Int. J. Communications, Network and System Sciences, 2009, 2, 746-753
doi:10.4236/ijcns.2009.28086 Published Online November 2009 (http://www.SciRP.org/journal/ijcns/).

A Real-Time Measurement Algorithm for Available Bandwidth

Yi YIN, Weidong WU
School of Computer Science and Technology, Wuhan University of Science and Technology, Wuhan, China
*Email: yinyi*1023*@sina.com, wwdtylwt@*163*.com*

Received March 26, 2009; *revised July* 10, 2009; *accepted August* 21, 2009

Abstract

Available bandwidth estimation is useful for route selection in overlay networks, QoS, and traffic engineering. Many measurement algorithms, such as Pathload, Pathchar, and Packet Transmission Rate (PTR) method, etc. have been proposed. PTR method sends a sequence of packet trains to characterize the interaction between probing packets and the competing traffic, and uses the average rate of the packet train as an estimate of the available bandwidth. However, this PTR algorithm does not fully consider the situation that the detection packets lost themselves. This paper improves the original PTR algorithm which considers the specialty of the burst of the network background flow. The improved PTR algorithm uses the method to match the initial gap value and gap step value to solve the problem about the burst of background flow, and the improved PTR algorithm record and control the number of packets with source and destination to solve the lost of some packets. Finally, theory and experiments, verified by the improved algorithm of PTR, can reflect the changes of the network stably and timely under the circumstance of the network fluctuates frequently. It improves the accuracy of a network measurement and makes the measurement results, which can reflect the changes of the network more clearly.

Keywords: Available Network, Background Flow, Detection Packet Pair

1. Introduction

Network measurement is the network application which analyzes the available bandwidth with some kind of technique for Internet. It acquires the availability of bandwidth through the analysis of the data and then obtains the state of the network [1]. Accuracy, precision and timeliness with network measurement will directly impact on network routing, quality of service, network load, volatility, and other issues. Therefore, choosing a good network measurement algorithm can not only measure the result more accurately, but also reflect more changes of the available network bandwidth.

With network measurement, the probe rate model (PRM) is based on the concept of the self-induced congestion [2]. The principle of PRM is to send a series of detection flow which has differently sent rate R_0 from the source to the destination host. The destination detects the rate of flow R_m. The PRM uses the data between R_m and R_0 with available bandwidth to measure the network. This measurement is based on the concept of available bandwidth by lead congestion. Tentative available bandwidth as A, the measurement method of the PRM principle is:

1) if $R_0 < A$, $R_m = R_0$

2) if $R_0 > A$, $R_m < R_0$

where the value of R_0 grows from small to large.

When the rate of destination size changes between R_m and R_0 by detection flow, it means the available bandwidth has been depleted, and now R_0 is the available bandwidth.

The PTR (packet transmission rate) algorithm is based on the PRM principle [3,4].

This paper attempts to consider the burst of background flow and the instability of the measurement packets to improve the PTR algorithm effectively, and uses the network measurement tool to validate that the improved PTR algorithm has smaller volatility and higher measurement accuracy than the original algorithm, and the measurement results are smooth and close to the theoretical value.

2. PTR Algorithm

PTR algorithm is to measure available bandwidth of network based on PRM principle. Its function is that detection packets send it out from the source to the destination with time interval from small to large. When the packets are received at the destination, the packet rate is calculated and PTR algorithm is used to analyze the background flow and measure the bottleneck link, and then regard it as the available bandwidth to measure the condition of network [4]. Here, the available bandwidth means the fact that the end-to-end path of the n links' minimum value in some time respective to pass by the maximum amount of the useful data [5,6]; bottleneck link means the fact that the end-to-end path of the n links' minimum value in some time respective passing by the non-maximum amount of the useful data [7]; and the background flow means the fact that the end-to-end path of the n links' value in some time respective to pass by the non-useful data.

Discussed PTR algorithm in the bottleneck link, assumption the background flow is transmitted in adjacent links. In order to detect the bandwidth available, the source sends more detection packets continuously and the destination will measure the time interval of these packets after the packets thread the router. Considering that the source sends n detection packets continuously, two back-to-back packets p_i and p_{i+1} $(1 \le i \le n-1)$ through the router from the end of the queue arrives at the other end. As shown in Figure 1.

