

Lossless Compression of Vibration Signals on an Embedded Device Using a TDE Based Predictor

Miodrag Malovic¹, Ljiljana Brajovic², Tomislav Sekara³, Zoran Miskovic²

¹*Faculty of Technology and Metallurgy, University of Belgrade,
Karnegijeva 4, 11000 Beograd, Serbia*

²*Faculty of Civil engineering, University of Belgrade,
Kralja Aleksandra 73, 11000 Beograd, Serbia*

³*Faculty of Electrical engineering, University of Belgrade,
Kralja Aleksandra 73, 11000 Beograd, Serbia
office@malovic.in.rs*

Abstract—A lossless compression scheme for the data acquired from three-axial microelectromechanical (MEMS) accelerometer is presented. Time delay estimation (TDE) was applied in conjunction with differential pulse code modulation (DPCM) as the preprocessor to entropy coding, to perform lossless compression on an embedded sensor device with limited memory and speed in real time. The essence of this method is to check if the signal exhibits certain level of periodicity, and code the differences between samples one signal period apart if it does. Limited choice of feasible mathematical operations was an important constraint implied by the architecture of the embedded sensor device. Algorithm execution time was improved by programming in assembler and avoiding unnecessary operations. Experiments have confirmed that considerable compression ratio gains can be achieved if the signal is quasiperiodic.

Index Terms—Autocorrelation; Huffman coding; vibration measurement; wireless sensor network.

I. INTRODUCTION

Data compression is an important tool for energy conservation in wireless sensor networks [1]. The transmission of data using a radio modem consumes much more energy than CPU operation. It is estimated that ten thousand machine instructions consume the same energy as the transmission of a single byte in a typical wireless sensor network [2].

This paper presents lossless compression scheme on an embedded wireless sensor device, developed for vibration measurements in civil engineering. A relatively complex algorithm was implemented on a 6 MHz microcontroller from the 8051 (also referred to as MCS-51) family. These processors are considered reliable and have long history of use in industrial applications. Reputable manufacturers still develop 8051 derivatives with new features and there is an overwhelming base of knowledge and inexpensive software tools. So contrary to some predictions that 8051

microcontrollers are doomed to die out, they make up a significant part of the world's market in recent years [3].

II. LOSSLESS DATA COMPRESSION

A two-step procedure is usually applied to perform lossless data compression. In the first step, raw data is transformed into groups of bits (such as bytes or words) so that its entropy is decreased. In the second step, known as the entropy coding, these groups are replaced by codes of lower bit length. Higher the frequency of a group, lower number of bits is assigned to its representation in the compressed data set.

One of the common preprocessors to entropy coding is differential pulse code modulation (DPCM) [4]. If each sample of a signal is encoded as the difference between it and the previous one, the total number of unique symbols is likely to be reduced. This is because variations between consecutive samples usually cover a smaller range, and multiple channels having different offsets but similar dynamic characteristics can be recorded this way eliminating the need for constant transmission of the DC information.

A difference can be calculated not only from the previous sample in the stream, but from a value predicted by an arbitrary algorithm too. Choosing a proper prediction algorithm is the key to successful data compression. Certain input signal properties must be known for this to be possible, though.

Entropy coding is usually performed by one of two algorithms: Huffman and arithmetic [5]. Huffman coding assigns fixed bit sequences to input symbols, whereas arithmetic coding features no fixed codes. True arithmetic coding is the optimal way to perform entropy coding [6], but its practical implementation does introduce some losses because finite precision registers and memory locations must be used in the process. Although binary arithmetic coding is superior to Huffman's in terms of compression ratio (typically more efficient by several percent), it is slower and usually avoided in embedded systems.

III. PERIODICITY DETECTION

Time delay estimation (TDE) is the procedure of identifying the time shift of a signal for which maximum cross-correlation function with a reference signal is obtained. The cross-correlation function is defined as the product of two signals with their direct offsets removed. TDE is most frequently applied in active systems such as sonars and radars to detect echoed signal match (and thus calculate the distance to the object) and in passive systems (where no artificial signal source exists) such as microphone arrays or seismic sensor arrays, to locate the source of the signal [7]. Similar methods are used in automatic control systems for the determination of process dead time [8].

Single signal autocorrelation function is obtained by multiplying a signal with its own image shifted in time. To recognize a repeating pattern in the signal (qualify it as periodic or quasiperiodic), autocorrelation function must have at least one (other than zero-shift) local maximum such that its value compared to the referent (one obtained for the zero shift) is significant. The exact value of this ratio (referred to as normalized autocorrelation) cannot be unambiguously defined. It depends primarily on where we put the margin between quasiperiodic and aperiodic, but it is also related to signal to noise ratio and other factors.

Most alternative techniques for periodicity detection, like Fourier transform and least squares spectral analysis (LSSA [9], also known as Lomb-Scargle periodogram [10]), include the use of trigonometric functions and are therefore inappropriate for embedded systems with limited computational capacity.

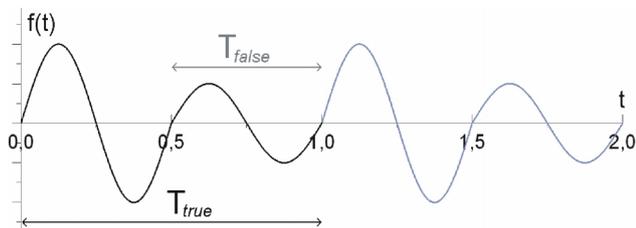


Fig. 1. A sample signal with alternating semi-periods.

