EFFICIENT FREQUENCY JUMPS DETECTION ALGORITHM FOR ATOMIC CLOCK COMPARISONS

Michał Marszalec[1], Marzenna Lusawa[1], Tomasz Osuch[1,2]

1) *National Institute of Telecommunications, Szachowa 1, 94-894 Warsaw, Poland* (✉ *m.marszalec@itl.waw.pl*, +48 22 5128 140, *m.lusawa@itl.waw.pl*)

2) *Warsaw University of Technology, Faculty of Electronics and Information Technology, Institute of Electronic Systems, Nowowiejska 15/19, 00-665 Warsaw, Poland* (*t.osuch@elka.pw.edu.pl*)

Abstract

In this paper a new method of frequency jumps detection in data from atomic clock comparisons is proposed. The presented approach is based on histogram analysis for different time intervals averaging phasetime data recorded over a certain period of time. Our method allows identification of multiple frequency jumps for long data series as well to almost real-time jump detection in combination with advanced filtering. Several methods of preliminary data processing have been tested (simple averaging, moving average and Vondrak filtration), to achieve flexibility in adjusting the algorithm parameters for current needs which is the key to its use in determining ensemble time scale or to control systems of physical time scales, such as UTC(PL). The best results have been achieved with the Vondrak filter.

Keywords: time interval, phasetime, frequency jump, ensemble time scale, atomic clock.

1. Introduction

In most time metrology laboratories equipped with several atomic clocks or having access to a precise method of time transfer from other laboratories, it is important to supervise and control a local time scale based on data from comparisons of available 1PPS time signals (*1 Pulse Per Second*) from the individual clocks available. These comparisons can be shown as series of phasetime results recorded in equal intervals of time τ_0. One of the methods to determine a set of parameters for precise local time scale control is based on direct determination of frequency offset differences between individual clocks. In the next step, comparison results are used for extrapolation of behavior of local reference in order to determine the prediction of its correction. Another method is to use the Kalman filter [1]. If several clocks comparison results are available, the most precise method turns out to be the calculation of the ensemble time scale. Ensemble time scale is a weighted average of corrected indications of a set of atomic clocks according to

the applied ensemble time scale algorithm where weights depend on the behavior of clocks over a set interval of time [2].

Unfortunately, ensemble time scales are very sensitive to any fluctuations in data series. Therefore, original data must be pre-processed before starting calculations. Another important issue is that the more accurate corrections are needed, the more accurate filtration system should be used. A typical, simplest and effective way to reduce the impact of phase, frequency and drift fluctuations in the input data on the operation of ensemble time scale algorithms is to increase the number of clocks. This method is used by the International Bureau of Weights and Measures (BIPM), where ensemble time scale is based on c.a. 350 atomic clocks. In this case, if any jump occurs, the algorithm lowers the weight of the unstable clock, or even removes it. This approach is not suitable for local national ensemble time scales with 20 or lower number of clocks which are used for steering UTC(k) realizations (where k stands for laboratory/country code *e.g.* PL). From this point of view another important issue is that original BIPM calculations (TAI/UTC) are delivered very rarely, *i.e.* in a monthly period with a two-week delay after the last day of a month [1]. The second BIPM scale (UTC Rapid), available for several years, is calculated weekly with a delay of 3 days. Typically, this scale differs from UTC by about 3 ns and can be very useful, but a faster available, independent, and self-controlled reference source is still needed.

Some of the reasons for the deterioration of accuracy of ensemble time scales consisting of a relatively small number of clocks are phase and frequency jumps in results of individual clocks comparisons (often referred to as phasetime) [3, 4]. The phase jumps are mainly due to temporary disturbances that originate in the generator element, the signal path, and also in the measurement system. They are usually the result of temporary changes in the operating point of individual electronic circuits caused, for example, by external electromagnetic pulses that induce accidental changes of electrical potentials in the system [4]. Other reasons could be the short-term mismatch or the interruption in the electrical signal path [4, 5]. In a measurement system, such abrupt changes often cause temporary changes in the trigger level that result in distortions in the recorded data. In the case of frequency jumps, the source of interference may be variations in the magnetic field [4, 5] as well as persistent or progressive fluctuations of the ambient temperature of the compared atomic clocks with the measurement systems [5, 6]. Badly configured or damaged components of the measurement system can also result in the frequency offset changes.

Due to different nature of phase and frequency jumps, phase jumps have to be detected and removed using separate specially designed sub-algorithms according to methods presented in [5, 7]. Detection and removal of frequency jumps are performed using methods originally developed for the analysis of financial markets (MAV, CUMSUM) [5, 7] or using the Kalman filter [8, 9]. MAV, CUMSUM use a sliding window approach which can create problems with implementation if the noise characteristics of the atomic clock change. Kalman filter-based methods are also sensitive to variations in the characteristics of the analyzed process, therefore controlling their operational parameters requires complicated implementation [10]. Additionally, the implementation may differ depending on the type of the anticipated changes [11, 12].

In this paper we propose a frequency jumps detection method that is based on automated histogram analysis. The presented approach, specially developed for the needs of atomic clock, is free from disadvantages of the above mentioned methods and allows precise control of the algorithm parameters. The principle of the algorithm that describes individual steps, data preparation, preliminary analysis, determination of histogram modal parameters and scope of applications is presented in details. All algorithm steps have originally been implemented and tested. In addition, the presented algorithm has been tested with weighted moving average (with different weighting curves) and the Vondrak filter to minimize the problems associated with reducing the number of measurement points when averaging measurement results. Details on the use of the Vondrak filter

for atomic clock comparisons are presented in [13], basic mathematical formulas are included in [14] and extended explanations are shown in [15, 16].