In Figure 1, there are two detection packets and the background flow through the router from the end of the queue arriving at the other end and then recombining. After these time interval of detection packets thread the router and recombine, the background flow and router include the time interval is different before the packet thread the router [8].

The PTR algorithm needs theoretical analysis from some variable, and has been received by the detection exploration.

Assumption g_I (the initial gap) is the detection packet pair [9] which has the initial time interval at the source; g_B (the bottleneck gap) is the detection packet pair time length on the output link; g_O (the output gap) is the detection packet pair which has the time interval at the destination; g_i is the detection packet pair i which has the time interval at the destination in the background flow; $\sum_{i=1}^{M} g_i^+$ is the time interval of a group had increased with packets pair M in packets n; $\sum_{i=1}^{K} g_i^=$ is the time interval of a group had equaled with packets pair K in packets n; $\sum_{i=1}^{N} g_i^-$ is the time interval of a group had decreased with packets pair N in packets n; B_O is the total bandwidth of the link; B_C is the background flow of the link; s is the detection packet size.

By describing PTR algorithm, the situation will make the following general regulation: detection packets pair has queued and through router, and detection packets has not existed deadlock in transmission network.

It is assumed that the background flow is stable at the network. Background flow occupies a little network bandwidth and has a little fluctuation; and the detection packets have not been lost in the process of transmission. Now as premise to analysis the algorithm [10].

When the background flow occupies the bandwidth constantly, according to the necessary measurement data and known data at PRM principles can be obtained by the packet rate which is equivalent to the ratio that the length of packet and the time of packet arrived destination in the background flow at one time. That is:

$$R_m = \frac{s}{g_O - g_B + g_I} \tag{1}$$

Here g_O means the time interval with the head of packets arrive at the destination and then all of the packets have been passed.

And now,

$$g_O = g_B + \frac{B_C}{B_O} \cdot g_I \tag{2}$$

Here $\frac{B_C}{B_O} \cdot g_I$ means the latency that the bandwidth has been occupied by the background flow that make the detection packets have not arrived at the destination on time.

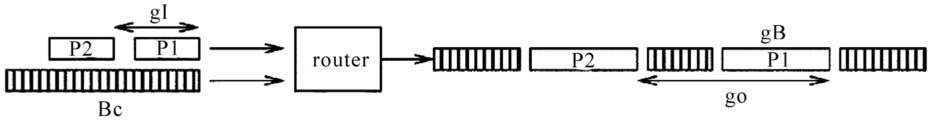

Figure 1. Detection packets and background flow competition through route.

However, the background flow based on a constant of bandwidth is only a basic situation that Equation (1) cannot be used for the actual network. In general, background flow of bandwidth is always changing. The variable bandwidth occupied by the background flow and the rate of the detection packets arrival terminal is:

$$R_m = \frac{(M+K+N)\cdot s}{\sum_{i=1}^{M} g_i^+ + \sum_{i=1}^{K} g_i^= + \sum_{i=1}^{N} g_i^-} \quad (3)$$

Equation (3) as PTR equation, and PTR algorithm is used by this equation.

The significance of this equation is the ratio that sending a total length of several detection packets and receiving all time of the detection packet at the destination of the link. According to the PRM principle, the source rate will gain the network available bandwidth information timely. The background flow, which has occupied the bandwidth variably in the network, is the application of measurement in the real situation.

Evidently, PTR algorithm is built on the PRM principle, which can be used by the changes of the background flow and detected the packet rate.

3. Improved Algorithm

In the actual network measurement, either volatility or loss detection packet has not been avoided. But the PTR algorithm in measurement has not considered these two issues. Therefore, it must remove these assumptions and discuss verification in the actual network.

The original PTR algorithm, based on the PRM model, has higher accuracy and faster speed of convergence with the background flow relatively constant and the utilization of the link is not high and the influence is small for networks. As mentioned earlier that this is the two assumptions of a description. When the flow is small or stable with the network background flow traffic, detection packet by the volatility of the time interval can be more objective response network conditions. However, when the network environment is poor, like the background flow changes on the network path, it is larger or the utilization of the bottleneck link is higher, as background flow volatility is more obviously, the existing PTR algorithm sends detection packets with the time interval from small to large, which have some false measurement value, and then the measurement results significant ups and downs, which means the measurement is not stable. As the detection packet is instable, it cannot be obtained with accurate value when the network condition has large fluctuation.

