ISPCS 2008 – International IEEE Symposium on Precision Clock
Synchronization for Measurement, Control and Communication
Ann Arbor, Michigan, September 22–26, 2008

Characterizing network synchronization potential with the minTDEV statistic

Greg Dowd
George Zampetti
Symmetricom
2300 Orchard Parkway
San Jose, CA 95131 USA

Abstract *– A growing number of applications today are based on the transfer of time and/or frequency over packet networks. This has created a requirement for new methods to support the modeling, testing and implementation of packet based time services. This paper describes a new metric called minTDEV. An overview of the Allan variance family and how this metric fits in is provided. A conceptual and formal definition of the metric is provided. Results of a test of the suitability of minTDEV to check compliance of a reference clock model are presented.*

Keywords *– PTP, Packet Synchronization, Allan variance, network modeling, minTDEV*

I. NETWORK BEHAVIOR

There is a large and growing body of knowledge characterizing packet networks as exhibiting self-similar load characteristics [1,2]. While self-similar behavior can be described in a number of ways such as 1/f noise or Hurst parameter 0.5 < H < 1, a key observation is that these types of processes have correlated second-order statistics over multiple observation intervals. This behavior has significant impact on the traditional Allan variance metric used to characterize performance in frequency systems, TDEV.

A. Network Modeling

Traditional measurements of frequency have been based on power spectral noise analysis using metrics such as the modified timing deviation (TDEV). This reflects the common understanding that frequency systems can be analyzed as a set of noise processes and how these processes interact in a spectral domain [3]. However, as network packet models were built and extended, analysis of the data showed that TDEV did not always adequately characterize the signal being studied [4]. Figure 1 shows the distribution function for a 2-switch model. A strong signal appears at the minimum (left) edge of the distribution, corresponding to those "lucky" packets experiencing a minimum of interruption in transmission. Intuitively, it appeared that a modification to the way TDEV was being calculated, to operate on these minimum delay samples, might provide a better understanding of network delay variation.

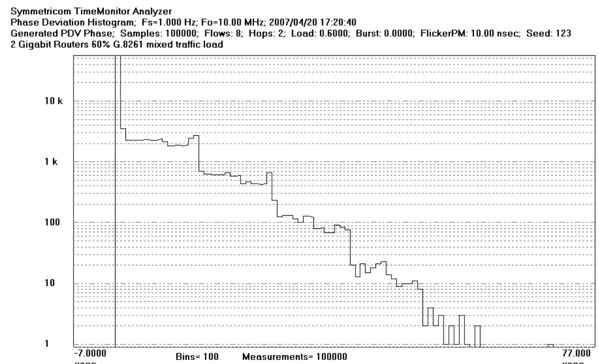

Figure 1. Simulated 2 switch chain

B. Modified TDEV Comparison

The results of the modified calculation versus the standard calculation of TDEV against two network simulations are shown in Figure 2. The two lower curves represent minTDEV plots while the upper two show corresponding TDEV plots for two typical network configurations. For the scenarios tested, the modified calculation converged earlier than the standard TDEV.

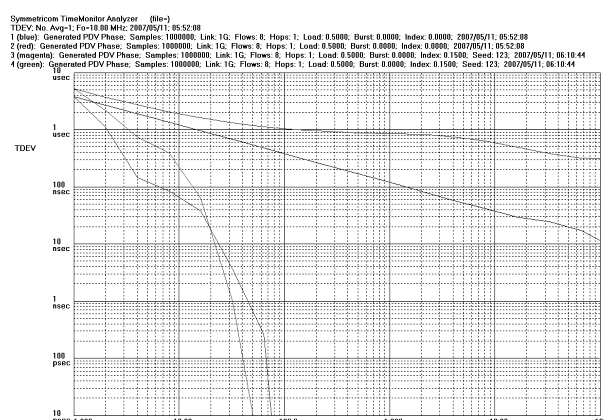

Figure 2. TDEV vs. minTDEV model comparison

II. MIN_TDEV ANALYSIS

A more formal analysis of the "new" metric, coined "minTDEV" was conducted, and its performance was contrasted with the performance of TDEV [5][6]. The results showed that the two metrics could complement each other in the analysis of network packet behavior.

The metric was analyzed under various test scenarios [7] and the results from the power-law noise cases are shown below in the table.

