Real-Time MTIE Assessment with Flexible Control of Computation Process

Andrzej Dobrogowski, Michal Kasznia
Chair of Telecommunication Systems and Optoelectronics
Poznan University of Technology
Poznan, Poland
dobrog@et.put.poznan.pl, mkasznia@et.put.poznan.pl

***Abstract*—In this paper some modifications of the methods enabling real-time assessment of Maximum Time Interval Error (MTIE) are proposed. These modifications allow to control the parameter's computation process. The results of computation experiments performed for different data sequences are presented.**

I. Introduction

Maximum Time Interval Error (MTIE) is used for describing some aspects of quality of telecommunication network timing signals [1, 2, 3]. The MTIE point estimate is commonly computed for a series of observation intervals using the sequence of time error samples measured at some network interface. The computation usually follows the time error measurement process. However, in order to reduce the evaluation time of the timing signal, we can compute the MTIE in the real time, during the time error measurement process. Some methods enabling real-time MTIE computation were developed and tested by the authors of this paper [5, 6].

In order to calculate the MTIE estimate simultaneously for several observation intervals in the real time, all necessary operations should be performed in the time period between two sampling instants, i.e. during the sampling interval. The methods proposed by the authors – DSDR method, combining direct search and sequential (for growing observation intervals) reduction of data, and EF method, consisting in independent (for each observation interval) search using shifts to the positions of extreme samples – brought very good results for short observation intervals or time error data representing random behavior (e.g. WPM phase noise). Unfortunately, for longer observation intervals and data series showing monotonic change of time error process (caused by the difference of frequencies of the compared oscillators), the time of calculation was longer than sampling interval considered [8, 9]. Because both methods are data dependent, such situations result from the accumulation of toilsome computational processes being the result of specific arrangement of data appearing at one sampling instant.

In this paper some solutions of this problem are proposed. Additional real-time procedure controlling the computation progress is introduced. As result, the computation for a few longer observation intervals (belonging to the considered set of intervals) may be suspended in order to reduce the number of operations carried out within one sampling interval. The analysis of suspended intervals may be performed later (for the successive sampling intervals), when smaller number of operations is executed, or at the latest, when the measurement is finished. Such flexible control of the real-time computation can protect the measurement and computation process against exceeding the limit of the computation time.

In the paper the results of experimental tests of the modifications proposed are presented. The calculations were performed for several different data sequences taken with sampling interval 1/30 s, which is often used in the telecommunication applications.

II. Maximum Time Interval Error computation

According to the international standards, the point estimate of the Maximum Time Interval Error is defined as the maximum peak-to-peak time error variation of a given timing signal, with respect to an ideal timing signal within a particular time period [1, 2, 3]. MTIE can be estimated from the formula

$$M\hat{T}IE(n\tau_0) = \max_{1\le k\le N-n}\left(\max_{k\le i\le k+n} x_i - \min_{k\le i\le k+n} x_i\right) \qquad (1)$$

where $\{x_i\}$ is a sequence of N samples of time error function $x(t)$ taken with sampling interval τ_0, $\tau=n\tau_0$ is an observation interval, and n can change from 1 till N–1 depending on the considered values of observation intervals.

This work was supported by the Ministry of Science and Higher Education in the frame of the project number N N517 1645 33.

A. Off-line computation methods

MTIE point estimate is usually computed off-line, after the time error measurement process is completed. Following directly the formula (1) in order to find the estimate of MTIE for the observation interval τ, all intervals having the width of τ, existing within the sequence of N time error samples, must be reviewed. The window having the width of $\tau=n\tau_0$ and including $n+1$ samples is set at the beginning of the data sequence $\{x_i\}$ and then it is shifted with the step of τ_0 (one sample step) along the items of the sequence. For each window's location the peak-to-peak value of time error in the window is found. The maximum peak-to-peak value found for all existing locations of the window is the value of MTIE(τ) estimate. The process of the window reviewing does not depend on the data value. The complexity of calculation grows with n and therefore the direct method applied for computation of the parameter for the series of observation intervals is really time-consuming. The idea of direct search (plain computation) of MTIE is presented in Fig. 1.

