Fast Algorithms for TVAR and MTIE Computation in Characterization of Network Synchronization Performance

STEFANO BREGNI
Dept. of Electronics and Computer Science
Politecnico di Milano
P.zza L. Da Vinci 32, 20133 Milano
ITALY
bregni@elet.polimi.it   http://www.elet.polimi.it/~bregni

Abstract: - Time Variance (TVAR) and Maximum Time Interval Error (MTIE) are historically the main time-domain quantities for the specification of network synchronization performance in telecommunications standards. Nevertheless, plain computation of the TVAR and MTIE standard estimators proves cumbersome in most cases of practical interest, due to their heavy computational weight. In this paper, TVAR and MTIE are first introduced according to their standard definitions. Then, fast algorithms based on recursion and on binary decomposition to compute the TVAR and MTIE standard estimators are provided, which effectively cut down the number of operations needed. The MTIE algorithm based on binary decomposition reduces the number of operations needed to a term proportional to $N\log_2 N$ instead of N^2. Both algorithms allow fast evaluation in most practical situations: even very long sequences of TE samples do not require more than few seconds of data processing for TVAR and MTIE computation.

Key-Words: - clocks, digital communication, jitter, MTIE, SDH, SONET, synchronization, TVAR, wander.

1 Introduction

Since the beginning, a major topic of discussion in standard bodies dealing with network synchronization [1]—[4] has been clock noise characterization and measurement. Among the quantities considered in international standards for specification of phase stability requirements, Time Variance (TVAR) (or equivalently its square root Time Deviation, TDEV) and Maximum Time Interval Error (MTIE) have played historically a major role for characterizing synchronization performance in digital telecommunications networks [5]—[12].

In this paper, TVAR and MTIE are first introduced according to their formal definitions. Then, the main issue of their experimental measurement is pointed out: the heavy computational weight in most cases of practical interest, due to the number of operations nested in the direct, plain calculation of their standard estimators. Therefore, fast algorithms to compute the TVAR and MTIE standard estimators are provided, which effectively cut down the number of operations needed. Based on these algorithms, characterization of synchronization performance in digital telecommunications networks is made feasible even on long sequences of Time Error (TE) samples.

2 Definition of TVAR and MTIE

Thorough treatments of clock stability characterization are provided by survey papers [12]—[15]. Moreover, detailed treatises of MTIE and of its properties can be found in [16][17]. In this section, solely the main definitions are summarized for the sake of understanding and to provide the reader with the background concepts.

A general expression describing a pseudo-periodic waveform which models the timing signal $s(t)$ at the clock output is given by [13]—[18]

$$s(t) = A\sin\Phi(t) \tag{1}$$

where A is the peak amplitude and $\Phi(t)$ is the *total instantaneous phase*, expressing the ideal linear phase increasing with t and any frequency drift or random phase fluctuation.

The generated *Time* function $\mathrm{T}(t)$ of a clock is defined, in terms of its total instantaneous phase, as

$$\mathrm{T}(t) = \frac{\Phi(t)}{2pn_{\mathrm{nom}}} \tag{2}$$

where n_{nom} represents the oscillator nominal frequency. It is worthwhile noticing that for an ideal clock $\mathrm{T}_{\mathrm{id}}(t)=t$ holds, as expected. For a given clock, the *Time Error* function $\mathrm{TE}(t)$ (in standards also called $x(t)$) between its time $\mathrm{T}(t)$ and a reference time $\mathrm{T}_{\mathrm{ref}}(t)$ is defined as

$$x(t) \equiv \mathrm{TE}(t) = \mathrm{T}(t)-\mathrm{T}_{\mathrm{ref}}(t) \tag{3}.$$

The *Time Variance* (TVAR) has been introduced aiming at measuring time stability. It is defined as

$$\mathrm{TVAR}(t) = s_x^{\,2}(t) = \frac{t^2}{3}\mathrm{Mod}\,s_y^{\,2}(t) \tag{4}$$

where $\mathrm{Mod}\sigma_y^2(t)$ is the Modified Allan Variance [12]—[15], well known in the field of frequency stability measurement. TVAR has dimension [time^2] and has been widely adopted in telecommunications international standards for the specification of timing interfaces. *Time Deviation* (TDEV) is defined as the square root of TVAR.