	Power Law (log-log) slope	
Noise Type	TDEV	minTDEV
Uniform White Noise	-1/2	-1
Gaussian White Noise	-1/2	-1/2<minTDEV<0
Flicker Noise	0	0
Random Walk	1/2	1/2

Figure 3. Power law noise comparision

An interesting result deals with stationary noise processes where the distribution function shows the critical issue. In the tradition oscillator model where the distribution function is two-sided (Gaussian White Noise), TDEV converges more quickly (-1/2 vs –1/2<minTDEV<0). In cases where the distribution shows a strong lower tail (Uniform White Noise), minTDEV converges first (-1 vs –1/2).

Another test was conducted using a network packet simulator. The simulator setup used a combination of noise processes to model network packet delay. The first process models physical layer noise. This includes noise associated with symbol clock mapping as well as transmission jitter/wander. The noise is a function of the number of mappings, with multiple mappings migrating to a two-sided distribution via the central limit theorem. The second noise process models the queuing delay introduced by switching elements within the network path. Queuing can only delay forwarded packets and is modeled as a long tail distribution that corresponds to the behavior observed in live networks. Modeling of the self-similar behavior of the network flows was not included in this Monte Carlo simulation.

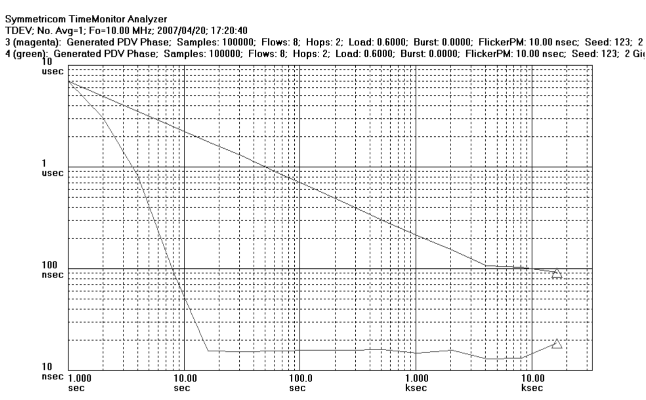

Figure 4. TDEV vs minTDEV (2 switch 60% load)

The results of the test show that minTDEV is effective at characterizing the underlying signal in the presence of queuing noise. In this test, minTDEV found the floor at τ = 20s while TDEV followed later at τ = 3600s. Also, the flicker floor as measured by TDEV is an order of magnitude higher for the same signal. Subsequent testing of larger network chains, not detailed, showed a growing separation in the observation interval required (τ) by the metric to find the floor.

III. ALLAN VARIANCE FAMILY

The history of the Allan Variance family has included a series of extensions to the metric as new needs arose. Figure 5 shows this history along with a view of where the minTDEV metric fits into the picture [8].

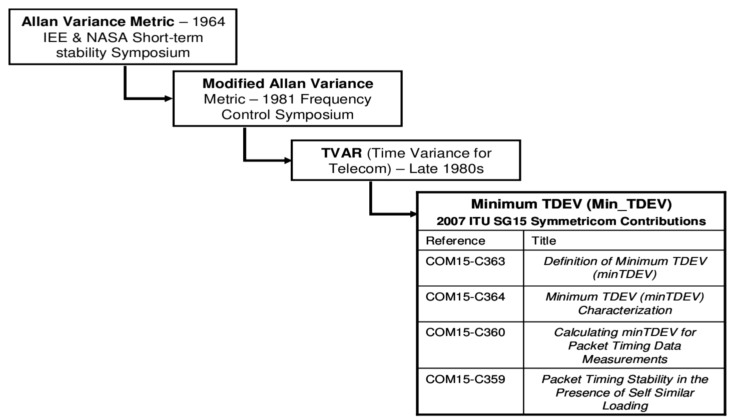

Figure 5. Historical view of Allan variance family

C. Overview of Allan Variance Metrics Family

Allan variance is a metric that generates a stability estimate that is a function of window size.

