Evaluation of IP Transmission Jitter Estimators Using One-Way Active Measurement Protocol (OWAMP)

Przemysław Skurowski[1], Robert Wójcicki[1], and Zbigniew Jerzak[2]

[1] Silesian University of Technology, Department of Computer Science,
ul. Akademicka 16, 44-100 Gliwice, Poland
{przemyslaw.skurowski,robert.wojcicki}@polsl.pl
http://inf.polsl.pl

[2] SAP Research Dresden, SAP AG
zbigniew.jerzak@sap.com
http://www.sap.com

Abstract. Network transmission with real-time constraints appears to be very sensitive to the jitter – short time packet delivery delay. In this paper we describe several methods for estimating that phenomenon. We perform quantitative and qualitative analyses for the gathered data using modern measurement tool called OWAMP (one way ping) and stratum 1 Network Time Protocol (NTP) time server.

Keywords: jitter estimation, network delay, estimator, multimedia transmission, distributed systems.

1 Introduction

1.1 Background

Reliability of the data transmission plays crucial role in many areas of computer networks. Transmission delay and its variability called 'jitter' is one of the most important parameters characterizing networks' transmission. Precise estimation of these parameters might be especially important in areas such as publish/subscribe systems or multimedia transmission (VoIP).

Transmission of multimedia content in its most advanced technological solutions is performed using streaming paradigm. The data is not stored locally but consumed as it comes with a small delay protecting against jitter which might appear as transmission 'hiccup'. In case of non-interactive transmission like Video-On-Demand (VoD) or even live broadcasts the problem is less troublesome due to large de-jittering buffers (up to several seconds) that protect well against jitter. More demanding case is the VoIP (Video or Voice over IP) interactive transmissions. Such one way transmission delay (latency) depending on the target application and a class of service quality should be less than 150 ms for high quality services (class 0) and less than 400 ms for worse cases (class 1) [1]. In

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2010, CCIS 79, pp. 153–162, 2010.

such cases precise estimation of jitter is important: its results should be conservative enough to buffer enough packets to support uninterrupted streaming and on the other side it should be small enough to ensure interactive transmission.

In this study authors decided to verify usability of jitter estimators using OWAMP (*One-Way Active Measurement Protocol*) [2] tool that allows for precise estimation of packets one-way transmission delays. Prior to OWAMP reliable measurement of transmission delays was both technological and organizational problem. Due to the impossibility of precise synchronization of distributed clock it was necessary to provide GPS calibrated clocks to both remote and local machines. There was also a need to prepare special software just for sending packets containing only timestamps. Therefore, ICMP echo (ping) requests were used to gather data for the analysis of jitter estimators. As the influence of transmission delays is proportional to the number of packet hops and due to the fact that ICMP echo is often treated by the routers as second class traffic the ICMP echo approach could not be treated as a real network delay measurement. Nowadays, there exist several publicly available [3] OWAMP PMPs (*Performance Measurement Point*) therefore it is possible to gather real network characteristics.

1.2 Jitter Classification

According to [4] in packet switching networking terms jitter definition is just a simple packet delay variation. Clark [5] identified three basic kinds of jitter: contant, transient and short term delay variation. There also exists one more phenomenon – slow delay change which is not jitter in fact but it is an obstacle that dejitter buffers has to deal with. These types and their root causes are as follows (see Fig. 1):

- constant jitter – present in flawless transmission with roughly constant packet to packet delay variation,
- transient jitter – where single packet is significantly delayed to the others in stream. It is observed in numerous cases and it has numerous reasons like routing table up-dates, LAN congestion, router packet scheduling, route flapping and others,
- short term delay variation – occurring when a burst of packets has increased trans-mission delay. It is usually connected with access link congestion or route change,
- slow delay change – appearing in graph as a ramp like characteristics – Clark also connects this phenomenon with access link congestion.

2 Jitter Estimators

There are various estimators described in documents of two interested institutions IETF and ITU. They can be divided into two main groups PDV (packet delay variation) related to some absolute reference value and IPDV (inter packet delay variation) where as a reference packet there is used preceding one. Additionally there is also used simple moving range statistics or a moving distance between selected percentiles.

