12thIMEKO TC4 International Symposium
Electrical Measurements and Instrumentation
September 25-27, 2002, Zagreb, Croatia

COMPRESSION OF DIGITAL MEASUREMENT SIGNALS BY REVERSE SCALING RECONSTRUCTION METHOD

Jerzy Jurkiewicz, Waclaw Gawedzki

University of Mining and Metallurgy,
Department of Measurement and Instrumentation, Krakow, Poland

Abstract – It is noticed in the paper that scaling of digital measurement signals during computer processing increase their volume so that after using standard compression methods it is impossible to obtain the volume of the compressed original signal. A reverse scaling operation with reduction of numeric errors is proposed as the method preceding compression. The operation restores the original signal and, therefore, restore the original signal volume. The rule of encoding measurement results into a certain number of significant digits is recalled. The binary version of this rule is presented and it is observed that its application improves signal compressibility.

Keywords: signal compression, measurement signal, data acquisition

1. INTRODUCTION

Digital technique is often used in measuring systems recording time variable signals. A typical recording system consists of a sensor of the measured quantity, being usually a measured quantity-voltage converter (to make the problem simple, we omit other possible carriers of the measurement signal), and an A/D converter where the signal is then sampled and converted into digital values $\{x_n\}$ (we assume that this is done according to the principles of engineering).

In popular measurement data recording systems, especially those with data acquisition cards, signal scaling, consisting in changing sample values so that they express values of the measured quantity $\{y_n\}$, is made (in the case of data acquisition cards) on the digital side.

Data transformed this way are usually saved in disk files in ASCII or IEEE Double Precision (DP) formats and transferred among various applications of data analysis and presentation (especially in the case of research work).

After some time we finish the work on a certain group of data and it is right time to archive the data. A natural approach to this problem is employing to data files one of the popular compression applications as RAR or ZIP [1]. And usually here comes a surprise.

Although the compressed file containing the $\{y_n\}$ series is clearly smaller, but may be greater than it is necessary to save the $\{x_n\}$ series with no compression. This suggests an obvious conclusion that instead of the $\{y_n\}$ series it is the $\{x_n\}$ series that should be compressed because the transformation is known. Moreover, the compression of the $\{x_n\}$ series, independently of its format: fixed-point, integer or double, leads to a similar (as to the order) size of the compressed file.

Not all data acquisition software products allow to direct access to the $\{x_n\}$ series, and even if they do we often do not remember this or we do not want to waste some space on the disk, as we carrying out analysis using $\{y_n\}$. It is therefore necessary to recover the $\{x_n\}$ series from the $\{y_n\}$ series using reverse scaling operation.

Successive numerically execution of scaling and reverse scaling operations, even at high precision calculations, does not restore the original value of $\{x_n\}$ because of round off errors generated during numerical calculations, especially when coefficients of this transformations has infinite binary-coded representations. In order to reconstruct the original values of $\{x_n\}$ from a result, a few least significant bits of the binary representation shall be ignored by rounding, eliminating this way possible (perhaps cumulated) numerical errors. The advantages of compressing a file containing $\{x_n\}$ series, usually smaller than the result of the compression of a file containing $\{x_n\}$ series are obvious.

2. THE RECONSTRUCTION METHOD

The $\{x_n\}$ series as the output from an A/D converter can be treated as 12÷16-bit numbers (depending on the resolution on the converter). Signal scaling consisting in changing the values of samples so that these values express the values of the measured quantity is performed through employing (usually) linear transformation:

$$y_n = a \cdot x_n + b \qquad (1)$$

Round off errors may occur then of value two times the value corresponding to LSB in DP fraction [2]. An operation reverse to (1) can be expressed mathematically as:

$$x_n = (y_n - b)/a \tag{2}$$

Again, round off errors may occur here of value defined previously. The worst-case total round off errors may reach four times the value corresponding to LSB. To restore the original values of $\{x_n\}$ after transformation (2), it is necessary to use in an obtained result (in e.g. DP format) rounding zeroing at least 3 least significant bits, canceling thus possible numeric errors. The DP format fraction part has 53 bits, so even after 20-bit converter rounding the binary representation of each number will contain at least 33 successive bits of zero value, which substantially improve the compression ratio. This will occur even if we abandon the suggested format change into fixed-point or integer, and encoding as 2, 3 or 4-byte numbers respectively instead of 8, as in DP. We can see, therefore, that we need not know even the number of bits of the A/D converter the $\{x_n\}$ series comes from (the corresponding bits will be zeroed by themselves).

