FACTA UNIVERSITATIS (NIŠ)
SER.: ELEC. ENERG. vol. 18, no. 3, December 2005, 421-429

The Error Estimate for the Modified Preset Count Ratemeter Algorithm

This paper is dedicated to Prof. M. Stojić on the occasion of his 65th birthday

Vojislav Aranđelović and Đorđe Šaponjić

Abstract: Analogously to the traditional preset count rate meter algorithm, the error estimate of the modified preset count ratemeter algorithm has been defined as the range around measured mean count rate containing 2/3 of the measurement results. The coefficients determining this range have been calculated for a set of values of the modification parameter (m) and preset count (N_T). The calculations cover the ranges of these parameters which are of practical interest ($m = 0.25 \div 1.5$, $N_T = 12 \div 150$). By applying the piecewise-linear interpolation scheme it is possible to calculate measurement error for any pair m, N_T within the ranges of these parameters covered by the present calculations.

Keywords: Error, estimate, ratemeter, algorithm.

1 Introduction

In an earlier article [1] a modified preset count ratemeter algorithm allowing an efficient limitation of statistical fluctuations has been presented. For the purpose of calculating mean count rate the modified algorithm restricts the values of statistical fluctuations of time intervals between successive input pulses to a range (t_a, t_b) around the average time interval $t_{av}(t_a < t_{av}, t_b > t_{av})$. Starting from an estimate of the mean count rate, it was shown that the ratio of the boundaries of the range (t_a, t_b) could be chosen so that despite the introduced modification of time intervals outside this range the correct mean value of the count rate would be obtained. The limitation of statistical fluctuations is significant because it allows reduction of the preset count, i.e. it results in an acceleration of the measurement process. Also, for

Manuscript received June 22, 2003.

V. Aranđelović and Đ. Šaponjić are with Institute of Nuclear Sciences VINČA, Electronics Department, 11 000 Belgrade, Serbia and Montenegro (e-mail: [avoja,djole]@vin.bg.ac.yu).

a selected preset count this limitation results in the corresponding improvement of the measurement accuracy compared to that of the traditional preset count ratemeter algorithm $\pm 1/\sqrt{preset\_count}$, such as e.g. moving average [2].

The modified preset count ratemeter algorithm is primarily intended for use at low count rates. The input pulses are then sparse and the time required to reach the selected preset count may become intolerably long. An example of such measurements is background radiation monitoring. A monitoring system applying the modified algorithm has been in operation in the Institute of Nuclear Sciences VINČA for over two years [3].

The purpose of this article is to present an assessment of the error of the modified preset count ratemeter algorithm. Namely, in the original article [1] no assessment of the measurement error was given. An a posteriori analysis of measured data was required in order to determine measurement error. The results presented here (Table 1 and the interpolation scheme) allow that the error estimate can be determined on the basis of the measured mean count rate and choice of the measurement parameters (preset count, t_a, and t_b). Alternatively, for an expected value of the mean count rate one can choose the measurement parameters to achieve desired accuracy.

2 The Basic Relation

It is well known that the distribution of time intervals between successive Poisson random events is described by [4]

$$I(t) = re^{-rt}. \tag{1}$$

In eq. (1) t denotes time interval between successive random events and r is the mean rate of the occurrence of these events.

Function $I(t)$ with the indicated boundary intervals t_a and t_b and the mean value between successive pulses t_{av}, i.e. the mean count rate is $r = 1/t_{av}$, is shown in Fig. 1.

The relation between t_a and t_b, which allows selection of t_b, given t_a, so that the correct value of r is obtained [1], is repeated here for convenience

$$t_b = \frac{1}{r} \ln \frac{1}{rt_a + e^{-rt_a} - 1}. \tag{2}$$

With a chosen value of t_a (the corresponding value of t_b is calculated by (2)) and a preassigned value of the preset count, N_T, it is possible to determine the numbers of time intervals within the characteristic ranges indicated in Fig.1: $(0, t_a)$, (t_a, t_b), and (t_b, ∞).