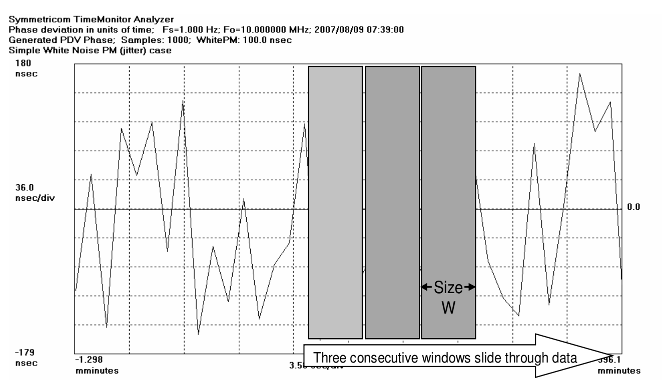

Figure 6. Sliding window concept

Figure 6 shows a plot of evenly spaced time error measurement data (1 second spacing). Superimposed on the plot are a group of three consecutive windows of size W. Window size is usually selected as a doubling sequence

(1sec, 2sec, 4 sec…). For each selected window size, the group of three windows is placed at the earliest (far left) portion of the time series of data and slid one sample at a time to most recent (far right) data.

The next step is to process the three time error samples, to generate a filtered time error value that represents the group of three windows. At each point that the windows slide to, we take the three time error values and calculate F(i)= X1-2*X2+X3. This "2nd difference operation" is common to all Allan Variance calculations. It can be viewed as a high pass filter that converts the low frequency content of real world non-stationary noise to filtered data. This process generates a single variance estimate[1]. Variance is a measure of the variability (spread or dispersion).

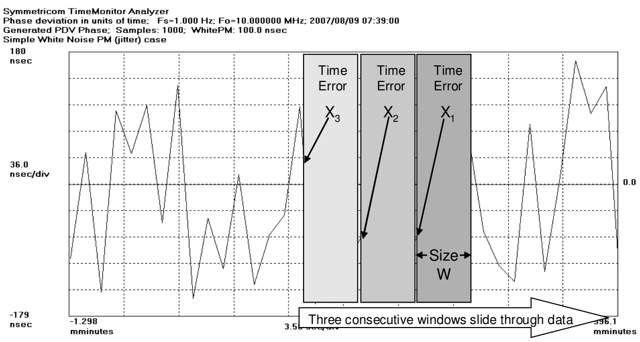

Figure 7. Allan Variance Selection

Figure 7 shows the group of windows at a particular observation point. Within a window we observed a section of the time error data. Common to all Allan Variance operators is the concept of selecting a single time error value to represent the data within the window. How this single time error value is determined is the key differentiation between members of the Allan Variance family. For example if we select the first time error point within a window to represent the entire window as shown, we would obtain three time error values X_1, X_2, X_3. This selection approach is how the classic Allan variance operation is performed.

D. Allan Variance Window Selection

The history of Allan Variance can be understood based on the key differentiator of how a phase estimate is extracted from a window. The time line shows the original Allan Variance estimation technique beginning in the mid-1960s. The primary application was to characterize oscillators. In the discussion about Figure 7, we stated the classic Allan Variance operator selects the first time error sample in the window to represent the entire window. In the case of signals from oscillators that behave very smoothly within a window, using any single point to represent the window seems plausible. This approach works fine for signals that are dominated by the more divergent noise process such as white noise frequency modulation (random run in time error). However, for signals that have short-term time error variation (jitter) this approach is not optimal. To observe the jitter processes we need to look within the window.

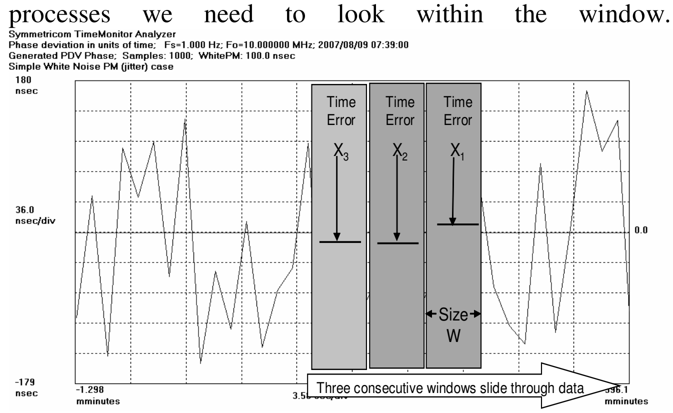

Figure 8. Modified Allan Variance Selection

In the 1980s, Allan Variance was extended to look within the window. Figure 8 shows the averaging approach adopted. Within each window all of the time error data is averaged to produce a single average estimate of time error. This method is used for two Allan Variance operators. The first is termed Modified Allan Variance. Up until now we have considered the stability estimate process to provide an estimate of the time error stability. Intuitively, the same estimate process can be provided an estimate of frequency stability. More precisely the concept of fractional frequency stability is used. Say we have a 10 MHz oscillator that is off by 1Hz from ideal center frequency. Fractional frequency stability is just the ratio of the error to the nominal center (1 Hz/ 10MHz). The fractional frequency stability has no dimensions and we typically use terms like parts per million (ppm) or parts per billion (ppb). For example the 10MHz oscillator offset by +1Hz is 100ppb high. Historically modified Allan Variance was developed first and later scaled to apply to timing signals in telecommunication networks (where the term TDEV or TVAR are used). The simple relationship between the two metrics is:

