MATEC Web of Conferences **79**, 01076 (2016) DOI: 10.1051/matecconf/20167901076
IME&T 2016

A COMPARISON OF DATA COMPRESSION METHODS FOR SOLVING PROBLEMS OF TEMPERATURE MONITORING

Hussein Mogahed[1], *Alexey* Yakunin[a2], *Larisa* Suchkova[2]

[1]Minia University, 61111, Minia, Egypt
[2]Altai State Technical University, 656038, Barnaul, Russia

Abstract. The paper considers the problem of large processing huge amounts of data for temperature monitoring of man-made and natural objects associated with the lack of data compression efficiency in real time when they are transferred and stored in the presence of anomalies in the information signal in the form of sudden changes and outlier. The solutions of existing methods were described and new approaches were proposed. The results of experimental comparison of proposed and known solutions are included.

1 General definition of the problem

Nowadays, remote control and monitoring systems are widely used in modern technologies of automated production and scientific research [1].

One of the main demands in the monitoring systems, especially, for environmental monitoring and energy accounting is the ability to reproduce the recorded data at sufficiently large time intervals, up to several years. In some cases, the dynamics of registered processes require that the sampling period does not exceed 5–30 seconds. Considering that the number of different types of stored data may reach tens or even hundreds and requires huge amount of storing volume.

Therefore, in most technological and environmental monitoring systems, all data are usually exposed to preliminary on-line processing, one of its objectives is converting the original data in order to minimize the amount of physical memory required to store them. To make such changes in the monitoring system, it requires a dedicated storage subsystem. This subsystem must guarantee storing the source data with sufficient accuracy for later reconstruction, and a partial or full treatment to simplify and improve the efficiency of subsequent sampling and analysis of the monitoring results. This is necessary for using the stored data to address new problems related to the study of the controlled objects, also, to use this data for further improvement of processing algorithms.

One of the major requirements for storage systems is to minimize the total volume of stored data, which is achieved by using various methods of compression. Another, no less

[a] Corresponding author: yakunin@agtu.secna.ru

important requirement, is to minimize the time access to the data. These requirements are particularly relevant when the amount of stored data becomes too large. For example, in studies of atmospheric turbulence, the temperature data sampling time can be 5–30s at a resolution of about 0.05 ° C and a number of sensors 10 or more [2]. Almost the same time and amplitude resolutions are needed to control the emergency situations in residential and office buildings, however the total number of temperature sensors in this case can reach several hundred [3–7]. It is easy to verify that the uncompressed storage in this case may require about 315 MB per month (100 sensor * 8 bytes (time) * 2 bytes (measured value) * 365 days * 24 hours * 120 sample / hour for one sensor at 30 – second sampling rate). As far as, the number of sensors in complex systems can be significantly bigger, and the storage period may be several years, it is clear that to solve the problem of fast compression and decompression of large data sets in real-time, the traditional approaches, for example in [8–17], may be not helpful. This is explained by the fact that purely statistical methods are universal, as a rule, the block processing of the input data, whereas the dictionary methods of compression are based on the adaptation of some models without taking into account the real properties of the recorded data stream.

From this it follows that it is the most reasonable to add the known universal methods of compression with specially developed methods of stream processing and transformation of the data allowing to carry out most effectively preliminary compression of information. As it will be shown further, such decision allows refusing the subsequent application of the algorithms of compression using resource-intensive statistical strategy. This is especially relevant in implementing the monitoring systems based on the Internet - technologies and microcontrollers with low power consumption and, therefore, low data processing capacity.

Another important issue relating to storing data is optimization of data structure and extent of their preliminary processing. Indeed, the demand for various kinds of data is determined by their relevance and pragmatic value, which in turn depends on the nature of the problem to be solved by an information system. Therefore, in some cases it is advisable to keep the same data with different degree of post-processing and therefore detailed. A typical example of this approach is widely used in SCADA – systems.

Nature of preliminary data processing in many respects is defined also by their original quality. Thus, in some cases, may require compensation of systematic errors due to the inaccuracy of the sensor calibration and the effect on it of various influencing variables.

To eliminate the influence of random errors due to internal noise and external equipment, as well as to minimize the dynamic distortion of the signal caused by the inertia of the primary device, typically use different filtering methods. It follows that the lossless data compression is not always the best solution and always need a reasonable tradeoff between signal quality and quantity of information loss during processing and controlled process granularity.