2. Basic description of the frequency jump detection algorithm

The proposed algorithm for the detection and elimination of frequency jumps can be generally presented as a sequence of successive steps shown in Fig. 1. The algorithm enables quantitative analysis of data, and thus complete automation of their processing. Depending on the selected averaging time, it is possible to precisely define the frequency jumps detection threshold.
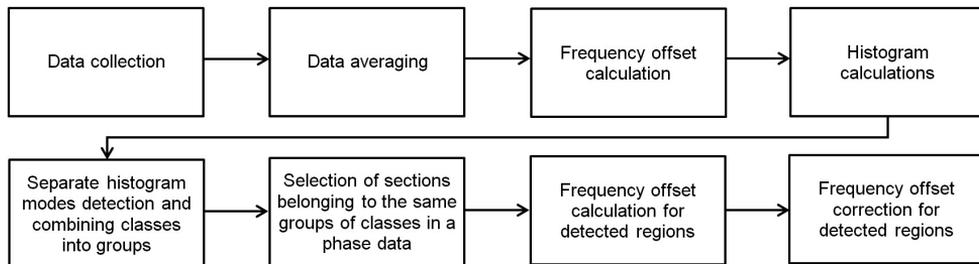

Fig. 1. Complete scheme of frequency jumps detection and removal algorithm.

Section 2.1 contains details on data collection used for further analysis. Data averaging and fractional frequency offset calculation are presented in Section 2.2. Section 2.3 explains how to determine the histogram for presented algorithm. The next Sections 2.4–2.7 cover the last four steps of the scheme depicted in Fig. 1.

2.1. Data collection (Step 1)

The result of direct comparisons of two time atomic clocks is usually shown as phasetime comparisons in time interval τ_0 which can be denoted as $\mathbf{X} = x_{0,1}, \ldots, x_{0,N}$, where N is number of all data points. The $x_{0,N}$ denotes raw data phasetime points before averaging, while data points after averaging are depicted as x_N. Thus "0" subscript denotes raw data. The example of the calculated phasetime comparisons of data recorded over almost three weeks with marked phase and frequency jumps is shown in Fig. 2a.

In the time domain, phase and frequency jumps can be easily distinguished from each other (Fig. 2a). The phase jump in the graph showing the compared waveforms of rectangular/pulse signals is characterized by a single change in the difference in the time interval that persists for subsequent periods of both signals. In turn, the frequency jump manifests itself as a constant change in the time interval for all subsequent periods of the signal waveform.

Usually, the results of phasetime measurements contain both phase and frequency jumps. Their occurrence depends on the class of compared clocks, the time/frequency transfer method that is used and the measurement systems that are responsible for the final comparison of two reference signals. Another problem is caused by discontinuities in the data series or gaps, where the best practice is to fill them in with interpolation data before further processing [5].

Description and operation principle of the proposed detection and elimination of frequency jumps algorithm is presented on the example of the phasetime measurement data of 1PPS signals.
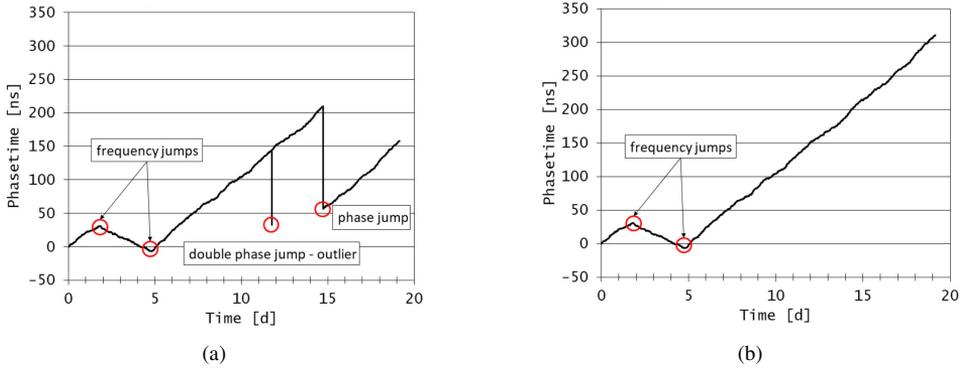
Fig. 2. Illustration of phase and frequency jumps in the phasetime diagram – the real comparison results of 1PPS signals between two cesium atomic clocks (a) raw data, (b) raw data after phase jumps removing.

Data points have been recorded in time interval $\tau_0 = 300$ s. The raw data of phasetime comparisons after phase jumps removing are shown in Fig. 2b.

After two days of measurements one of clocks suddenly has changed its frequency offset by approx. 3.7E–13. This change lasted for about three days and then the frequency returned to the previous value. As it is noticeable in the phasetime diagram, the frequency jump has a form of disturbance in the monotonicity of the waveform. This fluctuation could be caused by change in the magnetic field (one of the atomic clocks was restarted after being repaired) or by a temperature jump in the room where the clocks are located.

2.2. Raw data averaging and frequency offsets calculation (Step 2 and step 3)

This chapter presents an analysis of the use of data averaging for frequency jump detection purposes. This process should be carried out to obtain a sufficiently smooth frequency curve for the next step of the described algorithm. As a result of the averaging process, from raw data in the τ_0 interval we receive data in the τ interval:

$$\tau = k \cdot \tau_0, \tag{1}$$

where k is a consecutive positive integer. Then, frequency offsets y_i i.e. the differences between adjacent values (averaged from initial samples) are calculated, according to the following equation:

$$y_i = \frac{\dfrac{\sum\limits_{j=i\cdot k+1}^{(i+1)\cdot k} x_{0,j}}{k} - \dfrac{\sum\limits_{j=(i-1)\cdot k+1}^{i\cdot k} x_{0,j}}{k}}{k \cdot \tau_0} = \frac{x_{i+1} - x_i}{\tau}, \tag{2}$$

where x_i, $x_i + 1$ are phasetime adjacent values, τ denotes the time interval between adjacent values (e.g. τ_0). Then vector of frequency offset values y_i can be denoted as $\mathbf{Y} = y_1, \ldots, y_{Ny}$, where Ny is the reduced number of data points due to calculation of every frequency offset value from two adjacent phasetime samples. Formally, the sign before (2) should be negative, but for convenience it has been presented in accordance with the ITU Recommendation [17] as for frequency departure definition. Consequently, a positive value is adopted when the phasetime

diagram grows in the drawing, and drops otherwise. This behavior is very helpful in further analysis.