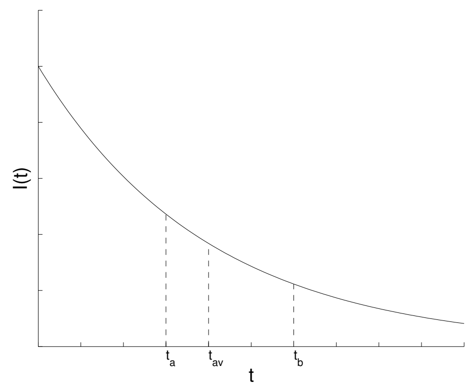


Fig. 1. The exponential character of the distribution of successive time intervals governed by Poissons distribution.

If the preset count is sufficiently large, the number of time intervals in $(0, t_a)$ is

$$n_a = N_T \int_0^{t_a} re^{-rt}\mathrm{d}t = N_T(1 - e^{-rt_a}), \tag{3}$$

in (t_a, t_b)

$$n_{ab} = N_T \int_{t_a}^{t_b} re^{-rt}\mathrm{d}t = N_T(e^{-rt_a} - e^{-rt_b}), \tag{4}$$

and in (t_b, ∞)

$$n_b = N_T \int_{t_b}^{\infty} re^{-rt}\mathrm{d}t = N_T e^{-rt_b}. \tag{5}$$

In the process of calculation of the current value of the mean count rate by the modified algorithm all time intervals within the range $(0, t_a)$ are replaced by t_a and all time intervals within the range (t_b, ∞) are replaced by t_b. Therefore the total time of N_T time intervals is

$$T = n_a t_a + n_b t_b + n_{ab} t, \quad t_a < t < t_b \tag{6}$$

where t is governed by distribution (1).

Since the measured mean count rate is given by

$$R = \frac{N_T}{T}, \tag{7}$$

the fluctuations of T will obviously cause the fluctuations of R, in accordance with (6) and (7). It is evident that the fluctuations of the results obtained by the modified algorithm will be considerably smaller than those of the traditional algorithm because (6) contains constant member $n_a t_a + n_b t_b$ and the variable member t in (6) can vary only within limited interval (t_a, t_b).

In this article, like in the original article [1], the parameter t_a is selected applying relation

$$t_a = t_{av}(1 - \frac{m}{\sqrt{N_T}}), \quad 0 < m < \sqrt{N_T} \tag{8}$$

where $t_{av} = 1/R$ is the average time interval between input pulses, $t_a = 1/R_{max}$ is the shortest time interval between input pulses acceptable for calculation of T in (6), and the modification parameter m relates t_a to the standard deviation of the traditional preset count ratemeter algorithm, $1/\sqrt{N_T}$. The longest time interval between input pulses acceptable for calculation of T in (6), t_b, is calculated by (2), i.e. $t_b = 1/R_{min}$.

Relation (8) implies that there is an expected value of the mean count rate $R = 1/t_{av}$. In the original article [1] it has been suggested that this value could be obtained by performing the initial measurement applying a traditional preset count algorithm. Each subsequent measurement step is performed by the modified algorithm. In each step the values t_a and t_b are redefined according to the current mean count rate value, and the algorithm converges towards the true mean value of the count rate.

3 The Error Criterion

From (6) one may conclude that the maximum value of R will be obtained if in all n_{ab} intervals t takes value t_a and the minimum value will be obtained if in all n_{ab} intervals t takes value t_b. Both of these two results are statistically most unlikely and the error estimate obtained on the basis of these considerations would be too conservative.