3.1. Theoretical Exploration with the Improved Algorithm

Through Equation (3), the detection packet rate is the ratio by two summations. Since the objective over PTR algorithm reflects the changes in the bottleneck rather than accurately to measure the timely rate with each detection packet, it can be arranged the PTR algorithm further.

Assumption n detection packets will be divided into l groups once a time, each group has k detection values. Firstly, calculate average send gap ($avg\_sed\_gap(i)$) and average receive gap ($avg\_rec\_gap(i)$) with the packet sequence i , ($i \in (0,k)$). Secondly, calculate the summation of average send gap and average receive gap with the group of packet sequence l . The significance by Equation (3) could be re-described

$$R_m = \frac{s \times l}{\sum_{i=1}^{l} avg\_rec\_gap(i)} \quad (4)$$

Apart from PTR algorithm over two data with g_i and s , that needs to introduce two new values.

From the Equation (4), other than the interval time g_i and the size s from the known detection packet with the background flow at the source in Equation (3), it introduces two new values, that are the average send gap($avg\_sed\_gap(i)$) and the average receive gap($avg\_rec\_gap(i)$).These two values mean the weighted average with the time interval which is the source and destination from the detection packets. As the PTR algorithm does not measure the timely rate accurately by detection packets, it only needs to measure the changes of the network truly. It makes every time interval values not accurate reflected, thus it just sum those average value which could direct to reflect some changes.

This methods estimate detection rates used to sample mean can be reduced the consideration of the detection packets that is not necessary to consider a small quantity of the detection packets lost.

3.2. Improvement of the Algorithm in the Circumstance of Frequent Fluctuations in the Background Flow

Reference Equation (4) can improve the PTR algorithm appropriately. Assumptions the detection packets have been sent to the initial time interval *init_gap* and transmitted to the time interval *gap_step*. *init_gap* and *gap_step* will be set the fixed value but not as described in the PTR algorithm from small to large to send the detection packet with time interval [11].

Write algorithm for these conditions.

```
Algorithm PTR:
{
  probe_num=PROBENUM;
    packet_size=PACKETSIZE;
    gB=GET_GB();
    init_gap=gB/2;
    gap_step=gB/8;
    dst_gap_sum=0;

  for(packet_size=PACKETSIZE;packet_size!=0;packet
_size--)
     {
  SEND_PROBING_PACKETS(probe_num;packet_size);
        inc_gap_sum=GET_INCREASED_GAPS();
        dst_gap_sum=GET_DST_GAPS();
      }
    c_bw=gB*inc_gap_sum/dst_gap_sum;
    a_bw=b_bw-c_bw;
  }
```

There are two fixed value in algorithm, that is, init_gap=gB/2, gap_step=gB/8. These two values are constant. When the gap values are constant, no matter how changes in the network background flow, the bandwidth is always constant over the detection packet. Therefore, it is easy to be consistent with measurements of the fluctuations and change of the background flow.

3.3. Improvement of the Algorithm in the Circumstance of Desert with the Detection Packets

If the original PTR algorithm encounters the high-intensity in network background flow, as the total bandwidth is always limited, even if the gap value is constant and exploration data are nice match for the background flow, detection packets may not arrival at the destination but lost, thereby affect the analysis of the data. Analysis the PTR equation, by the Equation (4) can be seen, if detection packet is lost, it will affect the convergence conditions of the summation, thus affect the accuracy of the detection data.

Now it describes the flow chart with improved PTR algorithm, and increases the data with recording and controlling in source and destination. At the same time, it takes advantage of an improved approach to matching the transmitter and receiver data, does accurate records, and finally analyzes the network (Figure 2).