$$\sigma_y(\tau) = MDEV(\tau) = \frac{\sqrt{3}}{\tau}\sigma_x(\tau)$$

There is now a need to define stability metrics relevant to timing flows in packet networks. Figure 9 shows the introduction of a new Allan Variance window selection concept termed minimum TDEV (or minTDEV). The formal definition of this new metric is presented in the next section however the concept is relatively straightforward.

[1] The actual sequence of operations may be different so long as the principles of superposition of linear operators are obeyed.

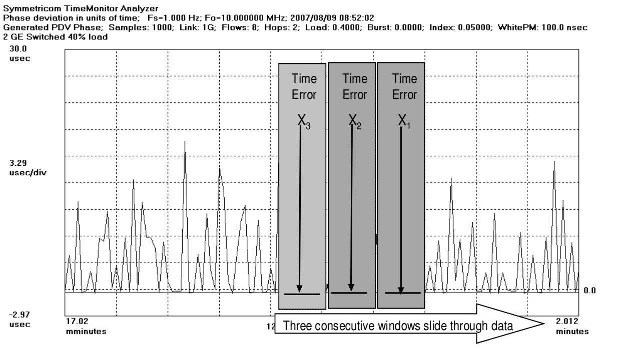

Figure 9. minTDEV Window Selection

Figure 9 shows a plot of time error data representative of a real packet timing flow. The time error data represents two GE switches at 40% load. The packet delay variation is a long tailed distribution. Packets with long delay are very poor samples of the originating clock. Packet clock algorithms are weighted towards the minimum delay samples. The principles underlying this asymmetrical distribution behavior were discussed above. Similar to TDEV we still look in the window; however, for packet timing flows we recognize that an averaging approach is not optimal. Instead we need to extract an estimate of the floor within the window. The simplest approach is to select the minimum value within the window to best represent the true time error between the send and receive clocks.

IV. MINIMUM TDEV (MIN_TDEV) FORMAL DEFINITION

E. Standard TDEV

TDEV is part of a family of Allan variance stability operations that operates on a uniformly spaced sequence of phase data ($\mathbf{x}_n$). Here is the defining equation from Annex C of T1.101 [9]:

$$\sigma_x(\tau) = \mathrm{TDEV}(\tau) = \sqrt{\frac{1}{6n^2}\left\langle\left[\sum_{i=1}^{n}\left(x_{i+2n} - 2x_{i+n} + x_i\right)\right]^2\right\rangle}$$

In the above equation, the angled braces represent an ensemble average.

For power law noise types, the result is the same if the ensemble average is replaced by an infinite time average, provided the square of the second difference averaged over n τ_0 is taken prior to the infinite time average. Note that $\tau = n\tau_0$ where τ_0 is the sampling interval. Some useful insights can be made by applying superposition to change the order of the linear operators. The new form is as follows:

$$\sigma_x(\tau) = TDEV(\tau) = \sqrt{\frac{1}{6}\left\langle\left[\frac{1}{n}\sum_{i=1}^{n}x_{i+2n} - 2\frac{1}{n}\sum_{i=1}^{n}x_{i+n} + \frac{1}{n}\sum_{i=1}^{n}x_i\right]^2\right\rangle}$$

Note that the operation within the square brackets consists of operating on three consecutive time windows of the phase data sequence. This is illustrated below for the case where n=4. The most current samples ($\mathbf{x}_9$- $\mathbf{x}_{12}$) are part of the current window. The current window can be seen in the first summation in the square bracket. The operation of summation over the four samples in the window and dividing by (n=4) can be viewed as simple calculation of the mean or average phase in the window. The same logic applies for the lag window as well as the double lag window. Thus, the first operation is to extract a phase estimate representing the phase data within the window. In the case of the standard TDEV operator this phase estimate is the mean.