The approach carried out in this article starts from counting the number of measurement results contained in the range defined by the selection of m and N_T , i.e. by the selection of t_a and t_b. The bounds of this initial range of mean count rates are $R_{min} = 1/t_b$ and $R_{max} = 1/t_a$. This range usually contains all measuremet results. Then, by an iterative procedure the bounds of this range are narrowed by gradually increasing the initial lower bound (R_{min}) and decreasing the initial upper bound (R_{max}). The shrinking of this range results in decreasing the number of measurement results contained between the two bounds. The iterative procedure is run until 2/3 of the total number of measurement results are within the bounds. The position of this final range within the initial bounds (R_{min}, R_{max}) is such that the number of measurement results bellow the lower (R_{low}) and above the upper (R_{high}) bounds are approximately equal. The narrowed range (R_{low}, R_{high}) determined in this way corresponds to the error estimate of the modified algorithm. Therefore, the error criterion adopted for the modified preset count ratemeter algorithm is equivalent to that of the traditional preset count ratemeter algorithm.

The method is numerical and any combination m, N_T would require the corresponding calculation of R_{low} and R_{high}. For practical purposes, however, it would be sufficient to calculate R_{low} and R_{high} for a strategically chosen mesh of values of the modification parameter m and preset count N_T. Then, by applying a corresponding interpolation scheme one could calculate the error estimate for any combination of the modification parameter m and preset count N_T covered by the above mesh.

4 The Simulation Results

By using standard Random Number Generator for Windows Applications [5] and the corresponding original software written in C^{++} programming language, a large number of simulation runs, for a range of values of pairs m and N_T, has been carried out[1]. In these runs the iterative shifting of the bounds of the initial "window" ($R_{min} = 1/t_b, R_{max} = 1/t_a$) is completed when the narrowed "window" (R_{low}, R_{high}) containing 2/3 of the total number of measurement results is reached. The shifts of the bounds have been expressed by the corresponding factors: *lower compression factor* $k_{min} > 1$ for the lower bound of the mean count rate, $R_{low} = k_{min}R_{min}$, and *upper compression factor* $k_{max} < 1$ for the upper bound of the mean count rate, $R_{high} = k_{max}R_{max}$. Here-in-after, these factors will be referred to as the compression factors. In this way it was possible to express in a universal way the shrinking of the initial "window", irrespective of the value of the mean count rate.

Each simulation run comprised 500 individual results of counting the number of these results contained by the "window". The average value of these 500 individual results was taken as the number of results contained by the current value of the "window". If this number was higher than 334 (approximately two thirds of 500) the iterative procedure was continued until the average value of 10 successive simulation runs (5000 results) gave 334 results within the narrowed "window". In these 10 simulation runs no correction of the compression factors was applied. The criterion "2/3" was allowed to vary from 0.660 to 0.672, i.e. the results within the range 330-336 were acceptable. The tolerance of the numbers indicating the position of the narrowed "window" within the initial "window" was $\pm 5\%$.

The values of the compression factors corresponding to this sequence of 10 successive runs, satisfying the above tolerances, are taken for k_{min} and k_{max}. This process was done for each (N_T, m) pair indicated in Table 1. The corresponding

[1]Some of the runs have been repeated by using MATLAB [6] package. The expected good agreement of the corresponding results was obtained

calculations resulted in the values of the compression factors presented in Table 1.

The compression factors of Table 1 and a measured value of the mean count rate specify the error estimate, i.e. the narrowed ”window” (R_{low}, R_{high}), for the specified mesh of pairs (N_T, m).

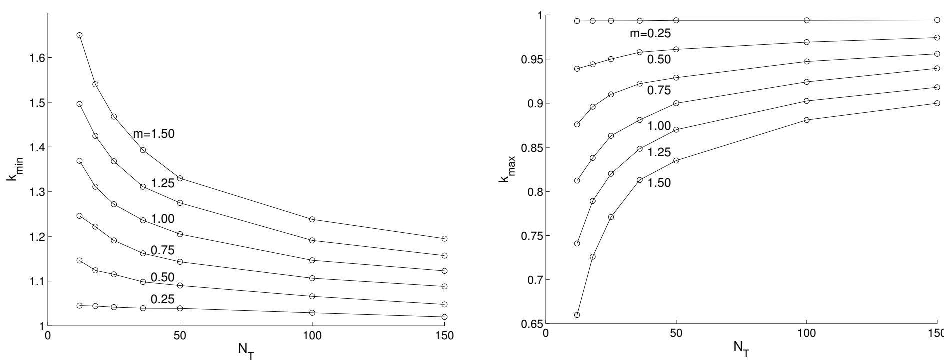
Table 1. The compression factors.