Source:

1) Take some group by detection packets from the host, each group have k packets, recorded as P_r, where $r \in [0,k]$, deliver to send cache

2) Saving a constant time t_s in send cache,

3) Deliver detection packet and t_s from send cache, then go 1)

Destination:

1) Waiting, record wait time t_w at the same time

2) Receive detection packet P_r from the source and send it to receive cache, and intercalate a variable R to record the detection packets number from the source, the initial value of R is 0

3) Waiting time t_w and detecting packet will be handed over the host, and endue a new variable t_n from t_w

4) Go (1), then $t_w \leftarrow 0$, $R++$

5) Compare t_n with t_s, and define three value $M=0$, $K=0$, $N=0$. If $t_n > t_s$, $M++$; if $t_n = t_s$, $K++$; if $t_n < t_s$, $N++$

6) When the compare is complete, $t_n \leftarrow 0$

It can be seen from the algorithm description to provide a very important parameter t_w. With the value of t_w, it can be distinguished in the detection packets which have been sent out to the destination, and the source is arrived or lost due to the net reasons. If time is over, the source without retransmission and the destination can also automatically be received and recorded in the number of packets that insure match for the packets at each side.

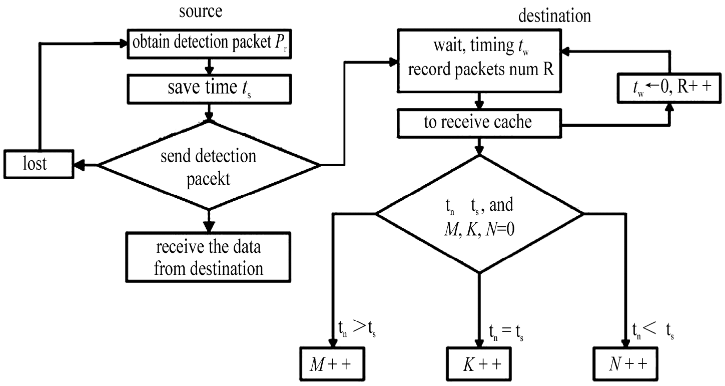

Figure 2. Flow chart with improved PTR algorithm.

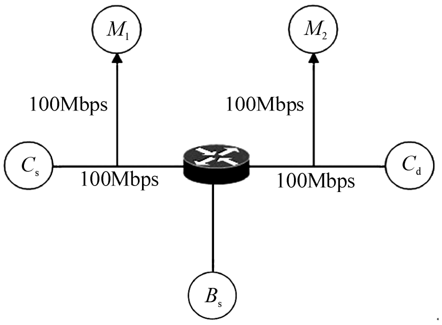

Figure 3. Measurement topology.

Improving these two points by the original PTR algorithm is that PTR algorithm is a method to change the detection packets through its own to measure network. This algorithm requires its own change to directly reflect the network, but if the change of background flow is also obviously, it is impossible to know the variation packets are in congestion state. So the measurement result is deviation. At the same time, once the background flow is too big, may lead the detection packets lost, which is another reason to deviation of the measurement. These deviations are PTR algorithm need to improve.

Improved algorithm, on the one hand, grasps the background flow of the changes is more accurately; on the other hand, the transmission of detection packet hold maximize control. Both of these improvements, the accuracy of measurement have increased greatly, and the changes of the load may too faster.

4. Experimental Results

First, using the improved PTR algorithm to measure network, and calculate the theoretical data and circulate algorithm to compare whether the results match. Second, through network to analysis receive data to prove its precision and timeliness, and compare the original algorithm to know the improved algorithm has higher accuracy.

Measurement topology as is shown in Figure 3. C_s is source and C_d is destination above detection packet. M_1 is monitor the variation in time interval with detection packet by source and M_2 is monitor by destination. Monitor and compare data by two ends of the router when detection packets have been sent. The topology of experiments use the emulator [12], On this basis, compare the original algorithm and the improved algorithm with the theoretical value, and accord statistical properties of the Internet flow [13,14].

According to the Equation (4), $\sum_{i=1}^{l} avg\_rec\_gap(i)$ can be used in the algorithm to express the value of dst_gap_sum, so

$$R_m = \frac{s \times 8 \times l}{dst\_gap\_sum} = \frac{s \times 8 \times n}{\sum g_O},$$

That acquire the output time interval g_O. Compare the measurement curves with the value of $\frac{g_O}{g_I}$ as is shown in Figure 4, it also can reflect the basic values of the network.