The transition to the frequency domain is aimed at selection of the histogram for the examined data set and detection of separate groups of classes. The frequency offset plot for $\tau = \tau_0$ (300 s in used data set) is shown in Fig. 3a. In this case, due to the too short averaging time τ, the data are so noisy that the frequency jumps are not noticeable. Furthermore, the Gaussian shape of the calculated histogram of the frequency offset confirms the dominant character of the white noise in the analyzed data (Fig. 3b).

(a) $\tau_0 = 300$ s
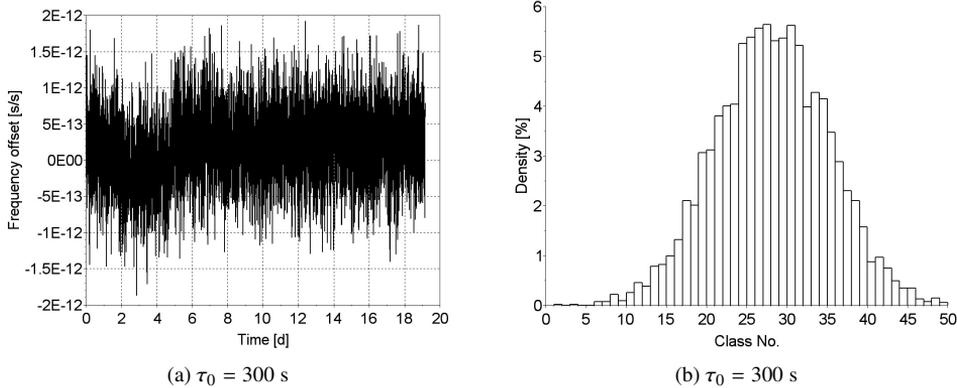
(b) $\tau_0 = 300$ s

Fig. 3. (a) Frequency offsets calculated from raw data $\tau = \tau_0$, and (b) their histogram.

However, when increasing the averaging time, frequency offset smoothing is observed due to the reduction of frequency white noise, and finally the frequency jump becomes more and more visible (Fig. 4). Unfortunately, with the increasing the averaging time, the number of frequency offset samples is reduced to *Ny* according to the following equation:

$$N_y = N_x - 1 = \left\lfloor \frac{N}{k} \right\rfloor - 1. \tag{3}$$

(a) $\tau_{avg} = 2$ h
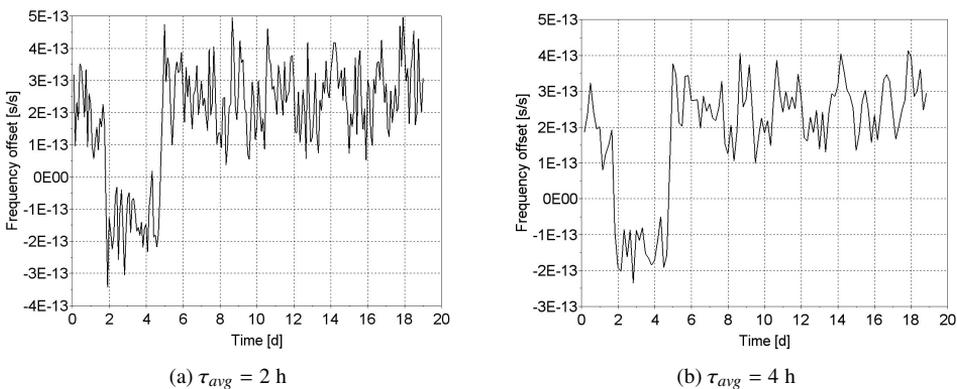
(b) $\tau_{avg} = 4$ h

Fig. 4. Calculated frequency offsets for the comparison results of 1PPS signals between two cesium atomic clocks for different averaging times $\tau = k \cdot \tau_0$, where $\tau_0 = 300$ s and (a) $k = 24$, (b) $k = 48$.

This reduction in the number of samples is a significant disadvantage in further data processing *i.e.* histogram calculation. Methods that avoid this drawback are presented in Sections 3.1–3.2.

2.3. Histogram calculations (Step 4)

In the next step of the algorithm, basing on the frequency offset data, a histogram is calculated. In the absence of a frequency jump, the frequency offset samples should oscillate around a fixed value resulting from the difference in the frequency of the particular clock. In this case, the histogram assumes the Gaussian shape according to the nature of noise of the atomic clock and the influence of the measurement system on clock comparison. This is in accordance with the usual behavior where an atomic clock in the free run mode has a normal distribution with a well-defined single maximum corresponding to the frequency difference of clocks. In turn, when at least one frequency jump appears in the analyzed data, then histogram tends to deviate from the Gaussian profile. However, as it was previously mentioned, if the averaging time is too short, for example when $\tau = \tau_0$, the amplitude of frequency offset noise is significant and thus the frequency jump is hardly observed (as shown in Figs. 3a and 3b).

With the increase of the averaging time τ, the histogram becomes more and more modally separable, and thus the frequency jumps are increasingly noticeable (Fig. 5). An optimal τ can be determined from Allan deviation stability characteristics. It is possible to make at least an initial selection of τ from the atomic clock specification provided by the manufacturer of the clock. After collecting a sufficient number of data points, an experimental stability analysis should be conducted for a clean fragment of raw data to determine the actual stability characteristics. Such an analysis should be repeated regularly to detect any possible deterioration in the stability of one of the tested clocks. If such changes are detected, the clock should be excluded from use.
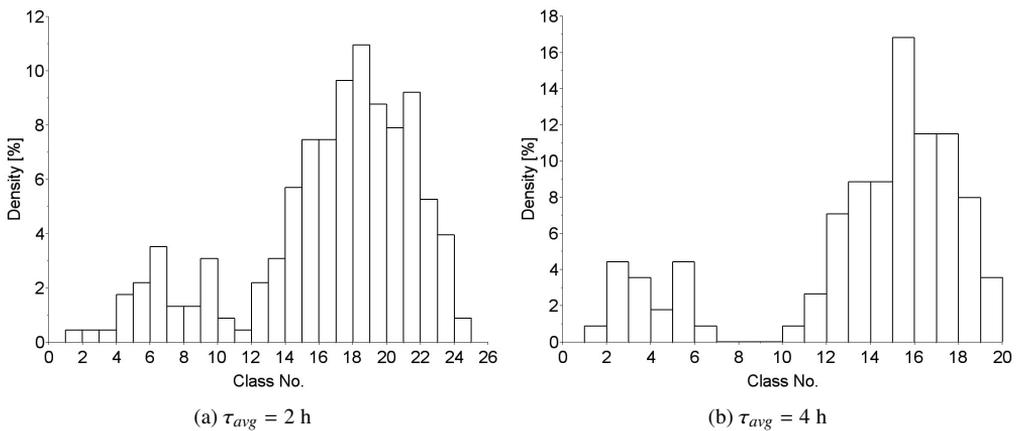

