International Journal of Information Technology and Knowledge Management
January-June 2011, Volume 4, No. 1, pp. 169-172

MEASUREMENT OF BOTTLENECK USING PATHNECK

Gouri M. Patil[1] & Ajay Kumar[2]

Technology scaling has caused Negative Bias Temperature Instability (NBTI) to emerge as a major circuit reliability concern for the circuit designers Simultaneously leakage power is becoming a greater fraction of the total power dissipated by logic circuits. As both NBTI and leakage power are highly dependent on vectors applied at the circuit's inputs, they can be minimized by applying carefully chosen input vectors during periods when the circuit is in standby or idle mode. Unfortunately input vectors that minimize leakage power are not the ones that minimize NBTI degradation, so there is a need for a methodology to generate input vectors that minimize both of these variables. This paper proposes such a systematic methodology for the generation of input vectors which minimize leakage power under the constraint that NBTI degradation does not exceed a specified limit. These input vectors can be applied at the primary inputs of a circuit when it is in standby/ idle mode and are such that the gates dissipate only a small amount of leakage power and also allow a large majority of the transistors on critical paths to be in the "recovery" phase of NBTI degradation.

The advantage of this methodology is that allowing circuit designers to constrain NBTI degradation to below a specified limit enables tighter guardtbanding, increasing performance. Constrain of NBTI can be limited by choosing input vectors probability based algorithms. Our methodology guarantees that the generated input vector dissipates the least leakage power among all the input vectors that satisfy the degradation constraint. We formulate the problem as a zero-one integer linear program and show that this formulation produces input vectors whose leakage power is within 1% of a minimum leakage vector selected by a search algorithm and simultaneously reduces NBTI by about 5.75% of maximum circuit delay as compared to the worst case NBTI degradation.

Keywords: NBTI, Leakage, Input Vector, Critical Path

1. Introduction

Pathneck[1] is based on a Novel probing technique–RPT (Recursive Packet Train)[7]. It allows end users to efficiently and accurately locate bottleneck links on the Internet. The basic idea is to combine measurement packets and the load packets in a single probing packet train.Load packets fires router responses to obtain measurements. The number of load packets is critical for the tool to work correctly; the value depends on the access link capacity and 400MHz CPU, if it is then use the default value (ie.60 packets) otherwise, eg. 10Mbps access link or <400MHz CPU the use the number of packets (20 or less). The 5 number of load packets are used in this online detection processing. The load packet size in byte is 500. The ICMP[6] probing packets (UDP) are set to measure Round-Trip-Time(RTT)[7] to a specific destination. TTL triggers responses from the routers along the network path, thus collects the hostname and RTT of the routers.

2. Methodology

Here the pathneck[1] tool is used to do measurement of Internet bottleneck. 50 probings[3] are used for this measurement[2]. The CPU load of the probing nodes is increased (by using option –l), online detection processing is enabled (by using the option –o) and the DNS lookup is also enabled (by using the option –x).

Probing configuration – -l 5 number of load packets, -c Use ICMP probing packets (UDP).

Output setting – -o Enable online detection processing, -x Enable the DNS lookup

3. Analysis and Results

Probe the path 50 times. The raw gap measurement and gap values on each hope are plotted in Fig. 1 for probes – 18, 23,30,33,36 and 41. We can see that the gap values maximally changes appear at hop 14, whose input link is the bottleneck link. There are also small gap changes at subsequent routers.

In some situations the gap value get increases, decreased or remains same as shown in Fig.1. In Fig.1(a), (b) and (d), gap value increases at hop 14. Which suggest that the lower bandwidth[4] is available on the link 209.85, 241.23, 209.85.242.255, and 209.85.242.255. In Fig.1, all probes gap value decreased. These decreases are probably due to temporary queuing caused by traffic burst ness.

[1]Department of Computer Science, P.O. Nahata College, Bhusawal, India

[2]Director, D.Y. Patil Institute of MCA, Pune-44, India

Email: [1]gourimpatil@gmail.com, [2]ajay19_61@rediffmail.com

For each probe the choke points are ranked out based on their gap value in probing set.

The packet train transmission rate R = ts/g where ts is the total size for all the packets in the train and g is the gap value.

As the gap value increase, the packet train was more stretched out by the link. It suggests that lower bandwidth is available on the corresponding link. And as a result, identify the choke point with the largest gap value as the bottleneck of the path.

Choke points are denoted by ● in Fig.1 (a),(b),(c),(d),(e),(f) for probe- 18,23,30,33,36 and 41 has choke points. That means out of 50 probing only 6 probes has choke points.