For $g_B = 0.08ms$, $g_I = 0.31ms$, $B_C = 7.2Mb/s$, $B_O = 20Mb/s$, the length of the detection packet is 700 Byte, and there are 60 packets in a group. Using the emulator tools to make the background flow intensity increased gradually from $0Mb/s$ to $20Mb/s$. The reason to take these values is whether it is time or rate value, the size of these values are relatively modest which have not been lost with changes in the quality of network easily and have not been obliterated with the

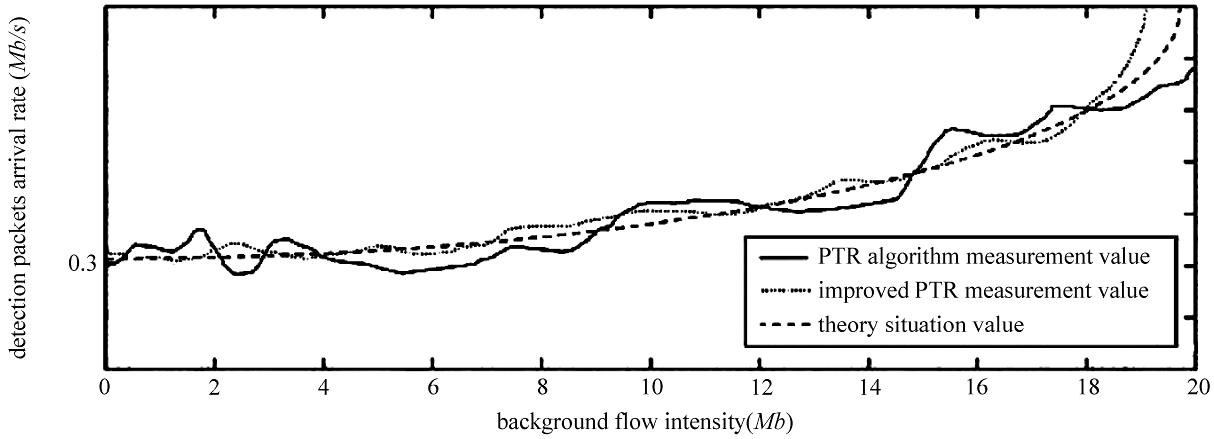

Figure 4. The theoretical value compared with the measured value.

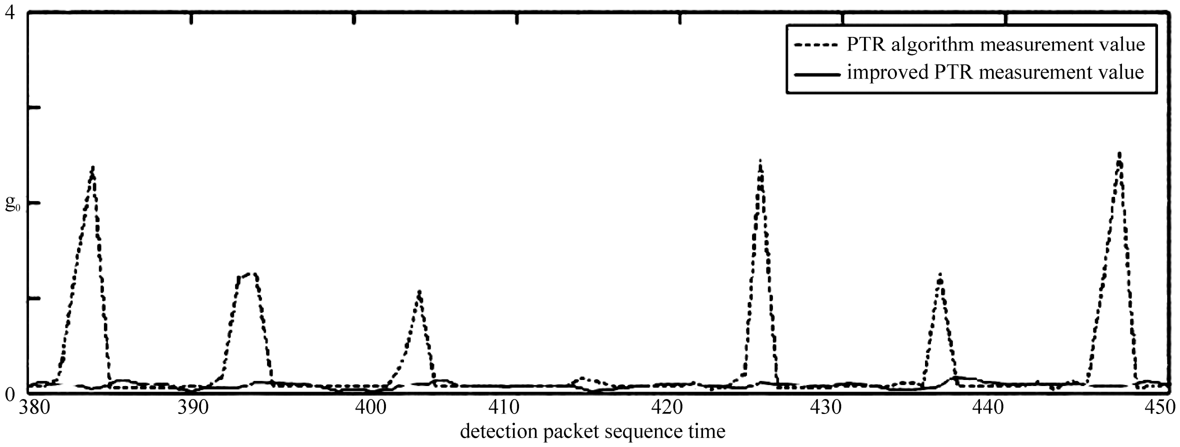

Figure 5. The comparison of the value of receiving which are based on the same source rate between the original algorithm and the improved algorithm.