(a) $\tau_{avg} = 2$ h

(b) $\tau_{avg} = 4$ h

Fig. 5. Histograms for comparison results of 1PPS signals between two cesium atomic clocks, for different averaging times $\tau = k \cdot \tau_0$, where $\tau_0 = 300$ s and (a) $k = 24$, (b) $k = 48$.

Basing on histograms calculated for y obtained in previous step, it can be concluded that an optimal value of the averaging time is $\tau = 4$ h (Fig. 4b) because two modal groups are well separated. Although for longer averaging times the separation between modal groups is even better, the number of samples decreases rapidly which greatly reduces the number of possible histogram classes. Finally, it complicates the implementation of a fully automatic algorithm.

Another important issue in the proposed algorithm is determination of number of histogram classes. There are two methods that are often used for this purpose. According to the first one, the number of classes can be derived as a square root of the total number of samples. It is also possible to use the Huntsberger's formula where the optimal number of class intervals l for N_y data samples can be calculated as $l = 1 + 3.3 \log N_y$ [18]. It reveals that if the excessive reduction of the number of samples due to the averaging process implies a decrease in histogram classes, then the number of histogram classes should be determined experimentally.

2.4. Separate histogram mode detection (Step 5)

The next step of the algorithm is to determine frequency offsets that correspond to boundary values of every existing group of classes (also called bins) in the histogram. Then the occurrence of the first and last frequency offset values within the frequency offset range determined by the boundary values should be found. All data points between them should be averaged to calculate mean frequency offset for a particular group of classes. This procedure needs to be repeated for every separated group of frequency offsets.

The best situation occurs when all existing groups of classes in the multimodal distribution are completely separated. Then, there are empty gaps between neighboring groups with a width of at least one class (as it is shown for bimodal histogram shown in Fig. 5b when the averaging time is significantly increased). In this case, the average value of the frequency offset and group of classes limits can be directly determined. Unfortunately, in order to avoid excessive reduction in the number of input points, it may be necessary to reduce the averaging time (τ). As a result, the groups of classes in the histogram may not be completely separated. To solve this problem, we propose to introduce a minimum class size limit (the threshold value corresponding to the minimum occurrence of data points y in a given class) which should be slightly bigger than the least numerous class between detected modes, as shown in Fig. 6a. Then, classes that are below the threshold value can be removed. As a result, a complete separation of existing groups of classes is obtained. However, the phasetime data points corresponding to the removed classes and to the discontinuity regions in phasetime diagram) should be replaced with extrapolated values.
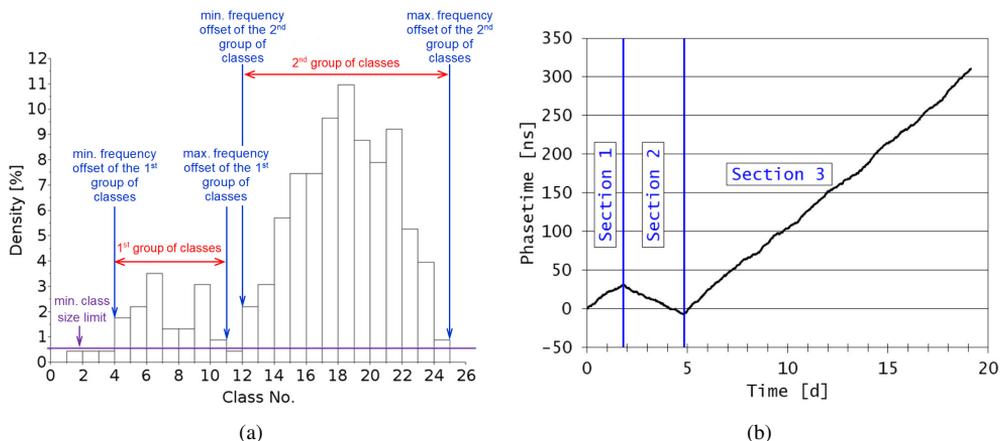

Fig. 6. Illustration of (a) combining histogram classes into groups, and (b) corresponding selection of sections belonging to the same group of classes in a phasetime diagram.

113

2.5. Group of classes section selection (Step 6)

With obtained group of classes limits, several scans of the phasetime data **X** should be performed. First, the review of **X** data is needed to assign each value to the appropriate group of classes. Then, the next step is to determine the minimum and maximum limits of phasetime data **X**, belonging to individual groups of classes and thus designate separate phasetime data **X** sections (Fig. 6b). It can be safely assumed that the sections of frequency offset data **Y** are the same as for phasetime data **X**. Then the data points in regions between designated sections should be marked as transient sections.

2.6. Frequency offset calculation for data regions (Step 7)

The next step is to determine the value of the frequency jump for each of the data sections, which can be obtained by linear approximation or to determine the average value of all frequency data points *y* in a given section. This method allows one to find multiple frequency offset changes even of a very similar value. Detected frequency jump values can now be used to correct phasetime data **X**.

2.7. Frequency offset correction (Step 8)

Then, depending on the needs, one of two strategies of insertion of calculated frequency jumps into phasetime data can be adopted. The forward strategy is used for "real-time" detection, calculation and alerting of appearing frequency jumps. The second, backward strategy is useful when an analysis tailored to the final, recently obtained value of the comparison results **X** is needed, as in the case of ensemble time scale calculation.