Double Lag Window				Lag Window				Current Window			
X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12

Figure 10. Variance Sampling Window

The TDEV operator is directly related to the modified Allan Variance operator (MDEV). MDEV is used to understand the power law noise structure in terms of fractional frequency stability. It is typically applied to free running oscillators. MDEV and TDEV are related in a simple fashion:

$$\sigma_y(\tau) = MDEV(\tau) = \frac{\sqrt{3}}{\tau}\sigma_x(\tau)$$

Historically, the modified Allan Variance was developed to provide a means to differentiate between white noise PM and flicker noise PM. This was achieved by effectively changing how a phase sample is selected within a window. In the original Allan Variance Operator the selection was simply the phase sample at the start of the window. The original Allan Variance operation can be expressed in the following form:

$$\sigma_{y-orig}(\tau) = ADEV(\tau) = \sqrt{\frac{1}{2\tau^2}\left\langle\left[x_{i+2n} - 2x_{i+n} + x_i\right]^2\right\rangle}$$

To more clearly see that the concept that the phase sample selected is the start of the window considering for the current window:

$$x_{i+2n} = \sum_{i=1}^{n}\begin{bmatrix} x_{i+2n}\,for(i=1) \\ 0\,for(i>1) \end{bmatrix}$$

In fact, by substituting selection of the first sample with the window phase mean we get:

$$\sigma_y(\tau) = MDEV(\tau) = \sqrt{\frac{1}{2\tau^2}\left\langle\left[\frac{1}{n}\sum_{i=1}^{n}x_{i+2n} - 2\frac{1}{n}\sum_{i=1}^{n}x_{i+n} + \frac{1}{n}\sum_{i=1}^{n}x_i\right]^2\right\rangle}$$

After rearranging linear operations:

$$\sigma_y(\tau) = MDEV(\tau) = \sqrt{\frac{1}{2n^2\tau^2}\left\langle\left[\sum_{i=1}^{n}\left(x_{i+2n} - 2x_{i+n} + x_i\right)\right]^2\right\rangle}$$

This is the normal form for the Modified Allan Variance operator.

This last exercise illustrates a key observation. The principle difference between the original Allan Variance Operator and the Modified Allan Variance Operator is the manner in which a phase estimate is extracted from a window. In the original case the phase sample is simply the first one in the window while in the modified case the estimate is the average or mean. In fact the purpose of adding the window averaging in the modified case is to discriminate between white noise phase modulation and flicker noise. The averaging function improves the estimation stability for a white noise process but essentially does not affect a flicker noise process.

It is very useful to consider what a valuable phase estimation technique in a window is when the instability is not associated with a free running oscillator but rather delay variation in a packet network.

F. Minimum TDEV

The key to the minimum TDEV operator is a change in the manner the phase estimate is extracted from a window of phase data. The selection of appropriate phase estimation operation is based on the following consideration.

Unlike oscillator noise processes that have two sided distribution functions, the distribution function for packet delay variation is intrinsically one sided. The notion of minimum delay of delay floor is critical. The floor is the minimum delay that a packet (or other protocol data unit such as a layer 2 frame) can experience in a given path[2]. The floor can be viewed as the condition where both output and system queues (in all equipment that is involved in the flow, including the source, destination, and intervening elements) are near their minimum when the particular packet needs the resource. Under normal non-congested loading conditions, a fraction of the total number of packets will traverse the network at or near this floor, even though others may experience significantly longer delays.

In the minTDEV operation the mean of the sample window is now replaced with the minimum of the sample window:

$$x_{mean}(i) = \tfrac{1}{n}\sum_{j=1}^{n} x_{j+i}$$

Replacing the mean with the minimum selector

$$x_{\min}(i) = \min\lfloor x_j \rfloor for(i <= j <= i+n)$$

yields the following.

$$\sigma_{x\_\min}(\tau) = \min\_TDEV(\tau) = \sqrt{\tfrac{1}{6}\left\langle \left[x_{\min}(i+2n) - 2x_{\min}(i+n) + x_{\min}(i)\right]^2 \right\rangle}$$

This is the equation for minTDEV.

[2] In real networks the floor is not strictly constant but may slowly change over time or abruptly change (for example a path reconfiguration). The concept of a stationary floor over all time is not mandatory for minimum TDEV.