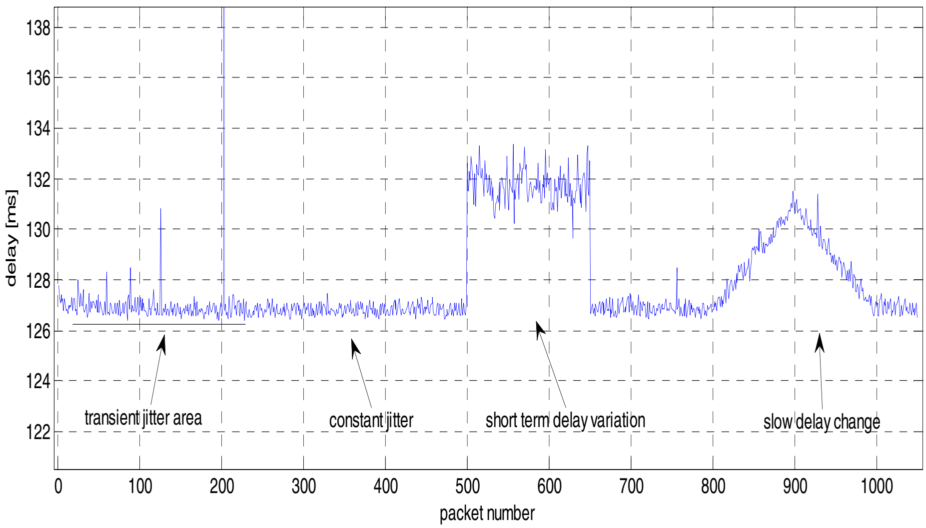

Fig. 1. Types of jitter

2.1 Inter-packet Delay Variation

The basic form of this measure appears in two variants: simple IP Delay Variation, IPDV [6] and Mean Packet to Packet Delay Variation, MPPDV [5]:

$$IPDV(n) = t(n) - t(n-1) \tag{1}$$

$$MPPDV(n) = \text{mean}\left(|t(n) - t(n-1)|\right) \tag{2}$$

where: $t(n)$, $t(n-1)$ is transmission delay of n^{th} and n^{th-1} packet, in case of $MPPDV$ mean value is computed for the most recent 16 packets.

2.2 RTP/RTCP Inter-packet Delay Estimator

In the family of IETF multimedia protocols transmission of audiovisual data is done using a pair of protocols RTP/RTCP [7]. The RTCP protocol reports among the others current jitter of RTP transmission. The calculated jitter is based on exponentially weighed moving average (EWMA) inter packet delay difference and it is computed according to the following formula:

$$J(n) = \frac{1}{16}|D(n)| + \frac{15}{16}J(n-1) \tag{3}$$

where: $J(n)$, $J(n-1)$ is current and previous jitter estimate, $D(n)$ is current packet delay change computed as difference of transmission time of two packets:

$$\begin{aligned} D(n) = t(n) - t(n-1) = (R(n) - S(n)) - (R(n-1) - S(n-1)) = \\ (R(n) - R(n-1)) - (S(n) - S(n-1)) \end{aligned}$$

where: $t(n)$, $t(n-1)$ is transmission delay, $R(n)$, $R(n-1)$ is arrival time and $S(n)$, $S(n-1)$ is timestamp (time of sending) of n^{th} and n^{th-1} packet.

2.3 Packet Delay Variation

There exist two variants of measure that refers to estimated local mean [5], [8]. Simple Mean Absolute Delay Variation, MAPDV, (Equation (4)) and its more sophisticated version MAPDV2 (Equation (5)).