Another important issue is to recognize which clock has jumped when the frequency jump is observed. The best way is the three-cornered hat method (as in the case of instability detection presented in [19]), or at least a comparison of each individual clock with two others. This allows the source of the jump to be clearly identified.

3. Alternative averaging and filtration methods implemented in the frequency jump detection algorithm

Although the increase of an averaging time results in better separation of individual histogram modes, two important issues need to be taken into account. First of all, as mentioned before, simple averaging causes serious problems associated with reducing the number of measurement points, especially for high τ values. Secondly, it also decreases the number of classes of the histogram which poses difficulties in modes separation. The problem becomes especially troublesome in the case of frequency jumps of intermediate values compared to the strongest, which causes the occurrence of the additional sub-modes in the histogram. To overcome the first drawback of data reduction, the simple method of averaging should be replaced with an algorithm which can control this effect. The first solution seems to be the use of the weighted averaging algorithm with a sliding window (weighted moving average) [2,7]. Another way is to use more an advanced filtration algorithm, such as the Vondrak filter [13]. Both methods are often recommended for handling time series data in scientific papers (also in the above-mentioned ones). They were selected as the most popular and the most promising averaging and filtration methods which are used in atomic clock data analysis. The other problem can be solved by shortening the observation window. It, in turn, limits the number of histogram modes (best balanced in terms

of the number of samples). To sum up, three most commonly used pre-processing data methods have been adopted in this algorithm for testing: simple averaging (the most fundamental method of averaging), weighted moving average (sample reduction effect correction) and the Vondrak filter (allows optimization of smoothness and fidelity of data).

3.1. Weighted moving average

As mentioned above, to overcome the problem of data reduction a weighted moving average algorithm has been used. Unfortunately, the use of a moving average causes a statistical dependence of successive data points. To resolve this issue, several weighting curves have been selected and then tested. New data points x_i have been calculated using three different curves of weighting functions which are calculated according to shapes: $-|i|$ (linear – LWA), $(-i)^2$ (quadratic – QWA) and $(\sin i)$ (sinusoidal – SinWA) presented in Fig. 7a.

(a)

(b) $\tau_0 = 300$ s, $\tau_{avg} = 4$ h

(c) $\tau_0 = 300$ s, $\tau_{avg} = 4$ h
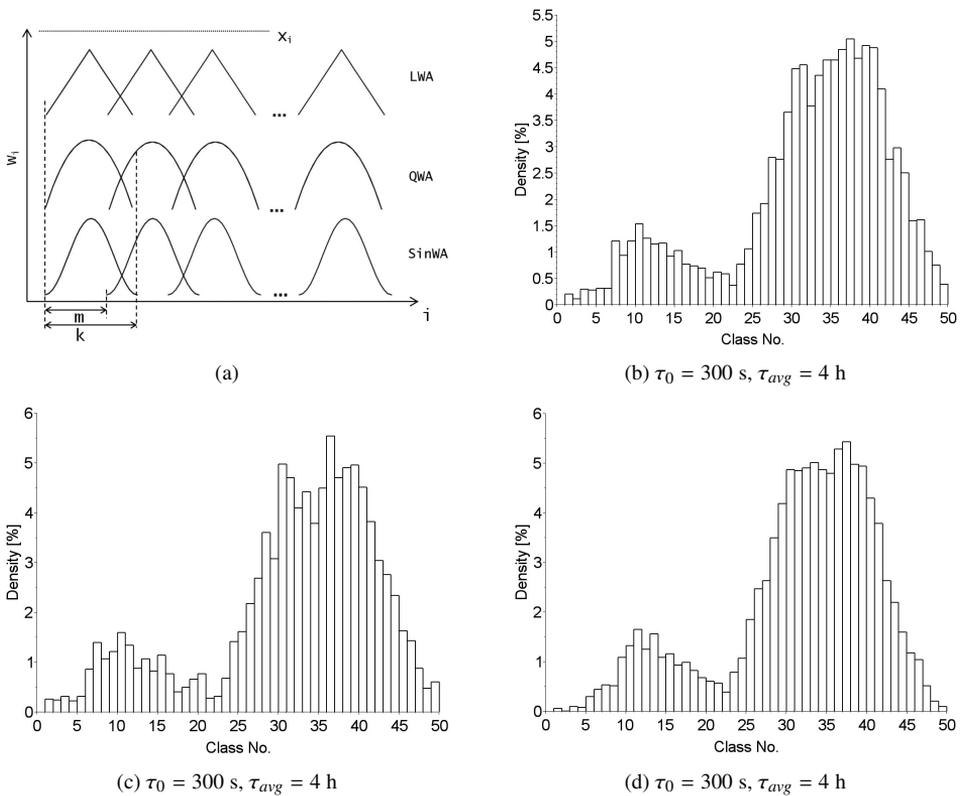
(d) $\tau_0 = 300$ s, $\tau_{avg} = 4$ h

Fig. 7. (a) Tested curves used for weighting averaging calculations and calculated frequency offsets for averaging with different weighting curves and corresponding histograms, (b) LWA, (c) QWA, (d) SinWA.

The output phasetime data x_i are averaged in a sliding window starting in every $m \cdot \tau_0$ point where m is a consecutive integer which determines the window shift and thus number of output samples. When $m = k$, the number of output samples is the same as in the simple averaging approach. For $m = 1$, the amount of data available for calculating the histogram is only slightly reduced ($N_x = N - k + 1$) or even does not change (with respect to raw data) if averaging at the

edges of the data series is calculated with the use of a reduced window length. For $m = 1$ the equation of determining the weighted average takes the following form:

$$x_{avg,i} = \frac{\sum_{j=i}^{k+i} \omega_j x_j}{\sum_{j=1}^{k} \omega_j}, \tag{4}$$

where ω_j represents weights appropriate for each tested curve.

An algorithm has been tested for $k = 49$ ($\tau \approx 4$ h) and $m = 1$ with different weighting curves. Histograms for calculated frequency offsets are shown in Fig. 7.