Several time effective methods of MTIE assessment were proposed in order to avoid the time-consuming plain computation [4, 5]. Some different mechanisms were applied to reduce the complexity of computation. In the process of the MTIE search using the extreme fix (EF) method some window's locations are excluded from inspection if the peak-to-peak value for each of these locations is not greater than the value found until now, or if a greater peak-to-peak value may be found for the successive window's locations [5]. The EF method is based on fixing the positions of minimum and maximum samples for a given window's location. The general rule is that the next window's location is originated at the first extreme previously found. The reviewing process is performed only for the new samples that come into the window's location. The review of the whole window at its new location is performed only in the case of one-sample shift (the first extreme is first sample in the window's location), when the new sample appearing in the window is not a new extreme. The idea of EF method is presented in Fig. 2.

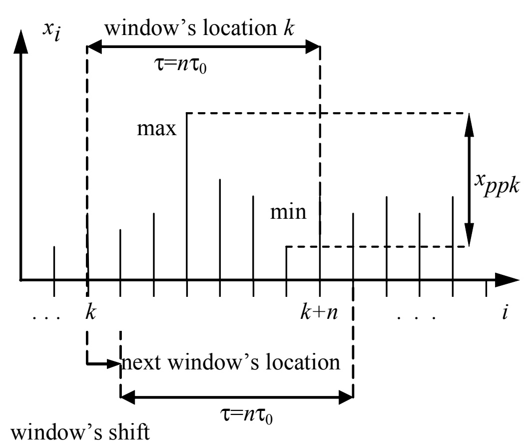

Figure 1. The idea of direct search for MTIE

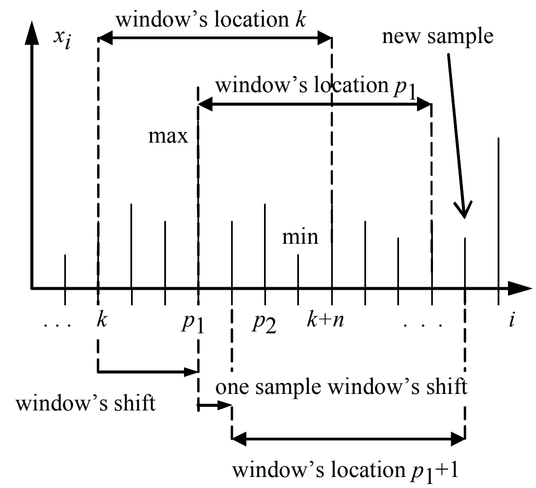

Figure 2. The window's shift in the extreme fix method

The methods with sequential data reduction were established for the MTIE assessment performed for a series of observation intervals, starting from some τ_{min} until some τ_{max}. The methods follow the suggestion according to which during the MTIE search process for some observation interval τ_i we find the extreme samples for some window's location from the set of extreme samples found previously during the MTIE search for the smaller observation interval τ_{i-1} ($\tau_i>\tau_{i-1}$). Therefore, we can reduce the number of time error samples used for the MTIE calculation. One of these methods, proposed by the authors of this paper – direct search with sequential data reduction (DSDR) – uses plain computation at each level of the procedure (for raw and reduced data) [7].

B. Real-time computation methods

The possibility of the MTIE computation performed in the real time during the time error measurement could be very useful in the process of evaluation of timing signal. We can observe the value of the parameter during the long lasting measurement process. Any possible wrong behavior of the analyzed signal (recognized if MTIE exceeds the limit) enables applying proper activity of a maintenance team. In order to calculate the MTIE estimate simultaneously for several observation intervals in the real time, all necessary operations should be performed in the time period between two sampling instants, i.e. during the sampling interval τ_0. The ability of real time assessment depends on the several conditions: number and length of the observation intervals considered, computational power of the measurement equipment, computation method applied, and time error data behavior.

Both time-effective methods described above, as well as the method with binary decomposition, proposed by Bregni and Maccabruni [4], were adopted for the real-time computation [7, 8, 9] by the authors of the paper. Only the principles of the real-time DSDR and EF methods will be presented below.