excessive bandwidth facilely [4,15]. The theory is that before the background flow intensity reached $12.8Mb/s$ ($20Mb/s - 7.2Mb/s$), the rate of the detection packets arrived at the destination will not obviously change which have been sent. So, before the background flow intensity reaches $12.8Mb/s$, the detection packets line in the Figure 4 shows as a curve that the slope grows slowly. When the background flow intensity from $12.8Mb/s$ to $20Mb/s$, detection packet rate will slow down gradually until the background flow can occupied available bandwidth fully and causing bottlenecks that lead to detection packets are inaccessible to destination. At the destination, the detection rate of the available bandwidth is $0Mb/s$. At this point in Figure 4, the rate of the detection packets have been showed the curve that the slope as $+\infty$. The measurements and theoretical values inosculate basically. At the same time, the improved algorithm compared to the original algorithm can be clearly seen, the original algorithm with a strong volatility, and the improved algorithm performance the value have been smoothed. It is just consistent with the situation of the algorithm performance which after setting the two fixed values in Caption 3.2 of the above described.

According to the Equation (2), when the background flow in the path and compete bandwidth with the detection packet flow, the output data stream for the time interval is $g_O = g_B + \frac{B_C \cdot g_I}{B_O}$. This equation is going to be validated with theoretical and experimental, and compare the improved algorithm to the original algorithm, as is shown in Figure 5.

Here, get $B_O = 100Mb/s$. The remaining data with the same on the past experimental data that the theoretical value is:

$$g_O = 0.08 + \frac{7.2 \times 0.31}{100} = 0.10ms$$

In the experimental model, the source send detection packets in the C_s can be measured in the value of g_I in the M_1 as shown in Figure 5 of the initial interval value. When detection packets compete with background flow which occupy the bandwidth $B_C = 7.2Mb/s$ to though the router, the measurement value of the improved PTR algorithm g_O is as shown in Figure 5 with the value “improved algorithm” in M_2, and the measurement the same value of the original PTR algorithm as is shown in Figure 5 with the value “original algorithm”. The experimental results indicate that the improved algorithm of the measurement results are nearly match with the calculation results in Equation (2) , and the data from the improved algorithm are more stable than original algorithm.

The experiment result is proved the usefulness of the detecting packets records as Caption 3.3.

The measurement result, which is gained by the experimental environment, and using the tool of MRTG (Multi Router Traffic Grapher), compares with the improved algorithm and the original algorithm. MRTG is a software tool of monitoring network link flux load. It use the snmp agrement to gain the flow information of the equipment, and displays flux load to users by HTML documents of graphics which included PNG format, displaying the flux load in a very intuitive form. In the interception 24-hour process of measurement, it has entered 40Mb/s background flow for one hour initiatively, but the intensity of other times background flow is only 20Mb/s. The measurement results are shown in Figure 6. From the measurement results, we can see the measurement results of improved algorithm and the stability of MRTG are almost always the same and the improved algorithm data are more stable than the original algorithm data at the same time. Because the loss of detection group is almost inevitable in the network, it has not

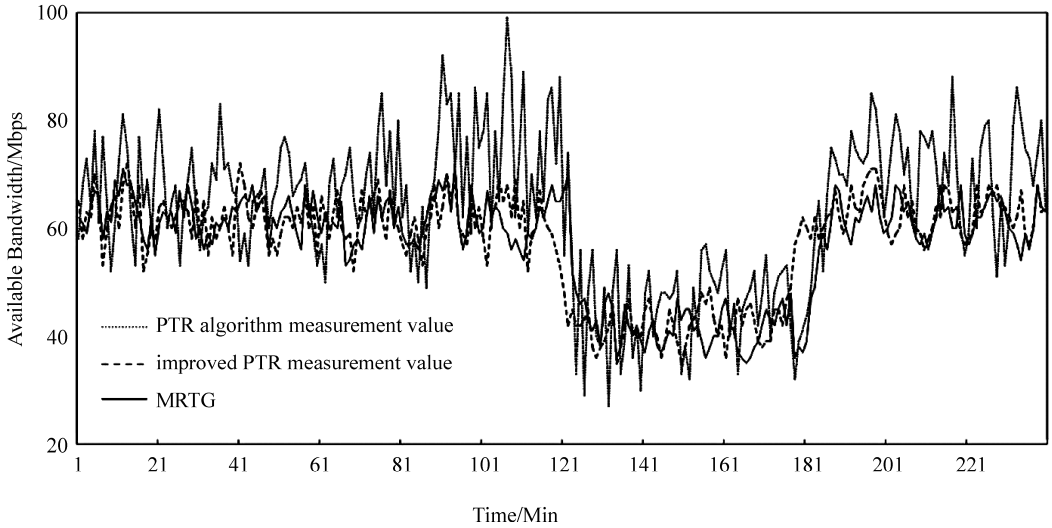

Figure 6. The original algorithm compete the same sending rate with the improved algorithm.