The use of the weighted average approach allows for an analysis of long series of data. Unfortunately, even if it does not reduce the number of samples, it causes worse separation of groups of classes (higher histogram minimum values) and makes configuration of entire algorithm harder especially for shorter window of observations. Worse separation means that below the minimum class size limit there will appear not only classes belonging to the transition regions, but also residual data inside the data sections already identified (with a specified frequency offset). The excessive number of such artifacts turns out to be very difficult to control, which causes the detection of non-existent frequency offset jumps.

3.2. Vondrak filtering

An alternative approach that has been investigated is the Vondrak filtering technique. The Vondrak filter is an algorithm that allows to meet two conditions simultaneously: smoothness of the filtered data (S) and their closeness to the original one (F) [14–16]. To fulfill these requirements the following Q function should be minimized:

$$Q = F + \varepsilon \cdot S, \tag{5}$$

where ε denotes constant determining the degree of smoothness and:

$$F = (N - 3) \sum_{i=1}^{N} q_i \left(x_i - x_i' \right)^2, \tag{6}$$

$$S = (t_N - t_1)^{-1} \int_{t_i}^{t_N} \left[\frac{d^3 \varphi(t)}{dt^3} \right]^2 dt, \tag{7}$$

where N stands for the total number of phasetime samples, x_i is a phasetime data point and x_i' is the expected solution of the system for this data point, while φ is defined as a third order Lagrange polynomial calculated of the group of four following data points x_i in τ_0 interval. Regularized value q_i of weight p_i can be calculated as inversely proportional to the root of standard deviation σ_i of x_i phasetime data [13]:

$$q_i = p_i (2k + 1)^{-1} \sum_{j=i-k}^{i+k} p_j, \tag{8}$$

$$p_i = \frac{1}{\sqrt{\sigma}_i} \, . \tag{9}$$

The value of ε is determined experimentally by direct comparison of results obtained for simple averaging and Vondrak filtering, selected for exactly the same moments. As a second method, for further analysis, a closeup observation of the frequency offset change region in relation to the original samples can be used. The influence of ε value on filtration efficiency is presented for the phasetime section in the vicinity of the frequency jump. Fig. 8a shows a closeup of raw phasetime data and (Vondrak-) filtered data for several different ε lying between $1E-10$ and $5E-7$.
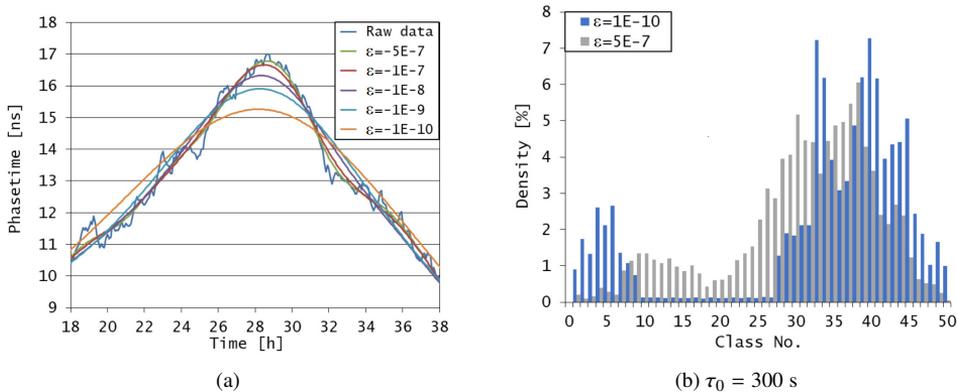

Fig. 8. (a) Closeup on raw and filtered data calculated in a two-day period near the first frequency jump for different ε constant, (b) Vondrak-filtered data histograms for $\varepsilon = 1E-10$ (blue) and $\varepsilon = 5E-7$ (gray).

It can be noticed that for $\varepsilon = 5E-7$ the curve fits well, but unfortunately, the corresponding histogram shows insufficient mode separation for automatic detection of a frequency jump (Fig. 8b). On the other hand, the constant $\varepsilon = 1E-10$ caused significant short-term differences in relation to the original values. The value $\varepsilon = 1E-8$ has been adopted as a reasonable compromise that allows for fully automatic operation.

The results of using Vondrak filtration approach are shown in Fig. 9. It can be observed that the smoothness of the frequency offset diagram (for $\tau_0 = 300$ s) shown in Fig. 9a is similar to that obtained for $\tau = 4$ h when simple averaging was used (Fig. 4b). In Fig. 9b, two groups of classes in the histogram are very well separated, comparable to the case of simple averaging with $\tau = 4$ h (Fig. 5b), while the number of classes remains unchanged (is not reduced). As compared to the weighted moving average approach, Vondrak filtering provides a much better separation of groups of classes in the histogram.

It reveals that the main advantage of using the Vondrak filter is the possibility of significant shortening of the observation period without negative effects on histogram calculations.

The other important advantage is the possibility of obtaining optimal smoothness (with properly chosen ε). As a result, it minimizes the noise of the calculated fractional frequency offsets and leads to minimal noise of frequency offset data points as well as the best mode separation in the histogram. Thanks to this, it becomes possible to detect frequency jumps with a delay of individual days (or less) or to detect much smaller jumps (short observational window).

(a) $\tau_0 = 300$ s, $\varepsilon = 1E-8$ (b) $\tau_0 = 300$ s, $\varepsilon = 1E-8$
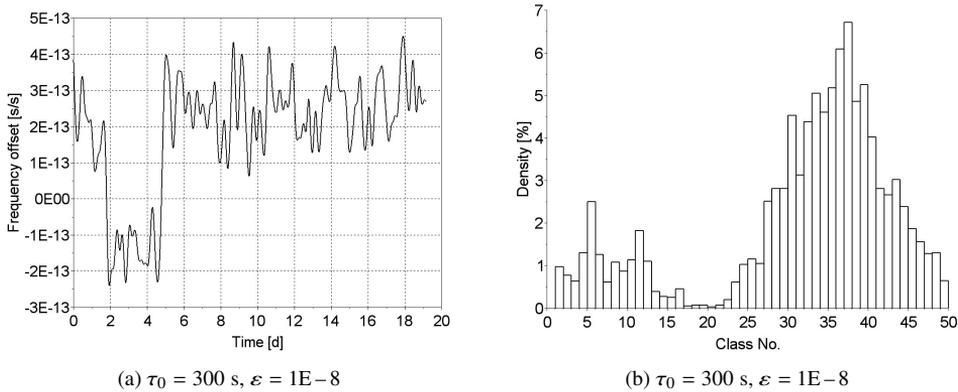
Fig. 9. Calculated (a) frequency offsets for Vondrak-filtered data and (b) corresponding histogram ($\varepsilon = 1E-8$).