The features of the pre-processing subsystem and data storage optimization in monitoring systems can be summarized as:

- optimization of storage data structures and methods of presentation specific controlled processes;
- the development of compression methods as the source, and pre-processing, "purified" data: both with and without the use of filtration to reduce random noise.

Let us continue briefly on the consideration of the use of both known and new approaches to improve the efficiency of data storage subsystems and results compare the effectiveness of the use of certain techniques for solving problems of climate, environmental and technical monitoring.

2 Compression methods based on optimization of the data structure

To reduce the amount of stored data, classical compression techniques can be used, for example, the approximation of the observed data using linear or parabolic function, Fourier series or the applications described in [18, 19].

The method, described in [20], an approach that allows no loss of accuracy to minimize the amount of data stored on the server simultaneously with reducing time access to database. The basis of this approach was the principle of optimization of the type of fields used for recording samples and to record values of time corresponding to these points. Its foundation was laid such an important feature of the monitoring of the recorded raw data, like the uniformity of the sample and the low accuracy of the measurement of controlled physical quantities limited in most practical applications, the value of reduced error is no more than 0.5…1%.

Therefore, optimization was proposed to use for the storage of regularly updated data fields with only fixed-length integer without sign (in the MySQL database and a number of other databases such fields have the option "unsigned" [21–24]). Since the actual data samples can be both positive and negative, and maybe have a fractional part, it is proposed to carry out the normalization of the record with the formula [20]:

$$y = C * x + y_0. \quad (1)$$

where x – observed data, C– some multiplier, and y_0 – shift.

Since many databases provide rich types of fields that differ in the number of bytes, always it is possible to pick up a field type with minimum amount of bytes that suitable for recording the normalized data. For example, to record temperature values with a resolution of 0.0625 ° C in the range from -55 to + 125C ° (that is provided, for example, when using digital DS18B20 sensors), two bytes is sufficient instead to keep it as a real number with a field of 4 bytes. Moreover, it will be possible to store data in this range, even with a resolution of up to 0.005 ° C, if selecting C = 200 and y_0=1000. Such resolution for temperature recording can be obtained by averaging several measurements. In addition, even for the storage of such sizes as daily and monthly consumption of heat, water and electricity for most objects with sufficient commercial and operational accounting accuracy, two bytes will be quite enough.

In addition, it is possible to reduce the size of the field due to the transition from the store values collected since the start (reset) to the values in the system, is accumulated in a relatively small intervals, just as in a sigma-delta ADC and video compression systems. With this campaign sometimes is sufficient to store the data in fields with sizes up to 1 byte.

The next step, which allows to move at least a two-fold reduction in the volume of stored data, is the transition from the field for storing the date and time (according to the standard ANSI / ISO SQL 92 is a field of type DATE, TIME, DATETIME, and TIMESTAMP [22]), to unsigned integer fields . The practice of long-term application of this approach has shown that it is expedient to transform this first convert the time to UTC – format that eliminates the problem of time shifts due to irregular daylight savings clock of administrative time and possible change of time zones, the local time for a specific area. Then to view data for a specific period of time in a particular locality translation UTC – time to local time can be done either with the use of stored in a separate database table on time shifts and standard database functions and CGI – handler, or by writing codes for conversion functions.

When determining the amount allotted under the temporary reference value of the field must be guided by the minimum necessary time data sampling period, and the time required to store these data. In the base of the form are recorded values.

$$k_i = (t_i - t_o) / \Delta t, \quad (2)$$

where t_i – observation time, t_o – system start time, and Δt – sampling interval. For example, the following Table 1 shows that for most practical applications of this approach can store the temporary sampling in field of 1 to 3 bytes instead of 8 bytes allocated for storing the date and time in most databases. Since the lifespan of software without modification makes about 3–7 years, in the table in bold fields that provide the most effective combination of terms of data storage, field width, and the data sampling interval.

Table 1. The dependence of the maximum possible storage time, years of the sample period and the size of the field.