3. THE TRADITIONAL METHOD

An alternative approach to data preparation for compression is to apply a binary version of the old rule of measurement result encoding [3] to strictly determined number of significant digits so that the last (least significant) digit (its quantum) corresponds the order of measurement uncertainty. The method is based on the knowledge of the quantum $\{x_n\}$ and consists in determining the quantum $\{y_n\}$ from (1) and then the number of significant bits in values of $\{y_n\}$. Because we do not want to increase twice the quantization error, allowing of errors substantially smaller, we add e.g. 3 bits to the obtained number of significant bits, and then we round up the $\{y_n\}$ binary representation to the same number of bits, while zeroing the remaining, less significant bits. Compression of such prepared $\{y'_n\}$ series is better than compression of $\{y_n\}$ series although not so good as compression applied to the $\{x_n\}$ series.

4. RECONSTRUCTION OF SCALING COEFFICIENTS

While the measurement data are being registered, scaling coefficients are settled by choice of the range of measurement torque or its calibration. They are known to the acquisition data program and most programs of that type record them in their configuration files. Unfortunately, the files are mostly in binary form but user manuals do not contain the description of their structure. If scaling coefficients are not recorded, in legible form to the compression program, then they will be often read in acquisition program by manual reading of data file (with special configuration file). The method as well as manual registration are not good for practical application because of arduousness. In such cases, it is necessary to recover the scaling coefficients of registered data.

The $\{y_n\}$ series expressed in (1) can be represented by different sets (a, b, $\{x_n\}$). It is easily to choose, among them, one set in which the $\{x_n\}$ values belong to the integer set.

Then coefficient *a* will be a quantum of values appearing in $\{y_n\}$ series but coefficient *b* can be chosen as one of values from $\{y_n\}$ series e.g. the value nearest to zero or the smallest one.

In order to find a value of quantum a set *W* of well ordered values appearing in $\{y_n\}$ series have to be created

$$W = [w_0, w_1, \ldots w_N] \tag{3}$$

and for each pair

$$(i,k); \quad k > i \tag{4}$$

it is necessary to calculate differences

$$d_{ki} = w_k - w_i \tag{5}$$

Then for all differences d_{ki} it has to be found the highest common factor (it is not integer). In practice, it is proved that mostly searched value is the smallest of differences $\{d_{ki}\}$. Let's note the smallest difference by

$$D = \min_{k,i}(d_{ki}) \tag{6}$$

The highest common factor $\{d_{ki}\}$ is equal to *D/M*, where *M* is natural number. In practice, the relation

$$L_i = \frac{(w_i - w_0)}{D/M}; \quad i = 1, \ldots N_i \tag{7}$$

corresponding to the (2) can be tested for the succeeding *M* values. With exactitude to numerical error ε

$$| L_i - \mathrm{round}(L_i) | < \varepsilon \tag{8}$$

If the threshold of error ε is exceeded then *M* is increased by 1 and testing is continued on from the index *i* in which a condition is not realized. There are possible different modification of algorithm what speed it up. Values *D* and w_0 can be calculated on the basis of fragment of data series, but testing relations (7) and (8) can be done on the following values of $\{y_n\}$ series, time saving for generating the set of well ordered values *W* (3) occurring in $\{y_n\}$ series.

5. EXPERIMENT

For experiment the measurement data registered on the different mechanical objects and climatical data are applied in general 35 series of measurement data.