$$MAPDV = \text{mean}\left(|t(n) - a(n)|\right) \tag{4}$$

where: $t(n)$ is current packet transmission delay, $a(n)$ is nominal (average) transmission time. One can easily notice that this approach requires prior knowledge of default transmission time, which means using local mean or median estimate to adopt it to short term jitter evaluation. The more complicated measure is MAPDV2 regarding explicitly short term average transmission delay. It is computed according to following formula: estimate a mean delay $a(n)$ using Jacobson's estimator [9] with gain set to 1/16 as a first step

$$a(n) = \frac{1}{16}t(n-1) + \frac{15}{16}a(n-1)$$

where: $t(n-1)$ transmission delay of recent packet; $a(n)$, $a(n-1)$ new and former estimate of mean transmission delay. In next step such approximate value is used in following computations given below as meta code for last 16 packets:

```
for (i=n-16; i<n; i++)        // n - current packet number
  if t(i)>a(i)                  // i - former packet numbers
    P(i)=P(i)+(t(i)-a(i))     // positive deviation
  elseif t(i)<a(i)
    N(i)=N(i)+(a(i)-t(i))     // negative deviation
  end if                      // if t(i)==a(i) do nothing !!!
end for
```

In a third step MAPDV2 is calculated as sum of mean values of positive and negative delays of 16 recent packets:

$$MAPDV2 = \text{mean}(P(i)) + \text{mean}(N(i)) \ . \tag{5}$$

2.4 IPDV and Relatives

Another existing solution for estimating a jitter is moving range statistic. To the author's knowledge, there are two similar solutions: one defined by ITU [10] called also IPDV (IP Delay Variation) given with formula:

$$IPDV_{\text{ITU}} = IPTD_{\text{up}} - IPTD_{\text{min}} \tag{6}$$

where: $IPTD_{\text{min}}$ is minimum IP transmission delay, $IPTD_{\text{up}}$ is some upper percentile (99.9^{th} is commonly used) thus $IPDV_{\text{ITU}}$ represents variability range of transmission delay. The other approach $IPDV_{\text{OWAMP}}$ to this proposal is used in OWAMP [3] implementation where client evaluates jitter as difference between median and upper percentile but there are used $IPTD_{\min}$ – median value of IP transmission delay and as $IPTD_{\mathrm{up}}$ transmission delay 95$^{\mathrm{th}}$ percentile.

Results obtained by using such estimators are affected by the moving window length for which the statistics are calculated. If the length of window is too short, the resulting values would be bound to few recent packets so predictor wouldn't exhibit any generalization. Otherwise if it is too long, it couldn't be used for predicting momentary value but would give general overview characteristics. Window of 200 ms used in VoIP applications seems to be a good compromise.

2.5 SMPDV

The SMPDV is an acronym of Switched Measure Packet Delay Variation estimator [11]. The concept is somewhat similar to classical EWMA AR-like estimator (Equation (3)). It is given using formula:

$$J(n) = A|D(n)| + B|D(n-1)| + RJ(n-1) \tag{7}$$

where $J(n)$, $J(n-1)$ are current and preceding SMPDV estimate, and $D(n)$, $D(n-1)$ are current and previous packet delay values, A, B, R are model coefficients fit experimentally having values 1/2, 1/16, 7/16 respectively.

The key fact is that the estimating of jitter is based on larger value chosen from two packet delay variation values as given below:

$$D(n) = \max\{D_{\mathrm{ap}}(n), D_{\mathrm{pp}}(n)\} \tag{8}$$

where: $D(n)$ is current delay value, $D_{\mathrm{pp}}(n)$ is inter-packet delay, $D_{\mathrm{ap}}(n)$ is absolute packet delay with reference to some estimate of base transmission time. They are described with following equations:

$$D_{\mathrm{pp}}(n) = |t(n) - t(n-1)|\,,\ D_{\mathrm{ap}}(n) = |t(n) - a_{\mathrm{m}}(n)|$$

where: $t(n)$, $t(n-1)$ are transmission delays of current and previous packets, $a_{\mathrm{m}}(n)$ is base transmission time estimate over some of past transmission delays which is median estimator over last 16 packets:

$$a_{\mathrm{m}}(n) = \mathrm{median}\,\{t(n), t(n-1), \ldots, t(n-15)\}$$

where: t(n-1), t(n-2), t(n-15) are transmission delay values of 16 previous packets.