Number of bytes	Data sampling interval									
	1c	10c	30c	1 min	5 min	10 min	30 min	1 hour	1 day	1 month
1									0.7	21
2						1.3	**4**	**7.5**	180	
3	0.53	**5.3**	**10.6**	32	160	133				
4	136									

As seen in this table, the most appropriate field is 2 bytes and sampling intervals 1 hour. This eliminates from the database table, which records the measurement results, a separate field for storing temporary labels, since, as already mentioned, the majority of controlled variables also require normalizing after its storage for two-byte field. Furthermore, if the structure of the incoming data clearly fixed and the data is received regularly, it is possible to remove the record timestamps and the identifiers of controlled parameters. Assuming that the size of the field for storing the identifier of the controlled parameter is one byte, and fields to store time stamps and the data itself - two bytes, then the transition to the fixed structure of the measurement results will reduce the total recording length of 5 bytes to 2 bytes, i.e. reduce the amount of data stored in 2.5 times. At the same time, this approach eliminates the need for indexing and data tables without increasing the time to search for the information required, since in this case, the right place in the table is calculated in advance, and is set in a typical option OFFSET SQL - SELECT command.

However, in real systems, failures and loss of receiving information from certain channels are always possible. Therefore, to improve the storage reliability and the possibility of recovery even in case of partial destruction of a file, it is offered to enter a tag for identifying bad or missing data and transmitting time label of feature using records that are not used for recording the monitored parameters values. The most convenient for this is using a combination of all zeros or all ones if the data is stored as unsigned integers. This does not violate the normalization mechanism and practically does not affect the dynamic range of possible changes in the controlled variable.

Additional data compression can be achieved if use bitwise processing when the values of several samples of the same or different controlled parameters are packed into one field in the database table. However, such a solution is difficult to standardize, since the nature of the packaging bit will always depend on the particular set of stored values and their dynamic range. In addition, for each case will have to write the code for the compress, and to decompress the data, which is not very convenient.

A much more effective in terms of data compression, and improve the reliability of storage and provides faster access method based on their grouping into separate clusters for some common characteristics. Among these, first of all, are the equal intervals of samples, as well as the probability of a simultaneous demand. As a result, a common database will

be a separate set of tables, each of which will be kept homogeneous sampling time, location, the destination, the measurement units or other status information.

The specificity of the majority of the monitoring systems, and first of all, interactive systems is the need for the best possible real-time access to current information, while access to information for prior periods is not so urgent, and may require more time. To implement this functionality, the database should provide a buffer tables with small volume for data with a small sample interval. Simultaneously, the current data is written into the buffer table and large blocks are overwritten in the main table. This approach minimizes both the access time and the time of recording data.

Consider further compression methods based on the account of features of controlled processes, as the presence of a strong correlation between adjacent samples of the controlled parameter.

2.1 Difference schemes compression

In practice, very often situations that some time for a series of samples of controlled magnitude does not change its consecutive values or these changes happen actually at the noise level, complicating the analysis of the observed process. Examples of such signals observed in conducting studies to assess the outdoor air temperature influence on the temperature in the controlled room, shown in Figure 1.

In this Figure, each step-jump signal corresponds to the temperature change by 0.0625 ° C. It would seem to get rid of such jumps generated by random noise is quite easily by smoothing the signal using digital filters. However, it is known that any low-pass filter increases the lag effect of the system [25–27].

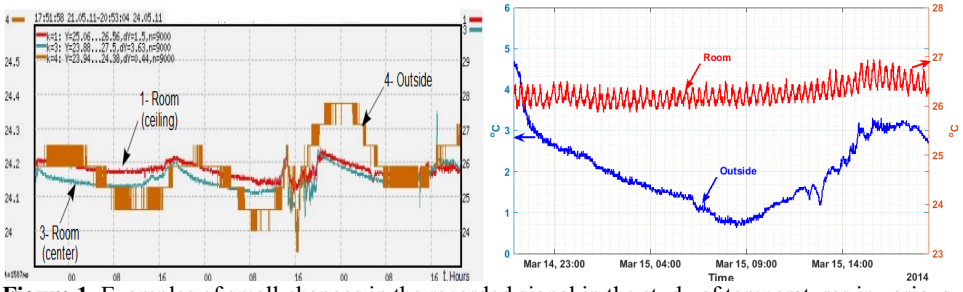

Figure 1. Examples of small changes in the recorded signal in the study of temperatures in various points of the controlled room and the outside air.

If the monitored parameter changes over time rather smoothly, has a clearly expressed trend line, and the difference between the parameter values between adjacent samples is much smaller than the range of its possible changes, a high compression ratio is provided by difference schemes using run-length encoding algorithm (RLE, run-length encoding) [28–30]. Compression scheme using RLE commonly used for streaming video and audio compression. However, unlike the multimedia processing systems, monitoring systems has a higher demands regarding reliability of information storage. Therefore, in the unacceptable situation, when failure in one count information can lead to distortion of the whole data block.

