mbps, 5 Mbps, 6 Mbps, 7 Mbps, 8 Mbps, 8.5 Mbps, 9 Mbps, 9.5 Mbps. We notice that *PathFind* gives acceptable accuracy at low values of available bandwidth, we can increase the accuracy or change the range of measurement by increasing the number of sent packets and by changing the time interval between those packets.
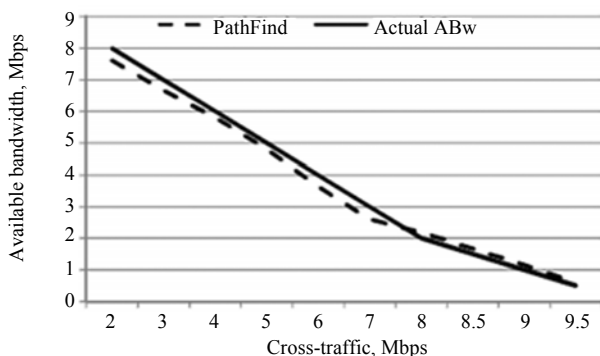

Fig. 6. Available bandwidth estimation results. Continued line is the actual available bandwidth, dashed line is *PathFind* results

Figure 7 shows the relationship between the relative error and the available bandwidth for *PathFind*.

Conclusion

We found that *PathFind* gives good accuracy at low values of available bandwidth, better than at high values of available bandwidth, in comparison with other existing tools for available bandwidth estimation. So, if we try to use the internet, and we have a low available bandwidth, we can use PathFind to optimally determi-

nate the value of available bandwidth that we have to use for sending our data, which offers us the ability to overcome congestion and to achieve load balance over the used network path.
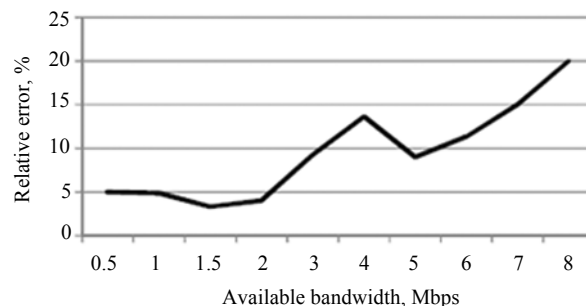

Fig. 7. Relationship between available bandwidth and relative error for *PathFind*.

References

1. Comparison of public end-to-end band-width estimation tools on high-speed links / A. Shriram [et al.] // PAM. – 2005.

2. *Goldoni E., Schivi M.* End-to-End Available Bandwidth Estimation, Tools, an Experimental Comparison. – Pavia : University of Pavia, 2010.

3. *Jain M., Dovrolis C.* Pathload: A measurement tool for end-to-end available band-width // Proceedings of the 3th International workshop on Passive and Active network Measurement PAM. – 2002.

4. PathChirp: Efficient Available Bandwidth Estimation for Network Paths / V. Ribeiro [et al.] // Proceedings of the 4th International workshop on Passive and Active network Measurement PAM. – 2003.

5 *Strauss J., Katabi D., Kaashoek F.* A measurement study of available bandwidth estimation tools. – ACM IMC, 2003. – P. 39–44.

6. *Johnsson A., Melander B., Bjorkman M.* DietTopp: A First Implementation and Evaluation of a Simplified Bandwidth Measurement Method // Proceedings of the 2nd Swedish National Computer Networking Workshop. – 2004.

7. *Jain M., Dovrolis C.* Ten fallacies and pitfalls on end-to-end available bandwidth estimation. – ACM IMC, 2004. – P. 272–277.

Муса Талеб, магистрант, Дамасский университет, Сирия
Мхд. Айман Аль Аккад, кандидат технических наук, Дамасский университет, Сирия

Оптимизация использования доступной полосы пропускания в компьютерных сетях

Показана важность доступной полосы пропускания как параметра, используемого для оптимизации характеристик компьютерных сетей. Его оценка и оптимальное использование позволяют преодолеть перегрузку и сбалансировать нагрузку в сети. Сначала описываются методы оценки доступной полосы пропускания и результаты их применения с точки зрения точности и скорости. Далее представлена точка зрения авторов на инструмент для оценки доступной полосы пропускания, который мы назвали PathFinder и который сочетает в себе характеристики предыдущих методов. Авторы разработали и провели оценку этого инструмента. Достигнутые результаты доказали его качество с точки зрения скорости и точности, которые обеспечиваются простым и быстрым алгоритмом. В заключение перечислены преимущества данного метода и важность его использования в приложениях, связанных с обменом данными в компьютерных сетях, особенно в Интернете.

Ключевые слова: доступная полоса пропускания, оценка полосы пропускания, оценка технических характеристик, перегрузка, выравнивание нагрузки.