considered the factor of loss of detecting packets in original algorithm. We can obtain the velocity of detecting packets by data overall, the original algorithm would produce some peak value as Figure 6 after detecting group loss. The improved PTR algorithm, by dealing with the lost detection packets, the results of detect is more stable and accord with the measurement value which was gained from MRTG.

When the background flow compete the whole bandwidth with the data flow, the improved PTR algorithm measure the data always accurate.

5. Conclusions

This paper discusses the theory and application of PTR algorithm of network measurement. After the analysis of the limitations and shortcomings of the algorithm, it proposed the improved algorithm and processes, and by comparing the algorithm's theoretical value and the actual measured value to induce conclusions for performance improved algorithm can match with the theoretical value better. At the same time, it also put up that improved algorithm can measure the ins and outs of the network accurately, increasing the veracity of the network measuring and boosting up the precision of the network measuring. Especially when the network speed fluctuates frequently, it has a greater improvement in reflecting of network conditions timely and accurately.

6. References

[1] S. Banerjee and A. Agrawala, "Estimating available capacity of a network connection [A]," IEEE International Conference on Networks [C], Singapore, pp. 131–138, September 2000. "NetDyn: Network Measutrements Tool," http://www.CS.umd.edu/suman/netdyn/.

[2] J. Strauss, D. Katabi, and F. Kaashoek, "A measurement study of available bandwidth estimation tools [A]," Proceedings of ACM Intemet Measurement Conference (IMC) [C], Miami Beach, Florida, October 2003.

[3] N. N. Hu and P. Steenkiste, "Evaluation and characterization of available bandwidth probing techniques [J]," IEEE Journal on Selected Areas in Communications, Vol. 21, No. 6, pp. 879–894, August 2003.

[4] C. Dovrolis, P. Ramanathan, and D. Moore, "What do packet dispersion techniques measure?" in Proceedings of Conference Computer Communication, pp. 905–914, April 2001.

[5] M. Jain and C. Dovrofis, "End-to-end available bandwidth: Measurement methodology, dynamics, and relation with TCP throughput [A]," Proceedings of ACM SIGCOMM Symposium on Communication Architectures Protocols [C], Pittsburgh, PA, USA, pp. 295–308, August 2002.

[6] K. Lai and M. Baker, "Nettimer: A tool for measuring bottleneck link bandwidth," in Proceeding of USENIX Symposium on Internet Technologies and Systems1, pp. 123–134, March 200.

[7] R. Prosad, C. Davrolis, M. Murray, et al., "Bandwidth estimation: Metrics measurement techniques and tools [J]," IEEE Network, Vol. 17, No. 6, pp. 27–35, 2003.

[8] V. Paxson, "Measurements and analysis of end-to-end internet dynamics," Ph. D. dissertation, Computer Science Division, U. C. Berkeley, Berkeley, CA, May 1996.

[9] N. N. Hu and P. Steenkiste, "Estimating available bandwidth using packet pair probing [J]," Carnegie Mellon University (CMU), 9 September 2002.

[10] Pasztor A, Veitch D. The Packet Size Dependence of Packet Pair Like Methods[C]. Proc. of IWQoS' 02, Mi-

ami Beach, Florida, USA, 2002.

[11] Ns2 [Online]. Available: http://www.isi.edu/nsnam/ns.

[12] K. Claffy, G. Miller, and K. Thompson, "The nature of the beast: Recent traffic measurements from an internet backbone," presented at the ISOC INET Conf., July 1998.

[13] Y. Zhang, N. Duffield, V. Paxson, and S. Shenker, "On the constancy of Internet path properties," in Proc. ACM SIGCOMM Internet Measurement Workshop, San Francisco, CA, Nov. 2001, pp. 197–211.

[14] K. Claffy, G. Miller, and K. Thompson, "The nature of the beast: Recent traffic measurements from an internet backbone," presented at the ISOC INET Conf., July 1998.