The *Maximum Time Interval Error* function MTIE(t,T) is the maximum peak-to-peak variation of TE in all the possible observation intervals t (in former standards [5][6] denoted as S) within a measurement period T (see Fig. 1) and is defined as

$$\mathrm{MTIE}(t,T)= \max_{0\le t_0\le T-t}\left\{\max_{t_0\le t\le t_0+t}\left[\mathrm{TE}(t)\right]-\min_{t_0\le t\le t_0+t}\left[\mathrm{TE}(t)\right]\right\} \quad (5).$$

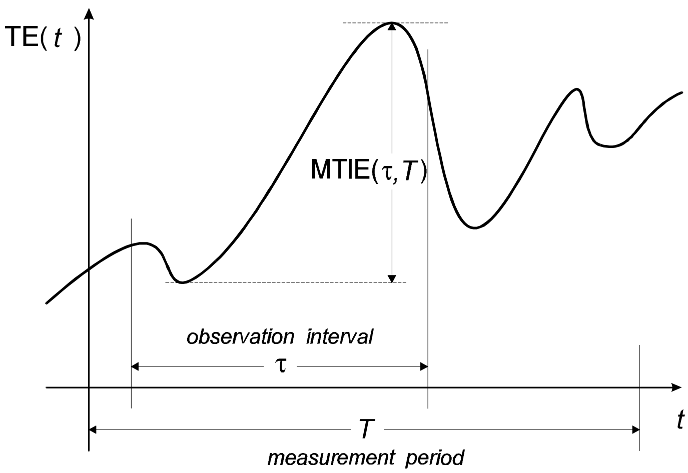

Fig. 1. Definition of MTIE(t,T).

3 Measurement of TVAR and MTIE and Their Standard Estimators

Measurement of TVAR and MTIE is based on the time-domain measurement of the TE process $x(t)$ between the output of the Clock Under Test (CUT) and a reference timing signal, which may be its input if the CUT is a slave clock (*synchronized clocks configuration*), or the output of a second Reference Clock if the CUT is a free-running clock (*independent clocks configuration*) [7][11]. Sequences of N TE samples $\{x_i\}$, defined as

$$x_i = x\left(t_0+(i-1)t_0\right) \qquad i=1,2,3,\ldots N \quad (6),$$

where t_0 is the initial observation time and t_0 is the sampling period, are measured using digital counters and stored for numerical post-processing over a total measurement period $T=(N-1)t_0$ [12][16]. The samples x_i are typically measured between two corresponding zero-crossings of the timing signals involved.

Starting from the sequence $\{x_i\}$ of TE samples measured, the definitions (4)(5) may be applied directly to compute TVAR(t) and MTIE(t,T). Thus, the following standard estimators have been defined by the ITU-T [7] and ETSI [11] bodies:

$$\mathrm{TVAR}(t)= \frac{1}{6n^2(N-3n+1)}\sum_{j=1}^{N-3n+1}\left[\sum_{i=j}^{n+j-1}\left(x_{i+2n}-2x_{i+n}+x_i\right)\right]^2 \quad (7)$$

$$n=1,2,\ldots,\left\lfloor\frac{N}{3}\right\rfloor$$

$$\mathrm{MTIE}(t)= \max_{1\le k\le N-n}\left[\max_{k\le i\le k+n}x_i-\min_{k\le i\le k+n}x_i\right] \quad n=1,2,\ldots,N-1 \quad (8).$$

where $t=nt_0$ is the observation interval and $\lfloor z\rfloor$ denotes "the greatest integer not exceeding z".