The essence of the proposed modifications RLE – compression methods is that the difference is not taken between adjacent samples (e.g., between image frames), but between the current count and a fixed time interval reference value. Due to this, failure of one reading will not affect, in any way, the reliability of data recording in the other readings.

Since, the algorithms with partial loss of data provide the best compression ratio and, on the other hand, increase the signal quality by smoothing the random noise. In [31] was carried out an experimental assessment of level of such noise. The initial data were taken of temperature indoors and outdoors, such as those shown in Figure 1. To determine the noise level of the temperature sensor was placed in a box made of foam with a wall thickness of 2 cm thermostat. Sampling interval was 30 sec with a total of 2030 samples, the first of 30 samples of them were used for the averaging and discarded from the end. To find the average (reference) values of the sensor, signal was smoothed by moving average that has a rectangular window with a width of 10 samples. At the same time, the noise was defined as the difference between the original and the smoothed signal. The latter was taken as it or rounded up to the nearest integer. For greater clarity, the signal was reckoned not in terms of temperature, but in its levels of quantization. For temperature, the digital sensor DS18B20 used in the experiment has quantization equaled 0.0625 ° C.

Histograms of the probability distribution of observed deviations for different selections of the reference signal are shown in Figure 2.

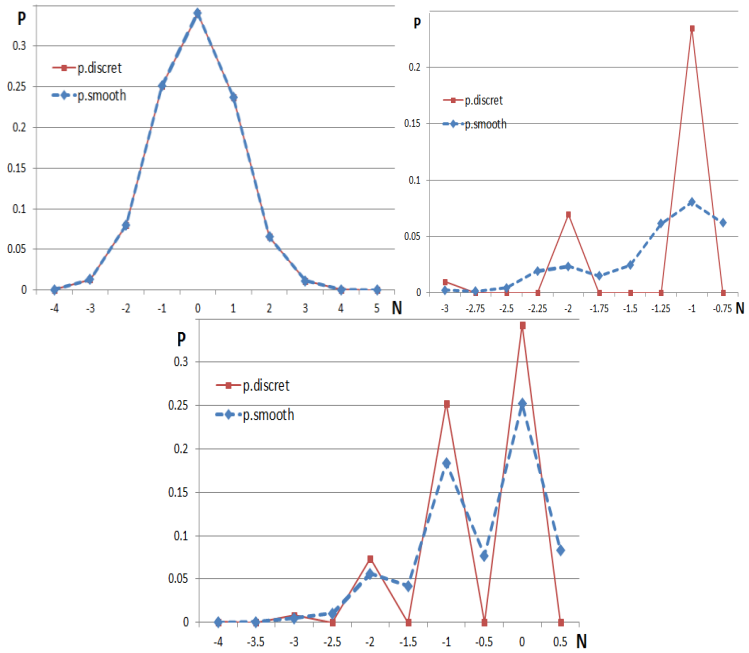

Figure 2. Probability Histogram of the temperature deviations (in N quantization levels) for various values of the partition intervals ΔN:
a) ΔN = 1; b) ΔN = 0.5; c) ΔN = 0.25. The deviations were calculated relative to the mean of the smoothed (p.smooth) or medium sampled (p.discret) of signal level.

As can be seen from this Figure, the sensor noise is pronounced discrete nature, and their function is well approximated by a normal distribution law. It can be assumed that the noise caused by the maximum deviation of the signal from its true value lies within three quantization levels, and noise STD is approximately one level of quantization. It follows that the modified RLE compression method should also provide a recovery error signal in the same range, i.e. within one - three quantization levels, without significant loss of information data.

To estimate efficiency of the proposed methods of data compression, a series of experiments based on the developed information and measuring system of technical and environmental monitoring described in [2, 32] was carried out.

2.2 Experimental Procedure for the comparison of various compression algorithms as applied to the temperature monitoring

The essence of the conducted field experiments was as follows. For processing, two temperature signals of 8200 samples (slightly less than 3.5 days) inside and outside building were used.

One were a relatively smooth in nature, while the other contained anomalies in the form of a sufficiently rapid temperature changes caused by both natural factors (wind, cloud) and anthropogenic (airing, the inclusion of a heater). Initially, all of the samples were subjected to two-fold smoothed moving average with a width of rectangular window were 2 samples and the first 4 samples are discarded. Simultaneously, controlled the maximum deviation between the original and the smoothed signal Δx does not exceed 1…3 quantization levels. This treatment allows, on the one hand, minimizing random noise, and on the other - to keep all the little details of the changes in the signal level.