The real-time computation using DSDR method for the first (shortest) observation interval τ_{min} begins with the first measured time error sample. Each new sample measured is compared with current maximum and minimum values, until the first window's location is filled out by the samples. Then the extreme values for this location are fixed. Each successive measured sample creates a new window's location. The extreme samples found for each window's location create new data sequences with reduced number of items. The items of reduced data sequences are used for the MTIE estimate calculation for the observation intervals longer than τ_{min}. The first location of the next longer window is not analyzed until all samples situated in this location are reviewed by the preceding window.

The example of computation using DSDR method for observation intervals having 6, 8, and 10 samples is presented in Fig. 3 [8]. Fourteen samples were measured and only two windows are active now – 6-sample window and 8-sample window following the end of the smaller window. The 10-sample window is not active now, because it must wait for the review of samples by the preceding window.

In the case of real-time calculation using EF method, the computation procedures for each observation interval run independently. Window's locations of longer observations intervals are analyzed after filling out by the samples without waiting for the analysis by the preceding shorter windows. All windows are activated after filling out their first locations by the samples. The extremes found for some observation interval do not affect the calculation process for other observation intervals.

The example of real-time MTIE calculation – early stage of the process – for observation intervals having 6, 8, and 10 samples using EF method is presented in Fig. 4. Ten samples were measured and all windows are activated. The extreme samples (black and white stars) in the relevant window's locations are found and the next window's locations are set (dashed line). At the end of measurement, after analysis of the last window's locations, the parameter's values for each observation interval are known immediately, without any delay [8].

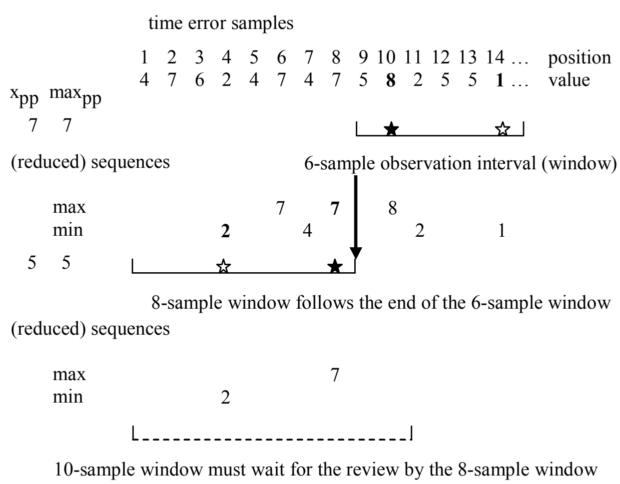

Figure 3. The idea of the real-time DSDR computation

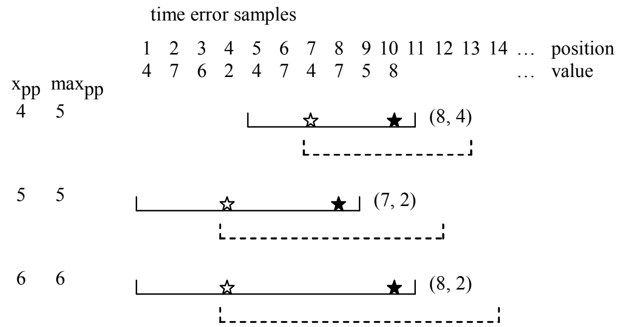

Figure 4. The idea of the real-time EF computation

III. Modifications of the Real-Time Computation Methods

The results of the experimental tests of real-time MTIE computation using DSDR, EF, and binary decomposition methods were presented in [8, 9]. The experiments were performed in order to verify the maximum time spent for necessary operations within one sampling interval. The results of the experiments have shown very good performance of the method with binary decomposition, but this method has one obvious disadvantage – the number of samples in the observation intervals considered must be a power of 2. The performance of two other methods – EF and DSDR – was good for limited range of observation intervals – small number of intervals having short length, e.g. for the range from 0.1 s till 10 s. The maximum computation time for longer observation intervals (for the range of 100 s – 1000 s) have exceed the length of considered sampling interval for the EF method. The DSDR method has proved good performance for the data series representing white phase noise, but the presence of monotonic changes in the data series (being the effect of the frequency difference), results in exceeding the established limit. Such situations result from the accumulation of computational procedures within one sampling interval caused by specific data arrangement. Therefore, if we want to avoid such troublesome situations during the measurement and computation process performed in the real time, some procedures controlling the computation progress should to be introduced. The goal of these procedures is to protect the computation process against exceeding the sampling interval. The modifications of the DSDR and EF methods are presented below.