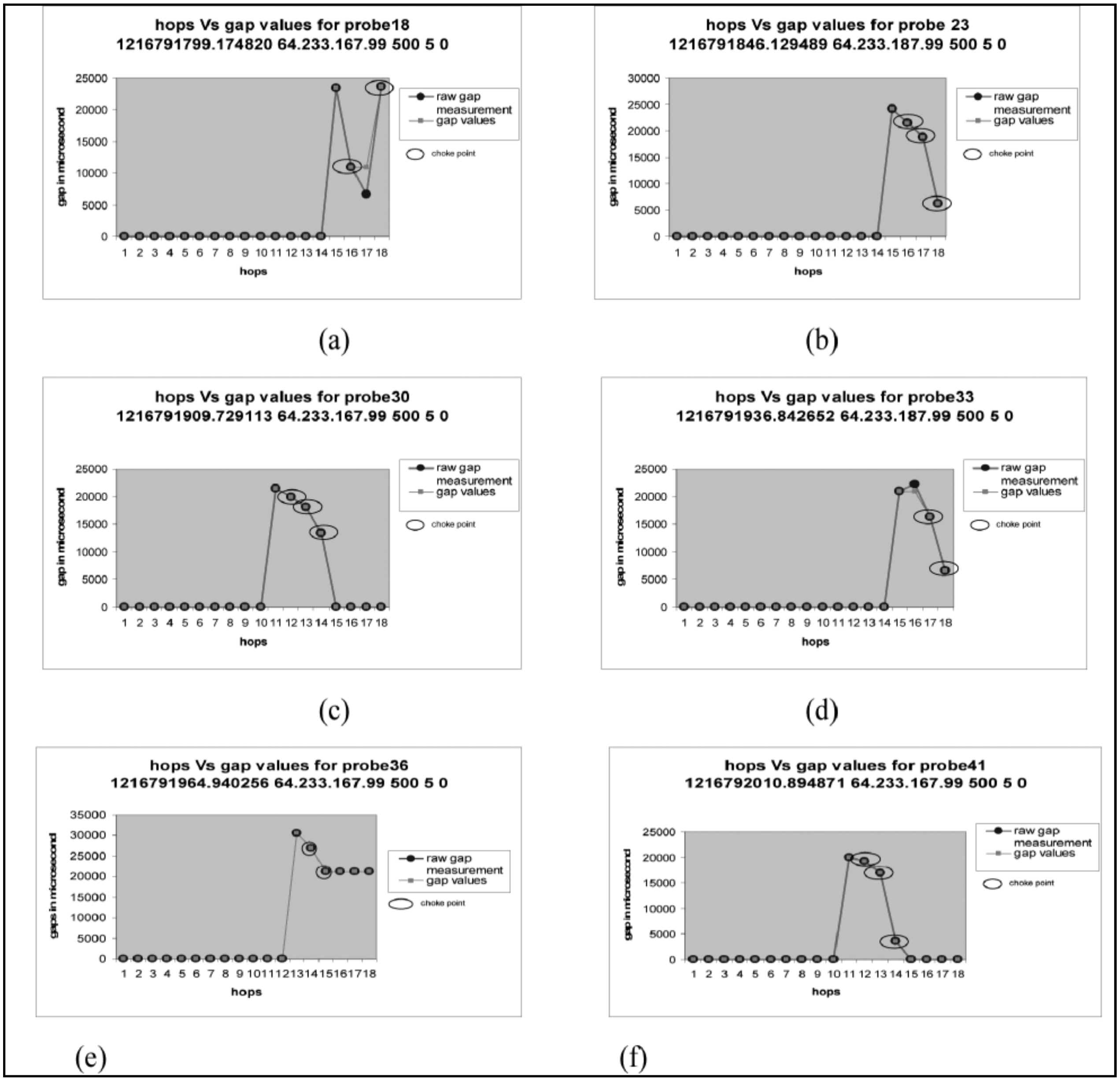

Fig. 1: Hopes Vs Gap Values for Probing Set 18, 23, 30, 33, 36, 41

TABLE-1-I
Candidate Choke Point

Router	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	Total
Conf.>=0.1													2	3	1	2	2	3	13

The TABLE-1-I indicates the number of times each hope being a candidate choke point. There are total 13 choke points are detected. (with conf .>= 0.1) The probing detect the hop 14 and 18 as the bottleneck and the bottleneck links are, 209.85.241.20 and 216.239.49.222 respectively.

There is a clear relationship between bottleneck and link loss and delay. Network traffic congestion[7][8] causes queuing, packet loss and hence bottleneck may exist.

Queuing delay is the metric, which is frequently used to indicate the congestion. Queuing delay is measured by using medianRTT and minimumRTT from the probing source to a router. Fig.2 shows the RTT in microsecond for every hop present on the path from source to destination for probes – 18, 23, 30, 33, 36 and 41.