m	0.25	0.5	0.75	1.00	1.25	1.5
			$N_T = 12$			
k_{min}	1.0452	1.146	1.246	1.369	1.496	1.65
k_{max}	0.9932	0.939	0.876	0.8123	0.741	0.66
			$N_T = 18$			
k_{min}	1.04414	1.1240	1.2218	1.311	1.4249	1.54
k_{max}	0.99330	0.9440	0.896	0.838	0.7892	0.726
			$N_T = 25$			
k_{min}	1.0417	1.115	1.191	1.272	1.368	1.468
k_{max}	0.9933	0.950	0.910	0.863	0.820	0.771
			$N_T = 36$			
k_{min}	1.03941	1.0982	1.16214	1.236	1.311	1.393
k_{max}	0.9934	0.9578	0.922202	0.881	0.8484	0.813
			$N_T = 50$			
k_{min}	1.039	1.090	1.143	1.205	1.275	1.330
k_{max}	0.994	0.961	0.929	0.900	0.870	0.835
			$N_T = 100$			
k_{min}	1.0290	1.0658	1.1065	1.1466	1.1910	1.238
k_{max}	0.994	0.9692	0.9473	0.9242	0.9025	0.881
			$N_T = 150$			
k_{min}	1.01976	1.04786	1.088	1.1228	1.157	1.195
k_{max}	0.99432	0.97434	0.956	0.9395	0.918	0.90

It can be noticed that for a given preset count N_T the values of both compression factors approach unity as the value of the modification parameter m decreases. In other words, given an N_T, for small values of m the bounds determining the error of the modified algorithm approach the boundaries of the initial ”window”. On the other hand, as the value of m increases, the compression factors become increasingly different from unity, i.e. the error estimate becomes noticeably smaller compared to that corresponding to the boundaries of the initial ”window”.

Another characteristic of the data in Table 1 may be of interest. Namely for small values of m the corresponding distributions of the results become very narrow and sharp edged. In order to fulfill the criterion 2/3 of the measurement results the compression factors have to be calculated very precisely even though only the first two-three digits to the right of the decimal point essentially determine the error estimate. As N_T increases this property becomes even more pronounced. The compression coefficients presented in Table 1 have been calculated with the precision indicated in Table 1 and they are used as true values in the applied interpolation scheme. On the other hand, for small values of N_T and/or large values of m, the precision of the compression factors calculations need not be very high because the edges of the corresponding distribution of the measurement results are now relatively slowly varying.

For the purpose of determining the dependencies of the values of the compression factors on N_T, for a selected set of modification parameter values m, the simplest, piecewise-linear approximation has been used. The illustrations of these dependencies are shown in Figs. 2(a) and 2(b). The N_T values in Table 1 have been



(a) (b)

Fig. 2. (a) The lower compression factor as function of preset count N_T for the indicated set of modification parameters m. (b) The upper compression factor as function of preset count N_T for the indicated set of modification parameters m.

selected bearing in mind that the application of the modified algorithm invariably leads to smaller values of N_T compared to those of the traditional preset count algorithm, given a measurement accuracy. For this reason values like $N_T = 12$, 18 or 25 are far more likely to be used than 100 or 150. Therefore, the density of the mesh of the selected (m, N_T) pairs is much higher for small values of N_T. By applying the above linear interpolation scheme the error estimate for any N_T from 12 to 150 has been calculated for the presented set of values of m.

The data presented in Table 1 can also be used to determine the dependencies of the values of the compression factors on m, for the selected set of values of N_T. Again applying the piecewiselinear approximation these dependencies are illustrated in Figs. 3(a) and 3(b). By combining the results presented in Figs. 2 and 3 it is possible to calculate the error estimate of the modified preset count ratemeter algorithm for any combination (m, N_T) covered by Table 1 as soon as the mean count rate is measured.