V. APPLICATION OF MIN_TDEV TO A PTP REFERENCE CLOCK

In order to study the suitability of the minTDEV metric as a tool for studying network delay and delay variation, a test was designed based on IEEE-1588 [10]. A PTP Grandmaster clock, referenced to a primary reference source was used to synchronize a reference client over a setup of two packet networks with various load conditions.

The grandmaster clock was a Symmetricom PTP blade operating under the telecom profile. The grandmaster was connected to the test network via a Gigabit SFP Ethernet link.

The reference client was an implementation of a Telecom Reference Clock Model [11]. The client was connected to the test network via an electrical 10/100 Ethernet interface. The client used a Type 1 ovenized oscillator. The client used a 64-pkt/second sync and delay request/response rate.

The test network consisted of an edge aggregation router/switch and a Gigabit Ethernet network including up to four additional switches. The network was loaded using Spirent test generators set up to generate G.8261 access loading [12]. The four test cases were

- 3 switches, 20% load
- 3 switches, 80% load
- 5 switches, 20% load
- 5 switches, 80% load

The following metrics were collected for each case:

1) One-way offset of sync packets from the Grandmaster clock via the interconnect network to the PTP reference client. The offset data was observed using two packet timing probes operating in a passive mode at both the ingress and egress points of the network. The probe was instrumented with a Symmetricom packet timing blade operating in passive probe mode.

2) The probe data collected above was post processed to extract an estimate of TDEV, min_TDEV and percentile TDEV. The Symmetricom Time Analyzer application was used to perform this function.

3) The Symmetricom Reference client provides TDEV family Performance Monitoring data. Four windows are reported: 2, 4, 8 and 16 seconds. A noise reduction factor is also reported for enhanced estimation.

The test results are shown for the two corner cases (best case – 20% load with three switches and worse case 80% load with five switches)