Since the RLE method works with integer values, all samples of the smoothed time series were rounded to the nearest integer values.

During compression, each signal was divided into equal time intervals. For each interval, polynomial coefficients of polynomial approximation were calculated. When approximating trigonometric functions of the coefficients of the Fourier series were determined using a fast algorithm "butterfly". For the possibilities of its application the number of the samples in the interval was selected multiple 2^n.

After finding the coefficients of the approximating functions for each interval, the difference between these functions and the original smoothed time series was calculated and the difference signal standard deviation, maximum deviation and coefficient of determination were calculated.

The described sequence of operations, since splitting a temporary row data into intervals, repeated for various extent of intervals (i.e. for various number of samples inside an interval) until the maximum value of the deviation did not exceed 2-3 quantization levels.

At the final stage of the experiment, the compression ratio was estimated for packed data; it always remains the same for the reference methods and is determined by the expression:

$$K_{compress} = 0.25 \cdot K_{samples} / K_{coef}. \quad (3)$$

where $K_{samples}$ – the number of samples within interval; K_{coef} – the number of coefficients in the approximation function. The presence in the formula of the constant coefficient 0.25 is because, unlike the original integer valued sample values that occupy at most 1-byte, the coefficients of the approximating functions are real type and for storing their values already require 4 bytes for each one.

It is important to note that, unlike the proposed compression method, the approximating functions did not store the samples in the database, but the real coefficients of the approximation polynomial.

3 Experimental results

Experimental results of applying different compression methods for compressing the data summarized in Table 2, where $K_{compress}$ – the compression ratio determined from the expression (3); STD – standard deviation of the error; Δx – the maximum modulus of the difference between the approximated and original signals, $K_{samples}$ – the number of samples within in the interval, K_{coef} – the number of real coefficients in the approximated function, K_{rep} – the number of repetitions of the new values of controlled quantities.

Table 2. The experimental results for temperature series compression.

Type of approximation function	K_{coef}	$K_{samples}$	Δx	STD	$K_{compess}$
Temperature fluctuations in the room, without anomalies					
Constant	1	6	0.29	0.59	1.50
Linear function	2	20	0.29	0.71	2.50
3rd polynomial	4	50	0.29	0.79	3.13
6th polynomial	7	70	0.29	0.56	2.50
9th polynomial	10	300	0.29	0.32	7.50
8 harmonics Fourier series	18	2880	0.29	0.79	40,00
RLE1	K_{rep}=3	-	0.29	0.49	77.59
Temperature fluctuations in the room, with anomalies					
Constant	1	4	0.29	0.59	1.00
Linear function	2	10	0.29	0.48	1.25
3rd polynomial	4	25	0.29	0.55	1.56
6th polynomial	7	35	0.29	0.53	1.25
9th polynomial	10	300	0.29	0.31	7.50
8 harmonics Fourier series	18	2880	0.29	0.80	40,00
RLE1	K_{rep}=3	-	0.29	0.83	67.84
Outdoor temperature without anomalies					
Constant	1	4	0.29	0.44	1.00
Linear function	2	8	0.29	0.36	1.00
3rd polynomial	4	20	0.29	0.48	1.25
6th polynomial	7	35	0.29	0.53	1.25
9th polynomial	10	300	0.19	0.52	7.50
8 harmonics Fourier series	18	1000	0.19	0.45	13.89
RLE1	K_{rep}=5	-	0.19	0.97	68.61
Outdoor temperature with anomalies					
Constant	1	2	0.29	0.22	0.50
Linear function	2	5	0.29	0.12	0.63
3rd polynomial	4	15	0.29	0.21	0.94
6th polynomial	7	20	0.29	0.22	0.71
9th polynomial	10	300	0.34	0.88	7.50
8 harmonics Fourier series	18	1000	0.34	0.58	13.89
RLE1	K_{rep}=5	-	0.34	0.95	61.65

As an example, Figure 3 shows a portion of a temperature series and its error of reconstruction using various approximation functions. It shows that the reconstruction error for anomaly areas (in this case – outliers) increases dramatically. This leads to reduce the number of samples within the approximation interval. Typical plots of the relation between the interval size and the approximation error STD, as well as, the maximum error of approximation are shown in Figure 4.