4 Fast Computation of TVAR Estimator by Recursion Algorithm

Plain computation of the TVAR standard estimator (7) requires execution of two nested summation loops, thus yielding a computational complexity in the order of N^2 operations for each value TVAR(t) to compute. To save evaluation time, the expression (7) can be written as follows:

$$\mathrm{TVAR}(t)=\frac{1}{6n^2(N-3n+1)}\sum_{j=1}^{N-3n+1}T_j^2(n) \quad (9)$$

where

$$T_j(n)=\sum_{i=j}^{n+j-1}\left(x_{i+2n}-2x_{i+n}+x_i\right) \quad (10)$$

with

$$n=1,2,\ldots,\left\lfloor\frac{N}{3}\right\rfloor \qquad j=1,2,\ldots,N-3n+1 \quad (11).$$

The computational weight of this expression can be reduced by noting that the terms $T_j(n)$ can be evaluated recursively, as

$$T_1(n)=\sum_{i=1}^{n}\left(x_{i+2n}-2x_{i+n}+x_i\right)$$
$$T_{j+1}(n)=T_j(n)+\left(x_{3n+j}-3x_{2n+j}+3x_{n+j}-x_i\right) \quad (12).$$

The first term $T_1(n)$ requires $3n$-1 additions, but the next terms $T_j(n)$ for j>1 can be calculated with only four further additions.

5 Fast Computation of MTIE Estimator by Binary Decomposition

Plain computation of the MTIE standard estimator (8) has computational complexity in the order of N^2 operations for each value MTIE(t,T) to compute. This Section describes the fast algorithm based on binary decomposition proposed originally in [19]. With this algorithm, the number of operations needed is reduced to a term proportional to $N\log_2 N$ instead of N^2. A heavy computational saving is therefore achieved, thus making feasible MTIE evaluation based on even long sequences of Time Error (TE) samples.

5.1 Plain Computation of the Estimator

Let $N=N_T=T/t_0+1$ be the total number of available TE samples in the sequence $\{x_i\}$ and $N_t=t/t_0+1$ be the number of samples available in a window (observation interval) of span t. Then, for *each* single value MTIE(t,T) the following expression has to be computed (cf. eq. (8)):

$$\mathrm{MTIE}(t,T) = \max_{j=1}^{N_T-N_t+1}\left[\max_{i=j}^{N_t+j-1}(x_i) - \min_{i=j}^{N_t+j-1}(x_i)\right] \quad (13).$$

As pointed out in [16], the number of samples N_T to process may get easily to the order of 10^5 in most cases of practical interest, if we are interested in a somehow accurate characterization of the clock noise. It is obvious that the plain computation of the estimator (13) is unadvisable and quickly tends to be unmanageable, due to the number of operations nested in evaluation loops.

5.2 Binary-Decomposition Algorithm

The fast algorithm is based on a binary decomposition of a TE sequence $\{x_i\}$ made of $N=N_T=2^{k_{\mathrm{MAX}}}$ samples in nested windows made of $N_t=2^k$ samples (k=1, 2, 3, ..., k_{MAX}). MTIE can be then evaluated recursively for each window size 2^k.

As the first step (k=1), all the possible 2-points windows ($t=t_0$) are analyzed in the TE sequence: for each of them, the maximum and minimum values are stored. Their difference is the MTIE(t_0) measured in that window, and the maximum of the MTIE values of all the 2-points windows is the resulting MTIE(t_0,T) of the whole sequence. At this first step, there is no computational saving yet compared to the plain computation of the standard estimator.

Then, as second step (k=2), all the possible 4-points windows ($t=3t_0$) are considered. The maximum and minimum values of each of these windows can be obtained by comparing the maximum and minimum values of the two 2-points windows in which the 4-points window can be split. The difference between the maximum of the two maxima and the minimum of the two minima is the MTIE($3t_0$) measured in that 4-point window. The maximum of the MTIE values of all the 4-point windows is the resulting MTIE($3t_0$,T) of the whole sequence.

The next step (k=3) is to consider all the possible 8-points windows ($t=7t_0$), split in two 4-points windows. Then so on, for increasing integer values of k. The computational saving of this algorithm, compared to the plain computation of the standard estimator, lies in avoiding the comparison of all the samples in the windows of size larger then 2. The price to pay is that we have to limit the evaluation of MTIE(t,T) just to the $\log_2 N_T$ values corresponding to the windows made of $N_t=2^k$ samples (this corresponds to a bit more than three MTIE values per decade on the t axis, which may be considered adequate in most practical applications).