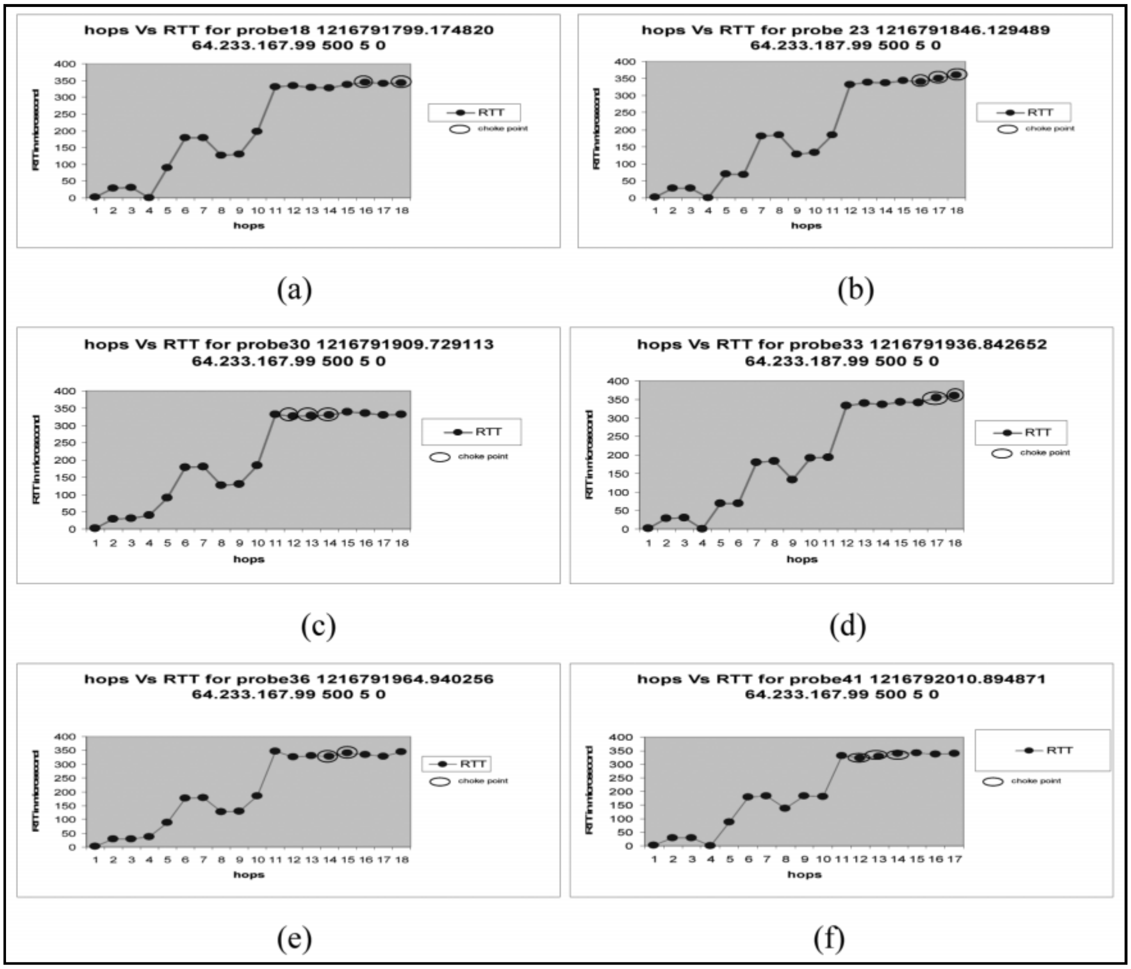

Fig. 2: Hops Vs RTT for Probing Set 18, 23, 30, 33, 36, 41

In these graphs it is observed that the RTT maximally get increases from source to the destination. The RTT for the routers to each probe is generally above 330 microseconds, having the gap values for the hops. Queuing delay between the routers having gap values is from near about 1-10 microseconds. The RTT at choke points for hops is denoted by circle.

4. CONCLUSION

The choke point with the largest gap value is identified as the bottleneck of the path. But as it is not possible to control the packet train structure at each hop, the RTT does not actually measure the available bandwidth on each link. So in some cases, pathneck selects the wrong choke points as the bottleneck.

Change in conf. And d-rate will result in change in the number of choke points. The results are sensitive to the conf. Value and d-rate. Value medianRTT-minRTT will helps to find out bottleneck. z

REFERENCES

[1] N. Hu, L.E.Li, Z.M.Mao, P. Steenkiste, and J. Wang., “Locating Internet Bottlenecks: Algorithms, Measurements, and Implication”, In Proc. ACM SIGCOMM, August 2004.

[2] Ningning Hu, Li(Erran) Li, Zhuoqing Morley Mao, Peter Steenkiste and Jia Wang, “A Measurement Study of Internet Bottleneck.”

[3] N. Hu and P. Steenkiste, "Evaluation and Characterization of Available Bandwidth Probing Techniques", IEEE JSAC Special Issue on Internet and WWW Measurement, Mapping, and Modeling, 21(6), August 2003.

[4] M. Jain and C. Dovrolis, "End-to-end Available Bandwidth: Measurement Methodology, Dynamics and Relation with TCP Throughput", In Proc. ACM SIGCOMM, August 2002.

[5] M. Mathis and J. Mahdavi, "Diagnosing Internet Congestion with a Transport Layer Performance Tool", In Proc. INET "96, Montreal, Canada, June 1996, http://www.isoc.org/inet96/proceedings/.

[6] RFC 792, Internet Control Message Protocol, Sept 1981.

[7] N. Hu and P. Steenkiste, "RPT: A Low Overhead Single-end Probing Tool for Detecting Network Congestion Positions", Dec.20, 2003, CMU-CS-03-218.

[8] M. Isreb and A.I. Khan, "Internet Congestion Modeling and Parallel Distributed Analysis", www.springerlink.com.