3.3. Final results

To examine the impact of various data filtration methods on the operation of the frequency jump detection algorithm, the raw phasetime data were averaged using the following approaches: simple averaging for $\tau = 2$ h (avg2h) and $\tau = 4$ h (avg4h), weighted moving averages (linear – LWA, quadratic – QWA, sinusoidal – SinWA) and Vondrak filtration. After calculating histograms with a separate group of classes, the frequency jumps corresponding to every section in the phasetime graph were designated for various filtration methods and are presented in Table 1. Basing on this data the frequency jumps were removed from the phasetime diagram, as it is shown in Fig. 10a.

Table 1. Calculated frequency offsets values corresponding to every section for various filtering methods.

Filtration method	Section 1 frequency offset	Section 2 frequency offset	Section 3 frequency offset
avg2h	1.958E–13	–1.400E–13	2.486E–13
avg4h	1.988E–13	–1.400E–13	2.483E–13
LWA	1.906E–13	–1.394E–13	2.489E–13
QWA	1.907E–13	–1.394E–13	2.489E–13
SinWA	1.899E–13	–1.395E–13	2.490E–13
Vondrak	1.954E–13	–1.396E–13	2.491E–13

The data presented in Table 1 for the particular section of the phasetime diagram are close to each other. Thus, it can be stated that all tested averaging methods work correctly and give consistent results. It is also confirmed by the efficient linearization of phasetime diagrams after removing the frequency jumps. A small deviation in phasetime diagrams is observed for the simple averaging method (avg2h and avg4h), but it results from the significant reduction in the number of samples in filtered data sets. This discrepancy is even more noticeable when phasetime diagrams are additionally detrended (Fig. 10b).

In order to quantify the efficiency of the filtering methods, the stability analysis ADEV (Allan deviation) was carried out for both raw and averaged data for various averaging times τ. The results, collected in Table 2, clearly show that for relatively short (300 s and 900 s) and medium
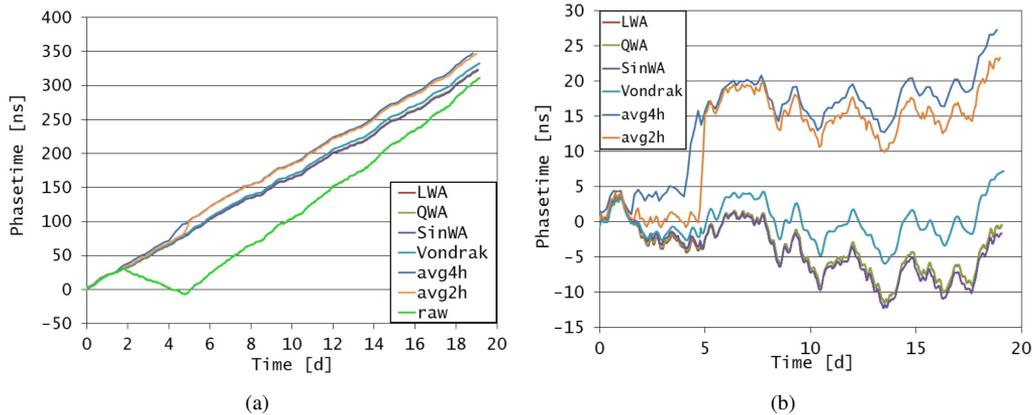
(a)   (b)

Fig. 10. Comparison of different filtration methods used for frequency jumps detection (a) raw data and data series with removed frequency jumps, (b) data series with removed frequency jumps and detrended.

(1–4 h) averaging times ADEV is the lowest when Vondrak filtering is applied. For longer τ, all weighted moving averaging algorithms behave as well as Vondrak filtering. Moreover, the discrepancy between simple averaging and other filtering techniques becomes higher and higher, due to the fact that for avg2h and avg4h methods, ADEV rises when τ increases. To sum up, among the averaging methods, Vondrak filtration appears to be the best solution, especially when rapid frequency detection and correction of frequency jumps is crucial.

Table 2. ADEV results for raw data and data sets after filtration.

Filtration method / τ	300 s	900 s	1 h	2 h	4 h	8 h	12 h	24 h
Raw data	4.9E–13	3.1E–13	1.7E–13	1.2E–13	8.9E–14	7.8E–14	7.5E–14	8.5E–14
avg2h	–	–	–	8.4E–14	7.9E–14	7.5E–14	7.4E–14	8.5E–14
avg4h	–	–	–	–	6.5E–14	7.1E–14	7.2E–14	8.4E–14
LWA	7.8E–15	1.6E–14	4.8E–14	6.7E–14	6.6E–14	5.6E–14	5.1E–14	3.9E–14
QWA	7.6E–15	1.6E–14	4.4E–14	6.4E–14	6.5E–14	5.5E–14	5.1E–14	3.9E–14
SinWA	1.6E–14	2.5E–14	4.8E–14	6.2E–14	6.4E–14	5.5E–14	5.1E–14	3.9E–14
Vondrak	6.5E–15	1.1E–14	2.4E–14	3.7E–14	5.4E–14	5.5E–14	5.1E–14	3.9E–14

4. Conclusions

In this paper we present an efficient algorithm for automatic detection and removal of frequency jumps in datasets from atomic clock comparisons. The algorithm is useful in preparation of data for calculating ensemble time scales and developing stable physical time scale realization. It is particularly well suited for ensemble time scales with a small number of atomic clocks due to protection against exclusion of clocks that temporarily behave incorrectly. Several methods of averaging and filtration of input data have been tested and their impact on the operation of the entire algorithm has been determined. It shows that for large data sets, simple

averaging is enough, but with the limited number of data it is necessary to use more advanced filtration methods. The algorithm can be used for quick detection of jumps while performing atomic clock comparisons by reducing the observation interval. It is also suitable for long-term analyses due to its ability to detect multiple jumps even with similar frequency values. After testing various filtration techniques, it is revealed that for relatively short averaging times Vondrak filtering is the best solution when rapid frequency jumps detection and correction is needed.