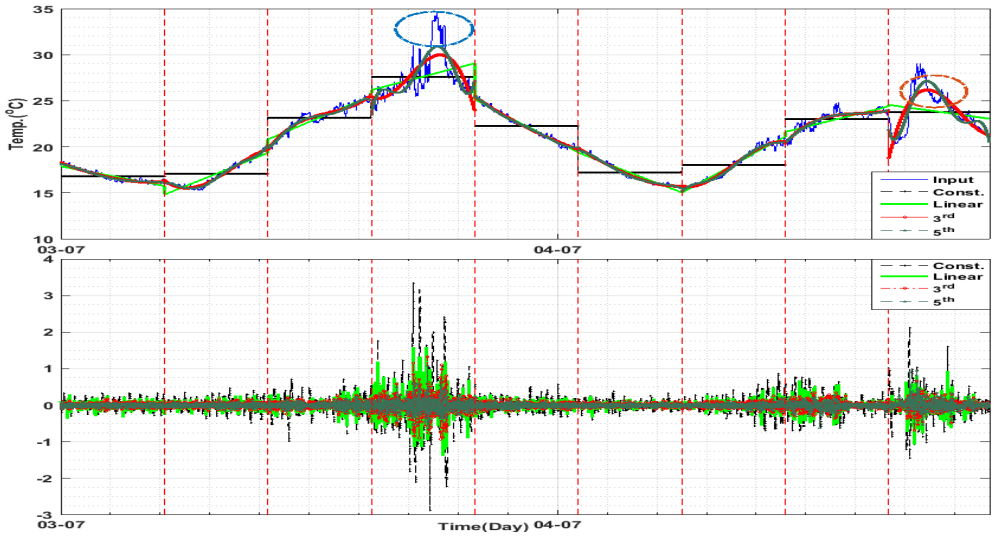

Figure 3. Fragment of the time series of temperature measurements (top graph) and the errors of its approximations using different approximation functions.

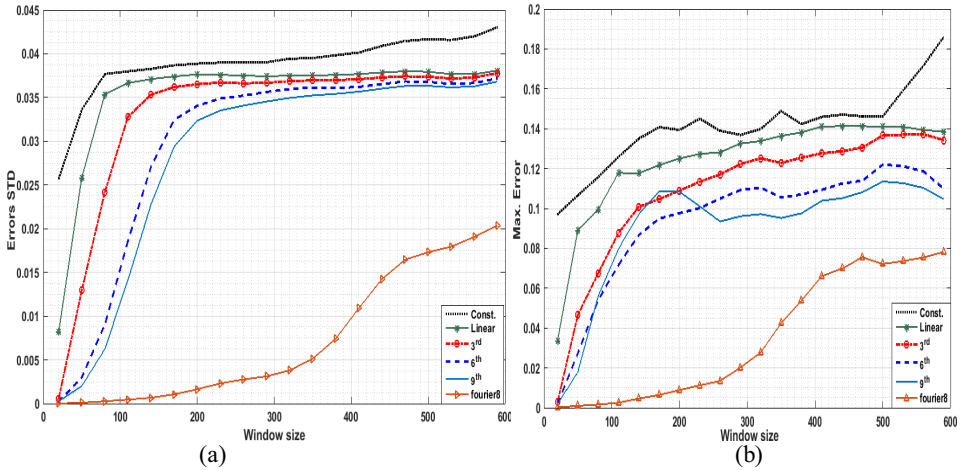

Figure 4. Dependency of approximation errors STD (a) and the maximum errors of approximations (b) on the number used to approximate $\boldsymbol{K}_{\text{samples}}$ samples.

4 Conclusion

As the performed experiments showed, the proposed method for compression has clear advantages with respect to approximation methods, especially in signals with anomalies. In addition, it has less computational complexity, since all the operations are performed in integer arithmetic and it uses input data directly from the ADC output. However, if the accuracy is not sufficient to reproduce the input data, it is better to use high order polynomial approximation or Fourier series for input data that has not anomalies.