3 OWAMP

OWAMP protocol and server are modern tools designed to measure network latencies. Since common ICMP echo packets (ping) is often classified by routers as low priority traffic so it cannot be used as reliable network performance measurement tool. Also conclusions about one way transmission on the round trip delay basis seems to be questionnable due to route asymmetry and changes [12]. General idea of OWAMP is based on sending as packet contents 64 bit long unsigned

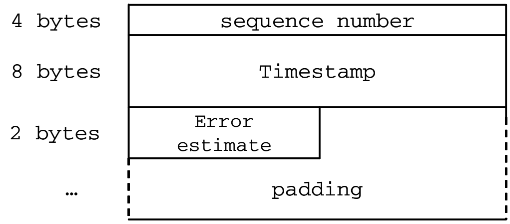

Fig. 2. Basic OWAMP unencrypted testing packet format

timestamps (see Fig. 2) derived from sender's well synchronized clock. Receiver complements packets with timestamps of arrival times from its own clocks that should be also well calibrated. It is obvious that the clock calibration is crucial for the precise OWAMP measurements. It is therefore recommended to deploy within the same computer a NTP server having at least 4 peer synchronization sources (or good reference, e.g., atomic clock) that would continuously calibrate the clock. The 64 bit long timestamp consists of two parts – integer and fractional. High-order 32 bits are integer part counting seconds since January 1st, 1900, low-order 32 bits are fractional part of second with theoretic precision up to $1/(232 - 1) \approx 2.33\ 10^{-10}$ s. Additional noteworthy information is that basic OWAMP packet as depicted in Fig. 2 has minimal size of 14 bytes meanwhile its encrypted form has 32 bytes.

In OWAMP system there are two parties involved in the testing procedure (Fig. 3) – the client and the server which results in two main usage scenarios that differ in the direction of the test packets transmission: the test packets are sent from client (owping) to server (owampd) – after the test transmission the server sends reports through the control connection containing send and arrival timestamps for each packet (among the others); test packets are sent on

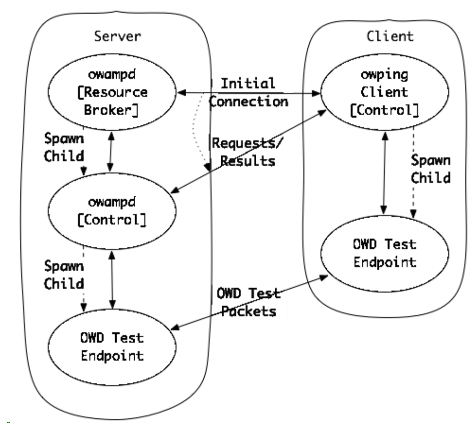

Fig. 3. OWAMP architecture [3]

client's request from server so it can compare their arrival time with the original timestamp included with the packet.

In opposite to usual ping, OWAMP is not commonly installed at the Internet connected hosts as it is rather specialized measurement tool but according to OWAMP promoting website [3] there are numerous PMPs available in the Internet. Very few of them allow for a free access, it is usually required to obtain AES keys or set up a ser account to be created on the OWAMP server. So usually it is necessary to have contact to the PMP administrator to get access to it.

4 Experimental Setup and Data Sets

To collect data sets to deal with jitter estimators the Compaq PIII host running FreeBSD OS has been used. The machine was running GPS driven NTP server. OWAMP client and server was installed on that machine as well. To simulate real traffic we decided to model VoIP transmission using following parameters: the very basic G.711 codec was used where the data is sampled at 8 kHz and it is packetized into 20 ms frames so there are transmitted 50 RTP packets per second, each containing 160 bytes of data plus additional overhead related to IP/UDP/RTP protocols stack headers – overall packet size is 196 bytes. To emulate real network workload it was necessary to set the OWAMP packet padding to fit the size of real transmission, which was straightforward as OWAMP uses the same lower layer protocols as VoIP transmission. Since IP/UDP/OWAMP packet size is 38 bytes we used 158 bytes padding.