Further research on the use of a Kalman filter for optimal phasetime white noise filtering or implementation of the kernel function instead of the histogram analysis can be carried out.

Acknowledgements

References

[1] Davis, J. A., Shemar, S. L., & Whibberley, P. B. (2011). A Kalman filter UTC(k) prediction and steering algorithm. *Joint Conference of the IEEE International Frequency Control and the European Frequency and Time Forum (FCS) Proceedings*, USA. https://doi.org/10.1109/FCS.2011.5977793

[2] Thomas, C., Wolf, P., & Tavella, P. (1994). *Time Scales*, BIPM.

[3] Riley, W. J. (2013). *Outliers in Time and Frequency Measurements*. Hamilton Technical Services.

[4] Ruusunen, M., Paavola, M., Pirttimaat, M., & Leiviskai, K. (2005). Comparison of three change detection algorithms for an electronics manufacturing process. *IEEE International Symposium on Computational Intelligence in Robotics and Automation*, Finland, 679–683. https://doi.org/10.1109/CIRA.2005.1554355

[5] Riley, W. J. (2006). *Gaps, Outliers, Dead Time, and Uneven Spacing in Frequency Stability Data*. Hamilton Technical Services.

[6] Marszalec, M., & Lusawa, M. (2014). Research on impact of the environmental factors on National Institute of Telecommunications time standards stability. *Proceedings of SPIE 9290*, Poland. https://doi.org/10.1117/12.2076058

[7] Riley, W. J. (2010). Frequency Jump Detection and Analysis. *Proceedings of 40th Annual Precise Time and Time Interval (PTTI) Meeting*, USA, 241–253.

[8] Nunzi, E., Galleani, L., Tavella, P., & Carbone, P. (2007). Detection of Anomalies in the Behavior of Atomic Clocks. *IEEE Transactions on Instrumentation and Measurement*, 56(2), 523–528. https://doi.org/10.1109/TIM.2007.891118

[9] Galleani, L., & Tavella, P. (2010). An algorithm for the detection of frequency jumps in space clocks. *Proceedings of 42nd Annual Precise Time and Time Interval (PTTI) Meeting*, USA, 503–508.

[10] Galleani, L., & Tavella, P. (2012). Detection of atomic clock frequency jumps with the Kalman filter. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 59(3), 503–508. https://doi.org/10.1109/TUFFC.2012.2221

[11] Huang, X., Gong, H., & Ou, G. (2014). Detection of weak frequency jumps for GNSS onboard clocks. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 61(5), 747–755. https://doi.org/10.1109/TUFFC.2014.2967

[12] Galleani, L., & Tavella, P. (2017). Robust Detection of Fast and Slow Frequency Jumps of Atomic Clocks. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, *64*(2), 475–485. https://doi.org/10.1109/TUFFC.2016.2625311

[13] Feissel, M., & Lewandowski, W. (1984). A comparative analysis of Vondrak and Gaussian smoothing techniques. *Journal of Geodesy*, *58*(4), 464–474.

[14] Vondrák, J. (1969). A contribution to the problem of smoothing observational data. *Bulletin of the Astronomical Institute of Czechoslovakia*, *20*, 349–359.

[15] Vondrák, J. (1977). Problem of smoothing observational data II. *Bulletin of the Astronomical Institute of Czechoslovakia*, *28*, 84–89.

[16] Štefka, V., & Pesek, I. (2007). Implementation of the Vondrak's smoothing in the combination of results of different space geodesy techniques. *Acta Geodynamica et Geomaterialia*, *4*(4), 129–132.

[17] International Telecommunication Union. (2017). *Measures for random instabilities in frequency and time (phase)* (Recommendation ITU-R TF.538-4). https://www.itu.int/dms_pubrec/itu-r/rec/tf/R-REC-TF.538-4-201707-I!!PDF-E.pdf

[18] Gardiner, V., & Gardiner, G. (1979). Analysis of Frequency Distributions. *Concepts and Techniques in Modern Geography*, *19*.

[19] Premoli, A., & Tavella, P. (1993). A revisited three-cornered hat method for estimating frequency standard instability. *IEEE Transactions on Instrumentation and Measurement*, *42*(1), 7–13. https://doi.org/10.1109/19.206671

Michał Marszalec is a head of the Time and Frequency Metrology Team in the Laboratory of Electrical, Electronic & Optoelectronic Metrology in the National Institute of Telecommunications. He received his M.Sc. in Telecommunications from the Faculty of Electronics and Information Technology of the Warsaw University of Technology (WUT) in 2000. He specializes in time and frequency metrology, measuring techniques and timescale ensemble algorithm development in the Database for Polish Atomic Timescale TA(PL). He is currently pursuing the Ph.D. degree at the Institute of Electronic Systems, WUT.

Tomasz Osuch received his Ph.D. from the National Institute of Telecommunications (NIT) in 2010 and D.Sc. from the Warsaw University of Technology (WUT) in 2017. He is currently Associate Professor and Head of the Optoelectronic Metrology Group at the NIT and head of the Fiber Optic Sensors and Measurement Systems Group at the WUT. He has authored or co-authored over 100 journal papers and conference contributions. His research activity focuses on modelling, technology and interdisciplinary applications of fiber optic sensors (especially based on fiber Bragg gratings) and measurement systems as well as fundamental metrology.

Marzenna Lusawa is the main specialist in the Time and Frequency Metrology Team in the Laboratory of Electrical, Electronic and Optoelectronic Metrology in the National Institute of Telecommunications. She received her M.Sc. from the Faculty of Physics of the University of Warsaw in 2008. She specializes in time and frequency metrology, measuring techniques and in timescale ensemble algorithm development in the Database for the Polish Atomic Timescale TA(PL).