More formally, starting from the TE sequence vector $\mathbf{x}$ made of $N_T=2^{k_{\mathrm{MAX}}}$ TE samples x_i, two matrices $\mathbf{A}_{\mathrm{M}}$ and $\mathbf{A}_{\mathrm{m}}$ are built. Matrices are made of N_T-1 columns (indexed by i) and $\log_2 N_T$ rows, indexed by k. The first N_T-2^k+1 elements of each k-th row of the matrix $\mathbf{A}_{\mathrm{M}}$ contain the maximum values of all the possible 2^k-points windows sliding from left to right along the TE sequence $\{x_i\}$. The matrix $\mathbf{A}_{\mathrm{m}}$ contains, in an analogous fashion, the corresponding minimum values of the 2^k-points windows. Therefore, the set of all the possible 2^k-points windows in the whole TE sequence is completely described by the couple of vectors

$$\begin{aligned}\mathbf{a}_{\mathrm{M}/k} &= \{a_{\mathrm{M}/k,i}\}\\ \mathbf{a}_{\mathrm{m}/k} &= \{a_{\mathrm{m}/k,i}\}\end{aligned} \qquad i=1,2,\ldots,N_T-2^k+1 \qquad (14)$$

where $\mathbf{a}_{\mathrm{M}/k}$ and $\mathbf{a}_{\mathrm{m}/k}$ are the k-th rows taken from the matrices $\mathbf{A}_{\mathrm{M}}$ and $\mathbf{A}_{\mathrm{m}}$ respectively.

The first row (k=1) of matrices $\mathbf{A}_{\mathrm{M}}$ and $\mathbf{A}_{\mathrm{m}}$ is obtained directly by the TE sequence vector $\mathbf{x}$ as

$$\begin{aligned}a_{\mathrm{M}/1,i} &= \max(x_i, x_{i+1})\\ a_{\mathrm{m}/1,i} &= \min(x_i, x_{i+1})\end{aligned} \qquad (15)$$

for i=1, 2,..., N_T-1. The next rows (k>1), instead, are obtained recursively as

$$\begin{aligned}a_{\mathrm{M}/k,i} &= \max(a_{k-1,i}, a_{k-1,i+p})\\ a_{\mathrm{m}/k,i} &= \min(a_{k-1,i}, a_{k-1,i+p})\end{aligned} \qquad (16)$$

where $p=2^{k-1}$, for i=1, 2,..., N_T-2^k+1.

Finally, the value MTIE(t,T) for $t=(N_t-1)t_0$ and $N_t=2^k$ (here denoted as MTIE_k for the sake of brevity) can be evaluated from the k-th rows of the matrices $\mathbf{A}_{\mathrm{M}}$ and $\mathbf{A}_{\mathrm{m}}$ as

$$\mathrm{MTIE}_k = \max_{i=1,\ldots,N_T-2^k+1}\left(a_{\mathrm{M}/k,i} - a_{\mathrm{m}/k,i}\right) \qquad (17).$$

5.3 Computational Saving

The number of operations involved in the estimator plain computation and that in the binary-decomposition algorithm have been evaluated, in order to assess the resulting computational saving. It can be shown [19] that:

- the MTIE plain computation involves a number of operations (mainly comparison-test branches and variable assignments) in the order of N^2, if we limit MTIE computation to one value per octave on the τ axis as in the binary-decomposition algorithm (if MTIE is computed for all the possible N-1 values of τ, then the number of operations required becomes proportional to N^3 instead);
- the number of operations (again, mainly comparison-test branches and variable assignments) needed by the binary-decomposition algorithm is reduced to be proportional to $N\log_2 N$.

Finally, it may be interesting to know how long do both algorithms take to execute with practical values of N on some average/low-power computer. To this purpose, both algorithms have been programmed in C language, compiled and run for testing on a SUN Sparc Server 10 under UNIX operating system (SunOS). The plain computation of the MTIE estimator, one value per octave, required about 1100 s of actual execution time on a sequence of N=65536 samples (practical TE sequences may be longer). The binary decomposition algorithm, on the same sample sequence, needed just something more than one second to complete execution.