From the data obtained by combining Table 1 and the corresponding interpola-

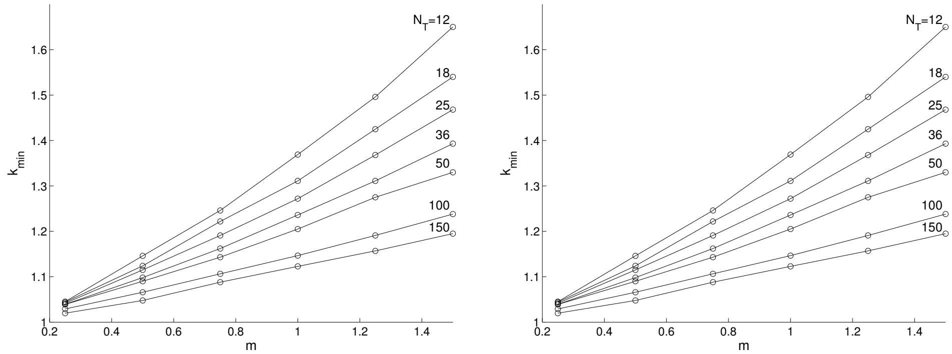


Fig. 3. (a) The lower compression factor as function of the modification parameter *m* for the indicated set of preset counts N_T. (b) The upper compression factor as function of the modification parameter *m* for the indicated set of preset counts N_T.

tion schemes for any m, N_T pair one can calculate the lower and upper bounds of the narrowed "window" of the mean count rate containing 2/3 of the measurement results, $R_{low} = k_{min}/t_b$ and $R_{high} = k_{max}/t_a$. The corresponding measurement error is now calculated as the pair of fractional standard deviations $-\Delta R/R = (R - R_{low})/R$ and $+\Delta R/R = (R_{high} - R)/R$, where R is the measured mean count rate.

5 Conclusions

The error estimate of the modified preset count ratemeter algorithm is defined as the range of count rates around the measured count rate containing 2/3 of the measurement results. This definition is fully equivalent to the error of the traditional preset count ratemeter algorithm, $\pm 1/\sqrt{presetcount}$ since it covers the range of mean count rates containing 2/3 of the measurement results.

By using data presented in this article an experimenter, after selecting the factor of limitation of statistical fluctuations, i.e. modification parameter *m*, and preset count N_T, may calculate the corresponding error estimate as soon as the mean count rate is measured. The analysis of measured data by the experimenter in order to establish measurement accuracy is no longer required.

Acknowledgement

The cooperation of prof. Milić Stojić with the Electronics Department of the Institute of Nuclear Sciences VINČA is longstanding. It began in the late seventies through realization of a joint project concerning application of electronic methods in the areas of measurement and control. The cooperation, often involving both

criticism and encouragement, still lasts. By this small contribution we would like to thank prof. Stojić for many years of joint efforts and a lasting friendship.

References

[1] V. Aranđelović, A. Koturović, and R. Vukanović, “A software method for suppressing statistical fluctuations in preset count digital-rate meter algorithms,” *IEEE Trans. on Nuclear Science*, vol. 49, no. 5, pp. 2561–2566, 2002.

[2] C. H. Vincent, *Random Pulse Trains*. London: IEE Peter Peregrinus, 1973, ch. 3.

[3] Šaponjić, A. Žigić, and V. Aranđelović, “HYPERION a distributed measurement system for monitoring background ionizing radiation,” *Nuclear technology & Radiation protection*, vol. XVIII, no. 1, pp. 42–46, 2003.

[4] G. F. Knoll, *Radiation Detection and Measurement*. New York: Wiley, 2000, p. 97.

[5] *Random Number Generator for Windows Applications*, Sunny Beach Technology, Inc, Minneapolis, MN.

[6] *MATLAB version 6.5, Release 13*, July 2002.