References

[1] M. Hirayama, MATEC Web of Conferences **59**, 01002 (2016) doi: 10.1051/matecconf/20165901002

[2] H. M. Hussein, R. V Kuntz, L. I. Suchkova, A. G. Yakunin, Sci. J. of the theor and applied researches (Phys.) (1), 210 (2013) URL: http://izvestia.asu.ru/2013/1-1/phys/ TheNewsOfASU-2013-1-1-phys-13.pdf.
[3] Y. Li, H. Zhao, J. Fan, MATEC Web of Conferences **22,** 04008 (2015) doi: 10.1051/matecconf/20152204008
[4] M. F. Othman and K. Shazali, Procedia Eng. **41**, 1204 (2012) doi: 10.1016/j.proeng.2012.07.302
[5] M. Haefke, S. C. Mukhopadhyay, H. Ewald, Conference Record - IEEE Instrumentation and Measurement Technology Conference, 5944154 (2011) doi: 10.1109/IMTC.2011.5944154
[6] G. Ohring , A. Gruber, Advances in Space Research **28**, 207 (2001) doi: 10.1016/S0273-1177(01)00350-7
[7] G. B. Wiersma, *Environmental Monitoring* (CRC Press, New York, 2004)
[8] M. Malyutov, B. Ryabko, J. Astola, *Compression-Based Methods of Statistical Analysis and Prediction of Time Series*. (Springer International Publishing, Switzerland, 2016)
[9] C. Zhen, B. Ren, Int. Conf. on Computational Intelligence and Software Eng., 5366670 (2009) doi: 10.1109/CISE.2009.5366670
[10] M. B. Lin, Y. Y. Chang, IEEE Trans. on VLSI Systems **17**, 1297 (2009) doi: 10.1109/TVLSI.2008.2003512
[11] I. Suarjaya, IJACSA **3**(8), 14 (2012)
[12] P. Lindstrom, M. Isenburg, IEEE Trans. on Visualization and Computer Graphics **12** , 1245 (2006) doi: 10.1109/TVCG.2006.143
[13] D. Salomon, G Motta, *Handbook of data compression* (Springer-Verlag, London, 2010)
[14] N. Kimura, S. Latifi, ITCC ’05 **2**, 8 (2005)
[15] G. Graefe, L. D. Shapiro, Proc. 1991 Symposium on Applied Computing, (1991)
[16] J. Ziv, A. Lempel, IEEE Trans. on Information Theory **23**, 337 (1977) doi: 10.1109/TIT.1977.1055714
[17] H. M. Hussein, A. G. Yakunin, Polzunovsky vestnik (2), 65 (2013) [in Russian]
[18] H. M. Hussein, A. G. Yakunin, XI inter. scientific-practical Conf. "Science and technology: step into the future-2015", 73 (2015)
[19] K. Sayood, S. Bhanja, N. Ranganathan, *Lossless Compression Handbook* (Academic Press, Cambridge, 2003).
[20] H. M. Hussein, L. I. Suchkova, M. A. Yakunin, Polzunovsky Almanac, AltSTU (2), 48 (2012). http://elib.altstu.ru/elib/books/Files/pa2012_2/pdf/048Hussein.pdf. [in Russian]
[21] MySQL 5.0 Reference Manual, URL: https://docs.oracle.com/cd/E19078-01/mysql/mysql-refman-5.0/
[22] V. Vaswani, *MySQL(TM): The Complete Reference* (McGraw-Hill, New York, 2004)
[23] C. Vicknair, D. Wilkins, Y. Chen, ACM-SE ’12, 176 (2012) doi: 10.1145/2184512.2184554
[24] B. D. Blansit, J. of Elec. Resources in Medical Libraries **3**, 135 (2006) doi: 10.1300/J383v03n03_10
[25] R. Losada, *Digital Filters with MATLAB* (MathWorks, Inc., Natick, 2008).
[26] S. W. Smith, The Scientist and Engineer’s Guide to Digital Signal Proc. (1), 261(1997)
[27] G. Taubin, IEEE Fifth Inter. Conf.on Computer Vision, 852 (1995)
[28] S. W. Smith, The Scientist and Engineer’s Guide to Digital Signal Processing, 481 (1997)

[29] B. Žalik, N. Lukač, Signal Processing: Image Communication **29**, 96 (2014) doi: 10.1016/j.image.2013.09.002
[30] D. Salomon, *Data Compression: The Complete Reference* (Springer Science & Business Media, Berlin, 2004).
[31] H. M. Hussein, Proc. 16th Inter. Conf. "Measuring, monitoring, information", AltSTU **1**, 11 (2015). URL: http://mca.altstu.ru/download/proceedings2015.zip
[32] Altai State Technical University, Monitoring system, [Online] URL: http://abc.altstu.ru