Presented method is based on restoring the original volume of data (spoiled by the comfort of conversion) but individual usage of the method is not useful. It is easier to use one of the compression programs: RAR or ZIP. Therefore, the comparison between compression rate using RAR program B and compression rate at the composition of reverse scaling reconstruction (RSR) following the compression using RAR program (according to scheme presented on Fig.1)

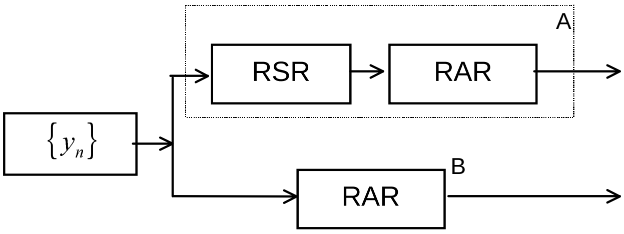

Fig.1. Investigation procedure of compression ratio of RSR method

It was found, that for longer data series advantage of usage of the RSR method is growing smaller. That is way comparisons are made separately for different data lengths (to obtain shorter data series the fragments of longer data series are used).

Additional compression ratio for each data series is calculated as a ratio of file size in B method to the file size in A method.

Exemplary results for data series length 1000 samples are showed in Table 1.

Table 1. Exemplary results for data series length 1000 samples

Signal nr	number of *W* set values	RAR file size	RSR+RAR file size	additional compression ratio (gain)
1	902	4535	1928	2,352
2	52	1470	973	1.511
3	82	1723	1142	1,509
4	633	3672	1656	2,217
5	950	6775	2015	3,362
6	928	4581	1954	2,344
7	43	1416	953	1,486
8	63	1499	1028	1,458
9	537	3352	1604	2,09
10	927	6350	1969	3,225
11	943	4577	1994	2,295
12	49	1430	963	1,485
13	68	1596	1086	1,47
14	549	3399	1602	2,122
15	935	6577	1972	3,335
16	9	599	445	1,346
17	46	1414	965	1,465
18	51	1463	1007	1,453
19	47	1407	954	1,475
20	395	2293	1391	1,648
21	7	356	243	1,465
22	10	389	244	1,594
23	167	1841	1145	1,608
24	67	1342	908	1,478
25	146	1694	1035	1,637
26	654	3015	1811	1,665
27	58	1026	604	1,699
28	401	2261	1355	1,669
29	936	3696	2003	1,845
30	372	2308	1438	1,605
31	17	903	658	1,372
32	489	2492	1553	1,605
33	69	1598	1135	1,408
34	13	899	642	1,4
35	45	1254	880	1,425

Distribution function of additional compression ratio is presented in Fig 2.

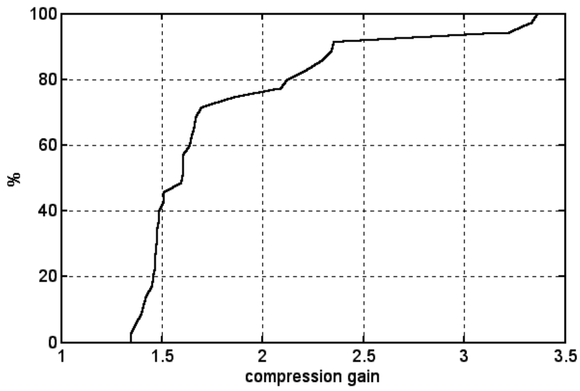

Fig.2. Distribution function of additional data compression ratio from Table 1

The obtained results for each length of data are presented in Fig.3. The figure illustrates distribution functions of additional compression ratio expressed by values: maximum and minimum, as well as percentiles 10%, 50%(median), 90%.

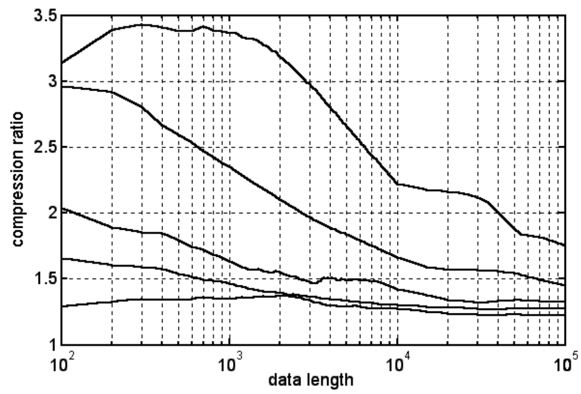

Fig.3. Additional compression rate as a function of data length (in samples)

A phenomenon of growing smaller the additional advantage of compression can be explained as the dictionary methods introduce to the dictionary the binary patterns of each value from *W* set and they can number them. Therefore, when long data are considered there is not large loss caused by delivery of dictionary containing the values.