A. Modification of DSDR method

The number of samples reviewed in course of the computation process performed within one sampling interval is tracked. If the number exceeds some threshold value, the analysis of the remaining (longer) observation intervals for this sampling instant is suspended. The analysis of the suspended intervals may be done during the computation process performed for the next sampling instant (if the threshold will be not exceeded again). As result, previously suspended windows will be delayed relatively to the shortest windows. At the end of the measurement, the part of the time error sequence, which remains for off-line analysis by the suspended windows will be longer than without suspension. The example of this procedure is presented in Fig. 5 and 6.

The sample no. 30 is measured. The analysis of 10-sample window (located between samples no. 6 and 15) is suspended for this sampling instant (Fig. 5). The analysis is continued (Fig. 6) for the sampling instant no. 31.

B. Modification of EF method

The number of samples reviewed in the course of the computation process performed within one sampling interval is tracked (as for the DSDR method). If the review of the whole window's location have to be performed for some observation interval (critical case for the EF method), and the number of reviewed samples (the number of reviewed samples until now and the length of the window considered) will exceed some threshold, then the computation for this observation interval is suspended until the end of measurement. The computation for the suspended observation intervals will be completed after the end of measurement. The example of this procedure is presented in Fig. 7 and 8. The extreme sample (8 at the position no. 10) leaves the current location of 10-sample window and the one-sample shift has to be done. Because the new sample coming to the window's location is not "more extreme" than the leaving sample – the review of the whole window at its new location is necessary. Therefore the analysis of the 10-sample window is suspended, because time of this review accumulated with the time of the operations done until now may exceed the length of sampling interval. The analysis of this window will be concluded after the end of time error measurement (Fig. 8).