6 Conclusions

In this paper, fast algorithms to compute the TVAR and MTIE standard estimators were provided. These algorithms prove effective in achieving a strong computational saving: MTIE algorithm reduces the number of operations needed to be proportional to $N\log_2 N$ instead of N^2. Both algorithms allow fast evaluation in most practical situations: even very long sequences of TE samples do not require more than few seconds of data processing for TVAR and MTIE computation. Therefore, they may be conveniently adopted by telecommunications engineers involved in time-domain measurement of clock stability.

References

[1] W. C. Lindsey, F. Ghazvinian, W. C. Hagmann and K. Dessouky, "Network synchronization", *Proceedings of the IEEE*, vol. 73, no. 10, Oct. 1985, pp. 1445-1467.

[2] P. Kartaschoff, "Synchronization in digital communications networks", *Proceedings of the IEEE*, vol. 79, no. 7, July 1991, pp. 1019-1028.

[3] J. C. Bellamy, "Digital network synchronization", *IEEE Communications Magazine*, vol. 33, no. 4, Apr. 1995, pp. 70-83.

[4] S. Bregni, "A historical perspective on network synchronization", *IEEE Communications Magazine*, vol. 36, no. 6, June 1998.

[5] ITU-T Recs. G.810 *Considerations on Timing and Synchronization Issues*, G.811 *Timing Requirements at the Outputs of Primary Reference Clocks Suitable for Plesiochronous Operation of International Digital Links*, G.812 *Timing Requirements at the Outputs of Slave Clocks Suitable for Plesiochronous Operation of International Digital Links*. Geneva: Blue Book, 1988.

[6] ANSI T1.101-1994 *Synchronization Interface Standard*.

[7] ITU-T Rec. G.810 *Definitions and Terminology for Synchronisation Networks*. Geneva: Aug. 1996.

[8] ITU-T Rec. G.811 *Timing Characteristics of Primary Reference Clocks*. Geneva: Sep. 1997.

[9] ITU-T Rec. G.812 *Timing Requirements of Slave Clocks Suitable for Use as Node Clocks in Synchronization Networks*. Geneva: June 1998.

[10] ITU-T Rec. G.813 *Timing Characteristics of SDH Equipment Slave Clocks (SEC)*. Geneva: Aug. 1996.

[11] ETSI Draft ETS 300 462 *Transmission and Multiplexing (TM); Generic Requirements for Synchronisation Networks* (Parts 1-6). 1997.

[12] S. Bregni, "Clock stability characterization and measurement in telecommunications", *IEEE Trans Instrum. Meas.*, vol. 46, no. 6, Dec. 1997.

[13] J. A. Barnes, A. R. Chi, L. S. Cutler, D. J. Healey, D. B. Leeson, T. E. McGunigal, J. A. Mullen Jr., W. L. Smith, R. L. Sydnor, R. F. C. Vessot and G. M. R. Winkler, "Characterization of frequency stability", *IEEE Trans. Instrum. Meas.*, vol. 20, no. 2, May 1971.

[14] J. Rutman and F. L. Walls, "Characterization of frequency stability in precision frequency sources", *Proceedings of the IEEE*, vol. 79, no. 7, pp. 952-960, July 1991.

[15] M. Carbonelli, D. De Seta and D. Perucchini, "Characterization of timing signals and clocks", *European Trans. on Telecommunications*, vol. 7, no. 1, Jan.-Feb. 1996.

[16] S. Bregni, "Measurement of Maximum Time Interval Error for telecommunications clock stability characterization", *IEEE Trans. Instrum. Meas.*, vol. 45, no. 5, Oct. 1996.

[17] S. Bregni and F. Setti, "Impact of the anti-aliasing pre-filtering on the measurement of Maximum Time Interval Error", *Proceedings of IEEE GLOBECOM '97*, Phoenix, AZ, USA, November 1997.

[18] S. Bregni, “Clocks in telecommunications”, in *Encyclopedia of Electrical and Electronics Engineering*, edited by John G. Webster. New York: J. Wiley and Sons, 1999.

[19] S. Bregni, S. Maccabruni, "Fast computation of Maximum Time Interval Error by binary decomposition", *IEEE Trans. Instrum. Meas.*, Vol. 49, No. 6, Dec. 2000.