A storage of dictionary in the RSR method requires two DP numbers and several bits for heading , whole about 20 bytes.

A certain illustration of that argument is a compression effect, when the integers obtained in RSR method are registered in DP format and then submitted to RAR compression (Fig. 4).

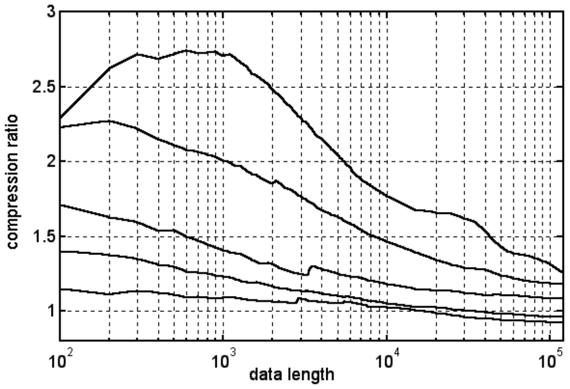

Fig.4. Additional compression ratio as a function of data length when integers are registered in DP format

It is observed an interesting relation between compression rate and power of *W* set (3), what is presented in Fig. 5.

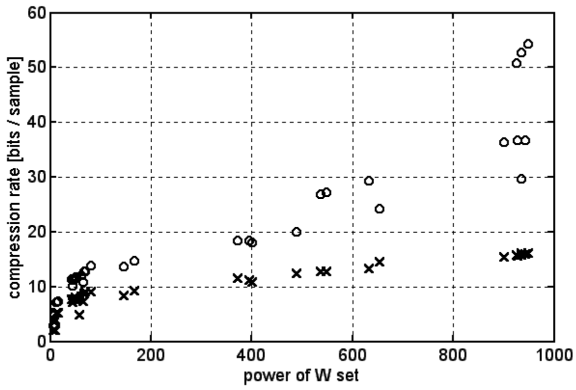

Fig.5. Compression rate versus power of *W* set (Table 1)
○ – RAR, **×** – RSR+RAR

6. CONCLUSIONS

The presented method of reconstruction of an $\{x_n\}$ series is also suitable for nonlinear functions (1) when the sensor nonlinearity correction (only rounding procedure shall be modified) is performed on the digital side. In the case of linear function (1) even automatic determination of *a* and *b* coefficients can be carried out based on the $\{y_n\}$ series although the appropriate algorithms are time consuming. In the case of nonlinear functions (1), when the function is not known, alternative (traditional) method remains to be applied.

The results of investigation on the efficiency (compression ratio) of signals prepared to compression by the discussed methods are presented in the paper. It is expected that the analysis answers the question of how large (additional) compression ratio can be obtained by adequate signal transformation.

7. ACKNOWLEDGMENT

This work is supported by the State Committee of Scientific Research, Poland (grant # 8 T10C 016 21).

REFERENCES

[1] K. Sayood, “Introduction to Data Compression” *Morgan Kaufmann Publishers*, 2 Edition, 2000 U.S.A.
[2] R.L.Burden, J.D.Faires, “Numerical Analysis”, *PWS-KENT Publishing Company*, 1985.
[3] J.R. Taylor, "An Introduction to Error Analysis", *University Science Books*, Sausalito, CA, 1982.

Authors: Ph.D. Jerzy Jurkiewicz, Ph.D. Waclaw Gawedzki, Department of Measurement & Instrumentation, University of Mining & Metallurgy, Al. Mickiewicza 30, 30-059 Krakow, Poland, Phone: (+4812) 6172828, Fax: (+4812) 6338565, E-mail: waga@uci.agh.edu.pl