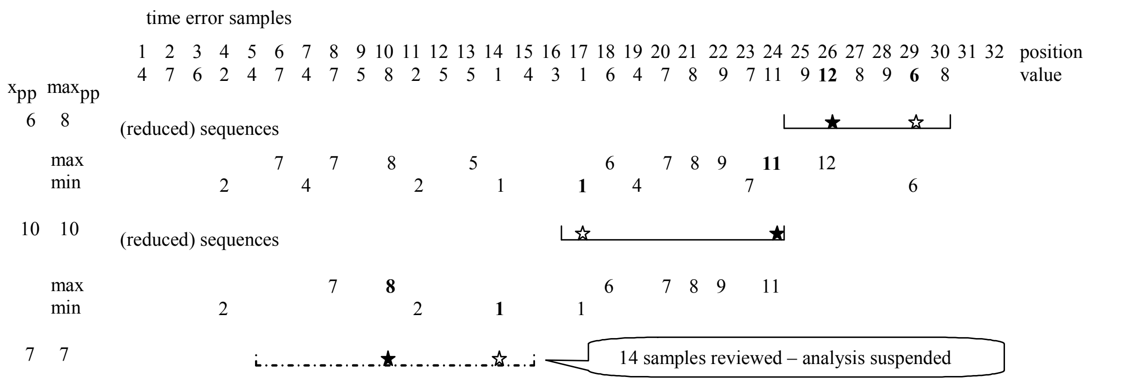

Figure 5. Suspension of the analysis for the real-time DSDR method

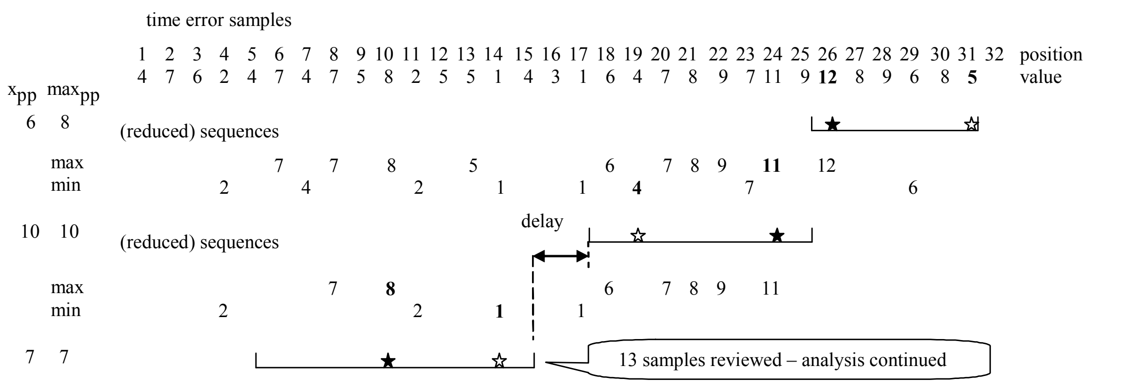

Figure 6. Continuation of the suspended analysis for the real-time DSDR method

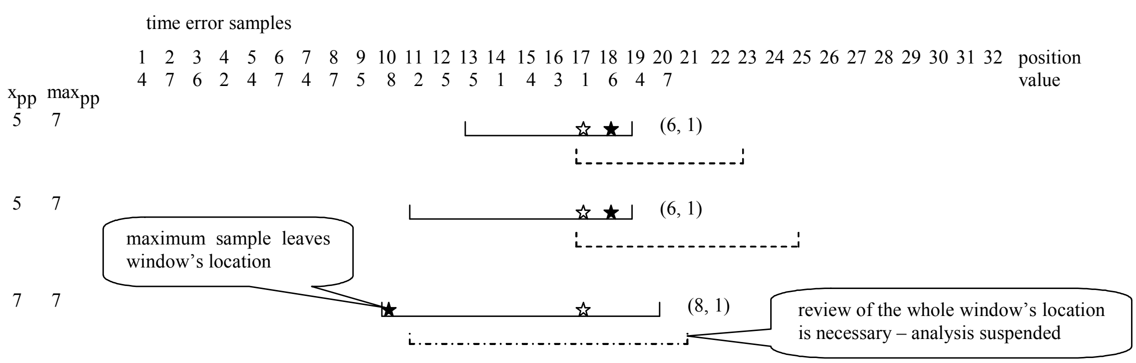

Figure 7. Suspension of the analysis for the real-time EF method

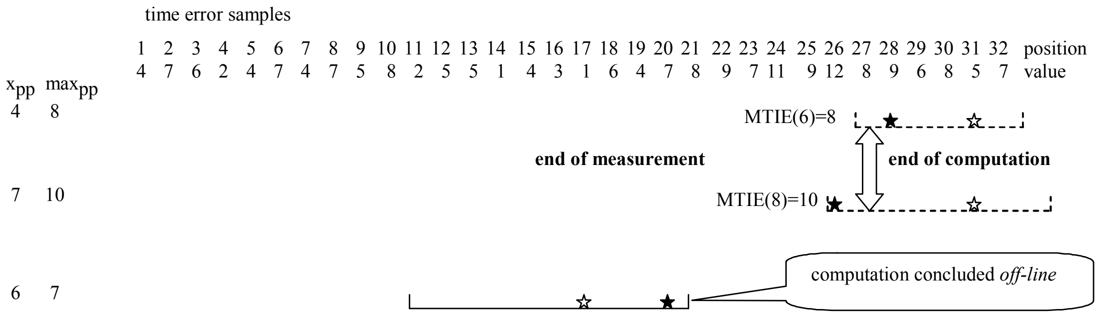

Figure 8. Continuation of the suspended analysis for the real-time EF method

The modifications of both methods consist in stopping the computation for some observation intervals considered. Therefore we may expect a delay in obtaining the final results of MTIE computation. Such delay is usual feature in the case of DSDR method – only the MTIE value for the first (smallest) observation interval is known immediately after the end of measurement; remaining windows are shifted off-line. In the case of EF method the delay will refer to few longest observation intervals. The conclusion of the computation may take some seconds.