Traces are 60 seconds (3000 packets) logs of two way OWAMP transmissions from and to the servers – as most of PMPs were inaccessible to us, we used the only open one in Korea (134.75.29.10) and we set up one in France (188.165.195.68) by courtesy of one of the hosting companies. The data was gathered on four successive days Saturday-Tuesday (twice during workdays) so there were 24 data sets collected. The data sets were manually classified (see Table 1) on a basis of observed episodes of types of jitter and relation between whole set $IPDV_{ITU}$ to the target inter packet offset (ΔT=20 ms). We observed the following classes:

- low variance constant jitter ($IPDV_{\mathrm{ITU}}/\Delta T < 0.05$) with no or few transient jitter episodes,
- low variance constant jitter ($IPDV_{\mathrm{ITU}}/\Delta T < 0.05$) with several episodes of transient jitter,
- moderate variance constant jitter ($0.05 < IPDV_{\mathrm{ITU}}/\Delta T < 0.2$) with numerous episodes of transient jitter,
- heavy variance ($IPDV_{\mathrm{ITU}}/\Delta T > 0.2$) with short term delay changes,
- low and moderate variance constant jitter sets with few episodes (1–3) of transmission jamming – packets bursts delayed above 400 ms and arriving almost at once after lag were considered to be lost so data sets were then reclassified on basis of their proper transmission.

Table 1. Test sets classification

	from Korea	to Korea	from France	to France
Saturday	B	A	A	A
Sunday	A	A	A	A
Monday 9:30	A	E(A)	A	A
Monday 12:30	D	C	D	D
Tuesday 9:30	C	E(C)	E(B)	C
Tuesday 12:30	C	C	C	B

5 Estimators Evaluation

As a fundamental criterion for judging efficiency of estimator we assumed the difference between real and predicted transmission delay. Such approach might be too arbitrary so we decided to consider also a number of underestimated packets and mean gap between underestimated and real delay value. The best results was highlighted in each table.

5.1 Predictor Precision

Predicted delay was calculated using formula:

$$P(n) = J(n) + a(n) \tag{9}$$

Mean absolue difference was calculated as:

$$E = \frac{1}{N} \sum_{n=1}^{N} |P(n) - D(n+1)| \tag{10}$$

where: $P(n)$ – predicted delay value, $J(n)$ – predicted jitter value, $D(n+1)$ – corresponding real transmission delay, $a(n)$ – running central tendency measure. In our case as $a(n)$ we decided to use moving median over last 16 packets as in our former research [11] it appeared to outperform simple moving average or EWMA. Results grouped by dataset class are presented in the Table 2.

Table 2. Mean absolute prediction error [ms] per data set in group

Group	MPPDV	RTCP	MAPDV	MAPDV2	$\mathrm{IPDV}_{\mathrm{ITU}}/2$	$\mathrm{IPDV}_{\mathrm{OWAMP}}$	SMPDV
A	0.410	0.410	**0.360**	0.543	0.555	0.742	0.446
B	0.475	0.476	**0.416**	0.612	0.645	0.884	0.517
C	1.368	1.374	**1.250**	1.960	1.872	2.724	1.502
D	3.802	3.774	**3.508**	5.090	4.849	6.105	4.199
E	7.619	7.076	10.007	11.881	10.494	12.363	**6.378**
All	1.987	**1.920**	2.196	2.907	2.689	3.401	1.936

Surprising fact is that relatively simple predictor MAPDV appeared to have better performance than sophisticated ones in all but three cases. The only notable difference is for those sets where transmission jams were observed in such cases SMPDV appeared to be the most exact one. One should remember that simple mean absolute difference gives just rough view on efficiency of the predictors so there were required in-depth considerations on the underestimated packet delay.

5.2 Negative Cases Analysis

Since absolute difference between predicted and real delay value give just brief view to estimators performance we decided to analyze also performance in negative cases where predicted delay was underestimated. We measured it in two ways: as a number of underestimated packets and an average difference for underestimated packet. Number of underestimated packets is shown in Table 3.

Table 3. Mean number of underestimated packet delays per data set in group

Group	MPPDV	RTCP	MAPDV	MAPDV2	$IPDV_{ITU}/2$	$IPDV_{OWAMP}$	SMPDV
A	440.444	485.667	668.778	312.000	330.778	**261.556**	512.333
B	430.500	472.500	636.000	319.500	339.000	**265.000**	531.500
C	510.667	555.167	685.833	366.167	423.500	**280.333**	598.500
D	557.667	626.333	741.333	358.333	424.000	**304.333**	558.000
E	**214.333**	512.667	612.667	328.000	407.000	280.667	579.667
All	436.750	519.542	669.708	331.625	374.667	**274.292**	547.625

Besides the number of incorrectly estimated packets there is important question how 'wrong' was the prediction therefore we calculated mean prediction difference for underestimated packets. The resulting values are shown in Table 4. The best (minimal) values are highlighted.