G. Test 1 (3 switches, 20% load)

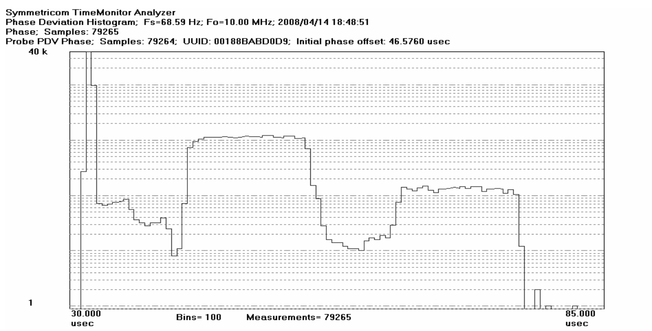

Figure 11. PDV histogram

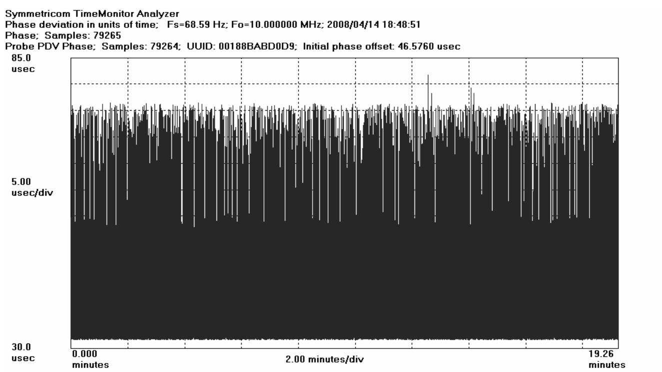

Figure 12. Packet Delay

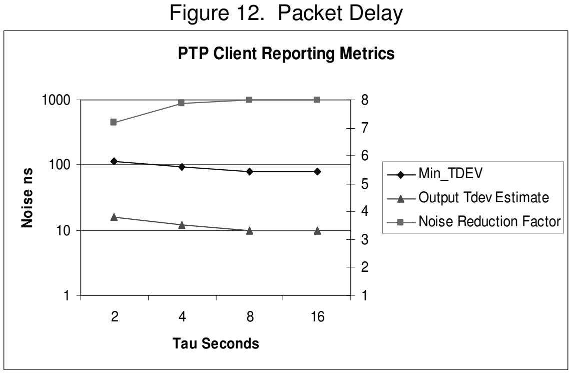

Figure 13. Client Performance Estimate

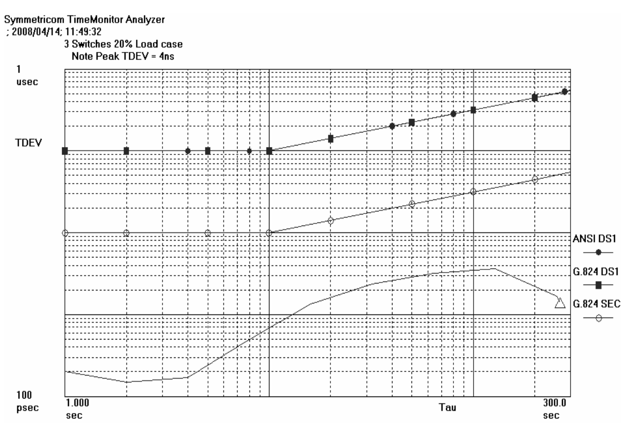

Figure 14. Client Results

H. Test 2 (5 switches, 80% load)

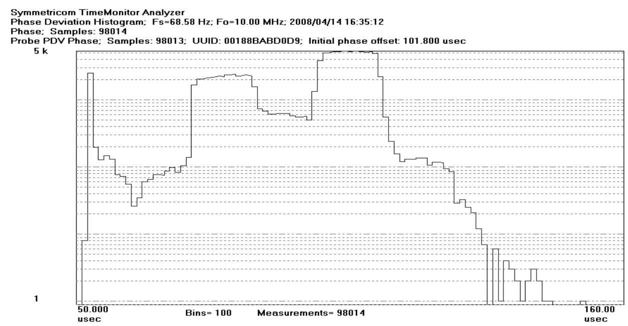

Figure 15. PDV Histogram

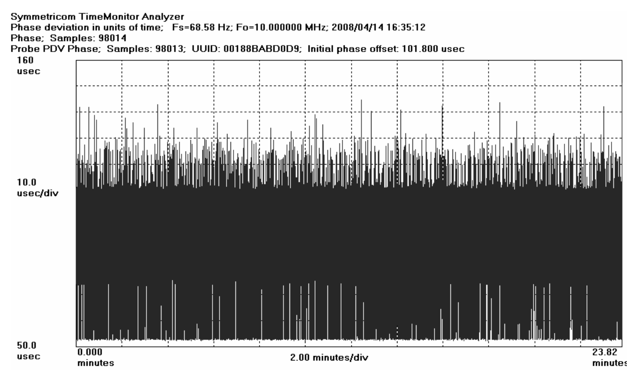

Figure 16. Packet Delay

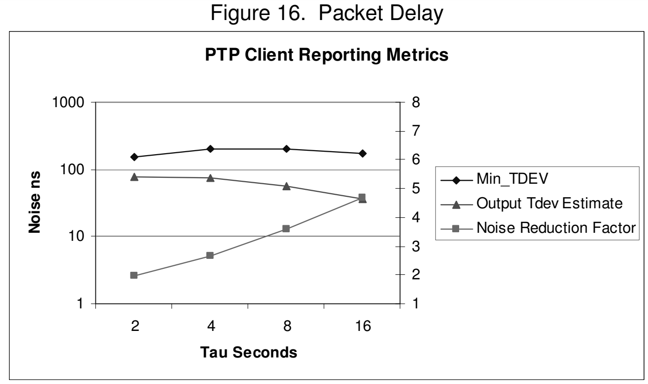

Figure 17. Client Performance Estimate

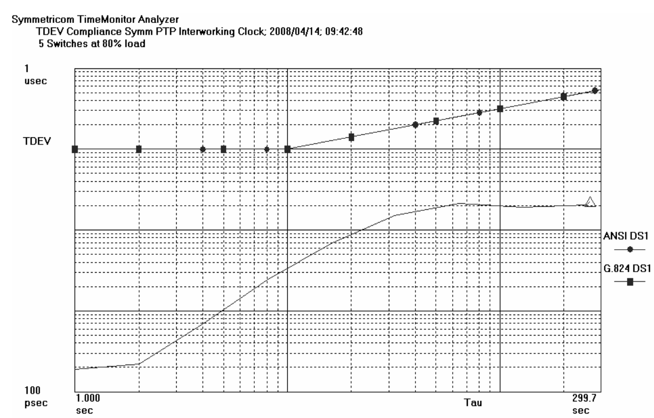

Figure 18. Client Results

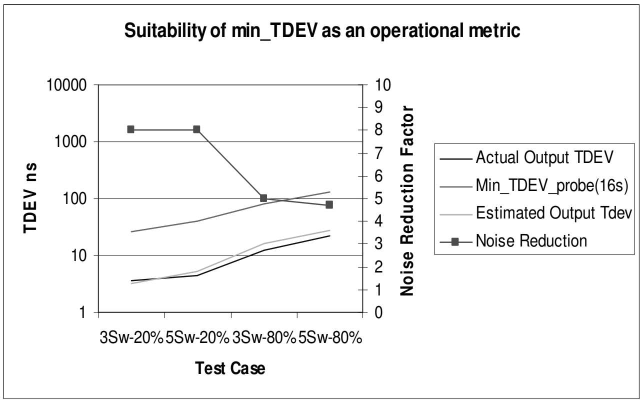

Figure 19. Suitable of minTDEV metric

Figure 19 provides a summary of the testing results for all four cases. The actual TDEV performance (lowest line) shows the maximum Time Deviation instability (in ns) of the reference clock as measured against the house cesium standard. The performance ranges from nominally 4ns (20 Load 3 switches) to a maximum of 22ns (80% Load 5 switches).

The minTDEV performance of the PTP packet flow at the input to the reference clock is also showed (middle line). The minTDEV performance metric provides a good operational bound of the actual client performance. The metric is observed for the 16-second window, as this is the relevant selection period for the reference client.

While the minTDEV estimate provides an upper bound of the real client performance, an enhanced estimated can be determined if the interworking client noise reduction factor is either known or reported. The reference client reports the noise reduction factor. The value is shown in Figure 19 (upper line and the secondary axis). The improvement factor is based on the density of points near the minimum floor as well as the loop dynamics. The improvement factor ranges from 8 down to 4.5 for the four cases. The enhanced performance estimate termed estimated output TDEV (second from bottom) shows excellent agreement with the actual measured performance for all four test cases

VI. CONCLUSIONS