IV. Experimental Test of the Modifications

The modifications of the real-time MTIE computation methods presented above were tested in the computational experiment. The experimental tests were performed according to the same rules as the experiments described in [8, 9] in order to compare the obtained results. The calculations were performed off-line but the on-line real-time work was imitated. Three different time error sequences were used: the first represents one of typical noises of the timing signals – white phase modulation (WPM); the second was the result of comparison of two different GPS disciplined oscillators; the third was the result of comparison of two independent internal oscillators of some measurement system (MSG). The time error samples were taken with the sampling interval τ_0=1/30 s.

Two different personal computers were used in the experimental tests: first computer with Pentium IV 3.0 GHz, and second one with Core 2 Quad 2.83 GHz. The observed quantity was the maximum time of calculation spent for sampling interval. We have assumed that this time cannot exceed the length of sampling interval τ_0=1/30 s = 33.3… ms.

The MTIE values were computed for the series of observation intervals arranged in the logarithmic scale in the range between 0.1 s and 100 s (16 intervals), and 0.1 s and 1000 s (21 intervals), starting from 0.1 s. These ranges of observation intervals were selected, because they give the critical results of the tests [8].

The computation using DSDR method with modification was performed for four threshold values (limit number of reviewed samples): 500, 200, 100, and 50. The time of MTIE computation using DSDR method with modification applied is presented in Table I (for the computer with Pentium IV) and in Table II (for the computer with Core 2 Quad). The results are compared with the time of computation without the control of the progress (without threshold), presented in the first two rows of the tables. Because good results were obtained for the WPM time error sequence without control (computation time does not exceed the length of sampling interval), this sequence was not considered in the calculation with threshold applied (progress control). The progress control with the threshold value of 50 samples only successfully protects the computation process against exceeding the 33.3 ms limit for the computer with older microprocessor (Pentium IV) and MSG data sequence for both ranges of observation intervals. Greater values of the threshold are not effective in the case of this sequence.

The computation using EF method with progress control was performed for four threshold values: 8000, 5000, 3500, and 2500. The time of MTIE computation using this method is presented in Table III (for the computer with Pentium IV) and Table IV (for the computer with Core 2 Quad). In the first rows of the tables the result of computation without progress control are presented. Time of computation without modifications using both computers exceeds the considered limit for each data sequence. Application of progress control with the 8000 samples threshold is successful for the computation using computer with modern processor (Core 2 Quad) for each sequence and both ranges of observation intervals. Therefore the tests for the thresholds of 3500 and 2500 samples were not performed for this computer. The computations using older processor need the threshold value of 5000 samples for WPM and GPS sequences and 2500 samples for MSG data sequence. Application of the progress control for wider range of observation intervals causes the suspension of computation for up to 6 observation intervals.

TABLE I. Time of calculation using DSDR method for computer with Pentium IV 3.0 GHz

Threshold	Range of intervals [s]	WPM	GPS	MSG
		t-max [ms]	*t-max [ms]*	*t-max [ms]*
none	0.1-100	5.5	7.7	24.7
	0.1-1000	7.1	11.0	54.9
500	0.1-100	-	7.7	24.7
	0.1-1000	-	11.0	54.9
200	0.1-100	-	6.1	24.7
	0.1-1000	-	8.3	40.1
100	0.1-100	-	6.1	24.7
	0.1-1000	-	6.6	39.5
50	0.1-100	-	3.9	4.4
	0.1-1000	-	3.9	4.4

TABLE II. TIME OF CALCULATION USING DSDR METHOD FOR COMPUTER WITH CORE 2 QUAD 2.83 GHZ

Threshold	**Range of intervals [s]**	**WPM** *t-max [ms]*	**GPS** *t-max [ms]*	**MSG** *t-max [ms]*
none	0.1-100	3.9	4.4	10.4
	0.1-1000	4.4	5.5	21.4
500	0.1-100	-	4.4	10.4
	0.1-1000	-	5.5	21.4
200	0.1-100	-	3.3	9.9
	0.1-1000	-	4.1	15.4
100	0.1-100	-	3.1	9.9
	0.1-1000	-	3.6	15.3
50	0.1-100	-	2.2	2.2
	0.1-1000	-	2.2	2.8