Table 4. Mean difference for underestimated packet delays per data set in group [ms]

Group	MPPDV	RTCP	MAPDV	MAPDV2	$IPDV_{ITU}/2$	$IPDV_{OWAMP}$	SMPDV
A	0.1290	0.1113	**0.1011**	0.1222	0.1177	0.1229	0.1048
B	0.1477	0.1414	**0.1293**	0.1556	0.1466	0.1459	0.1303
C	0.5108	0.4606	0.4341	0.4712	0.4484	0.4701	**0.4131**
D	0.9764	0.9304	0.9300	0.9088	0.8775	**0.8739**	0.8782
E	19.1802	7.2951	5.0090	6.9569	7.7705	10.4780	**2.0139**
All	2.7212	1.2066	0.9070	1.1737	1.2614	1.6081	**0.5237**

6 Conclusions

We proposed the testing procedure and evaluated the performance of a number of jitter estimators. The test was performed against the test data gathered using OWAMP server using the scenario as close to real life application as it was possible to achieve.

Obtained results are ambiguous. The minimal mean absolute differences are observed for simple $MAPDV$ predictor except the situation where occurs transmission jamming – in such cases $SMPDV$ seems to be the most precise predictor. Overall best performance was achieved for EWMA used in RTCP. The worst results were obtained for $IPDV_{\mathrm{OWAMP}}$ but on the other hand it seems to be the most protective as it underesti-mates the least number of packets but the cost is in the fact that its precision is the lowest one due to relatively large and protective overestimation margin. Finally, when taking into account not only the number of underestimated packets but also the size of that underestimation then it appears that $SMPDV$ seems to be the most reliable.

Summarizing, its hard to point a universally best estimator. Choice should be made depending on the application area. Both the data and procedure are reusable it is possible to perform such experiment using other data and other estimators. Our further research in this area will focus on scenario based analyses using synthetic data sets to test behavior and properties of estimators in case of specfific jitter cases.

Acknowledgments. This work has been partially supported by the Polish Ministry of Science and Higher Education grant number N N516 375034.

References

1. ITU-T Recommendation G.114: One-way transmission time. ITU (2003)
2. Shalunov, S., et al.: A one-way Active Measurement Protocol (OWAMP). IETF RFC4656 (2006)
3. One-Way Ping, `http://www.internet2.edu/performance/owamp/`
4. Morton, A., Claise, B.: Packet Delay Variation Applicability Statement, IETF RFC5481 (2009)
5. Clark, A.: Analysis, measurement and modelling of Jitter. ITU-T Delayed Contribution COM 12 - D98, ITU, Geneva (2003)
6. Demichelis, C., Chimento, P.: IP Packet Delay Variation Metric for IP Performance Metrics (IPPM). IETF RFC3393 (2002)
7. Schulzrinne, S., et al.: RTP: A Transport Protocol for Real-Time Applications. IETF RFC3350 (2003)
8. ITU-T Recommendation G.1020: Performance Parameter Definitions for Quality of Speech and other Voiceband Applications Utilising IP Networks. ITU (2003)
9. Jacobson, V.: Congestion Avoidance and Control. In: Proc. ACM SIGCOMM 1988, pp. 314–329 (1998)
10. ITU-T Recommendation Y.1540: Internet protocol data communication service – IP packet transfer and availability performance parameter. ITU (2007)
11. Skurowski, P., Gruca, A.: SMPDV – A New Jitter Estimator Proposal. Studia Informatica 29(4B), 37–47 (2008)
12. Anagnostakis, K.G., Greenwald, M., Ryger, R.S.: cing: Measuring Network-Internal Delays using only Existing Infrastructure. In: Proc. IEEE INFOCOM 2003 (2003)