The research shown indicates that the minTDEV metric is a useful metric for analyzing network packet delay and delay variation, particularly as applied to the area of frequency transfer. The definition and derivation of the metric shows how it fits in the family of Allan variance metrics that has supportive of additions as new measurement problems have arisen. The minTDEV metric has been included as an informative annex of the ITU G.8261 specification, indicating a growing interest in new network pdv analysis tools and techniques.

REFERENCES

[1] Leland, Taqqu, Willinger and Wilson, "On The Self-Similar Nature of Ethernet Traffic", IEEE/ACT Transactions on Networking, 1994

[2] Park, Willinger, *Self-Similar Network Traffic and Performance Evalution*, Wiley and Sons, 2000.

[3] D.B. Sullivan, D.W. Allan, D.A. Howe, and F.L. Walls, "Characterization of Clocks and Oscillators", NIST Technical Note 1337, National Institute of Standards and Technology, U.S. Department of Commerce, March, 1990.

[4] Zampetti, "Packet Timing Stability in the Presence of Self Similar Loading" contribution to ITU Q13 Meeting, May 2007, Pub COM15-C359, Symmetricom.

[5] Zampetti, Cosart, "Definition of Minimum TDEV (minTDEV)" contribution to ITU Q13 Meeting, May 2007, Pub COM15-C363 Symmetricom.

[6] Cosart, Zampetti, "Calculating minTDEV for Packet Timing Data Measurements" contribution to ITU Q13 meeting, May 2007, Pub COM15-C360, Symmetricom.

[7] Zampetti, Cosart, "Minimum TDEV (minTDEV) Characterization" contribution to ITU Q13 Meeting, May 2007, Pub COM15-C364, Symmetricom.

[8] D.W. Allan, "Introduction to Time-Domain Standards", 32nd annual NIST Time and Frequency Seminar, Section 1-2-1, June 2007.

[9] *Synchronization Interface Standards for Digital Networks*, ANSI T1.101, ANSI, 1987

[10] Zampetti, "Suitability of min_Tdev and other floor metrics using the Telecom Reference Clock Model" contribution to ITU Q13 Meeting, May 2008, Symmetricom

[11] Zampetti, "Packet Timing (IEEE-1588, NTP) Telecom Reference Clock Model" contribution to ITU Q13 Meeting, Feb 2008, Pub COM15-C644, Symmetricom.

[12] *Timing and synchronization aspects in packet networks*, G.8261/Y.1361, ITU-T, May 2006.