TABLE III. TIME OF CALCULATION USING EF METHOD FOR COMPUTER WITH PENTIUM IV 3.0 GHZ

Threshold	**Range of intervals [s]**	**WPM** *t-max [ms]*	**GPS** *t-max [ms]*	**MSG** *t-max [ms]*
none	0.1-100	16.5	20.9	36.4
	0.1-1000	141.2	145.5	180.2
8000	0.1-100	15.4	14.8	36.4
	0.1-1000	36.3	38.4	80.7
5000	0.1-100	15.4	14.8	34.6
	0.1-1000	23.6	23.1	54.4
3500	0.1-100	15.4	14.3	34.1
	0.1-1000	16.5	16.5	34.6
2500	0.1-100	9.9	9.9	22.0
	0.1-1000	9.9	11.5	22.0

TABLE IV. TIME OF CALCULATION USING EF METHOD FOR COMPUTER WITH CORE 2 QUAD 2.83 GHZ

Threshold	**Range of intervals [s]**	**WPM** *t-max [ms]*	**GPS** *t-max [ms]*	**MSG** *t-max [ms]*
none	0.1-100	5.5	5.5	12.6
	0.1-1000	49.5	50.6	58.3
8000	0.1-100	-	-	-
	0.1-1000	12.7	13.2	29.1
5000	0.1-100	-	-	-
	0.1-1000	8.3	8.3	19.2
3500	0.1-100	-	-	-
	0.1-1000	-	-	-
2500	0.1-100	-	-	-
	0.1-1000	-	-	-

V. CONCLUSIONS

Both methods proposed – EF and DSDR – allow to perform the MTIE computation in the real time, during the time error measurement. However the number and the range of observation intervals considered simultaneously is limited by the length of the sampling interval. In some cases the time spent for computation within one sampling interval may exceed this limit.

Application of the modifications proposed in this paper – control of computation progress with threshold – enables effective protection against exceeding the limit. The results of the experimental tests have proved the ability of the real-time MTIE computation for the range of observation intervals from 0.1 s till 1000 s performed during time error measurement with sampling interval τ_0=1/30 s.

REFERENCES

[1] ETSI EN 300 462, "Generic requirements for synchronization networks" (1998).

[2] ITU-T Rec. G.810, "Considerations on timing and synchronization issues" (08/96).

[3] ANSI T1.101-1999, Synchronization interface standard.

[4] S. Bregni, S. Maccabruni, "Fast computation of Maximum Time Interval Error by binary decomposition," IEEE Trans. Instrum. Meas., vol. 49, No. 6, pp. 1240-1244, Dec. 2000.

[5] A. Dobrogowski, M. Kasznia, "Time effective methods of calculation of Maximum Time Interval Error," IEEE Trans. Instrum. Meas., vol. 50, No. 3, pp. 732-741, June 2001.

[6] A. Dobrogowski, M. Kasznia, "Quasi-parallel computation of Maximum Time Interval Error estimates," Proc. of the 2002 IEEE Frequency Control Symposium, pp. 733-738, New Orleans, 29-31 May 2002.

[7] A. Dobrogowski, M. Kasznia, "Some problems and their solutions in quasi-parallel MTIE assessment," Proc. of the 2004 IEEE International Ultrasonics, Ferroelectrics, and Frequency Control Joint 50th Anniversary Conference, pp. 494-499, Montreal, 23-27 August 2004.

[8] A. Dobrogowski, M. Kasznia, "Testing of the methods of real-time MTIE calculation," Proc. of the 2005 Joint IEEE Frequency Control Symposium and Precise Time and Time Interval Systems Application Meeting, pp. 397-403, Vancouver, 29-31 August 2005.

[9] A. Dobrogowski, M. Kasznia, "On-line computation of MTIE using binary decomposition and direct search with sequential data reducing," Proc. of the 2007 IEEE International Frequency Control Symposium Jointly with the 21st European Frequency and Time Forum, pp. 877-882, Geneva, 29 May – 01 June 2007.