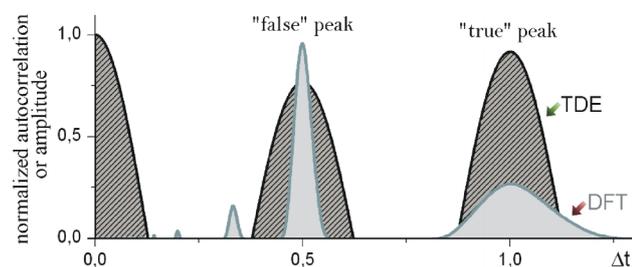


Fig. 2. The comparison of TDE and DFT applied on the signal from Fig. 1.

Autocorrelation also has distinct advantage over other methods if the fundamental frequency is masked by its harmonics. An example is shown in Fig. 1 (a sine with varying amplitude). A theoretical question arises which value should be declared the period of this signal. For prediction purposes, it is better to compare sections that are as similar as they can be, so it is preferable to declare the large period “true”. The results of autocorrelation and Fourier transform are presented in Fig. 2, proving that TDE is a better tool for this purpose. Similar results are obtained

for signals with repeating sub-periods (two or more) that vary not only by amplitude, but also by length and shape.

An alternative method for periodicity detection, attractive for its relative mathematical simplicity and low computational cost, is Enright's (chi-square) periodogram [11]. It is based on averaging the values of equally spaced samples and observing the standard deviation of the averaged set. If the distance between samples matches the signal period, this value nears the overall standard deviation (since the averaged set resembles one full period), unlike if uncorrelated values are averaged, in which case the signal fades. This method is sensitive to harmonics and noise, but it could be considered in processing smoother signals.

Periodicity based DPCM, similar to the one described here, was first considered in telephony in the 1960's, but due to speech signal nature (variable short-term periodicity) and relatively high computational cost, it was mostly restricted to pitch period detection related algorithms [12]. On the other hand, data compression on modern computers has advanced far beyond period detection and employs very complex algorithms. Typical wireless sensor network nodes fall somewhere in between. They do not need to actually stream the data (a certain delay is allowed), yet they lack CPU strength for advanced algorithms (such as MP3, for instance). Therefore, DPCM of intermediate complexity is suitable for such systems.

IV. PRACTICAL IMPLEMENTATION

A. Embedded Sensor Device Design

A wireless sensor network for measurements in civil engineering was developed at the Faculty of Civil engineering in Belgrade [13]. It is composed of sensor devices (nodes; block diagram is shown in Fig. 3) and a base station (hub). Its primary purpose is structural health monitoring (SHM) related vibration measurements [14]. Sensor device boards are made in surface mount technology and use Analog Devices ADuC845 microcontroller. It features multichannel A/D converter, real time clock, embedded thermometer, and negligible consumption sleep mode. Peripheral components are denied power supply during the sleep, so only the circuitry needed to keep the processor functional is powered all the time. Solar cells can be mounted optionally, as additional energy source. Main sensor of the device is three-axial MEMS accelerometer LIS3LV02DL. A packet modem is used for radio communication.

Embedded real time operating system (RTOS) is completely original. It was designed in embedded C and assembler from scratch, providing full control of device operation on the lowest possible level.

A personal computer, usually a laptop, serves as the hub of the system. A program was developed for MS Windows to run the measurement process using another modem connected to the PC serial port.

One of the functions of this system is modal analysis of vibrations of big civil engineering structures, such as bridges, dams and towers. The processing is performed at the base station, where the data from different sensors is compared to determine precise relations between phases of

mechanical oscillations at different points. The system is optimized for good time synchronization.

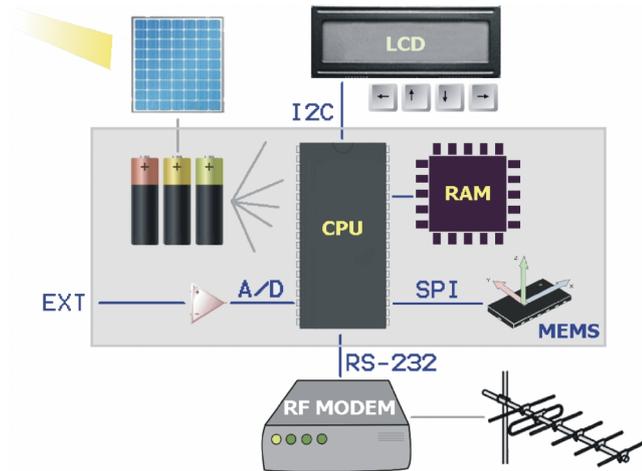


Fig. 3. Block diagram of the wireless sensor device (node).

Sensor devices spend most of their time in sleep mode, and awake periodically to check for the presence of base station broadcast. If it is detected, they remain in stand-by mode for a prolonged period, awaiting commands. One measurement cycle acquires 3200 3×12 -bit samples (for three axes) from the accelerometer. Sampling frequency can be programmed in the 40 Hz to 2560 Hz range. Measurement results are compressed before being transferred by radio. The process is performed with limited memory resources (32 KB, where raw data buffer and system variables occupy more than half) on a processor with 6.29 MHz pace.

B. TDE Triggering and Speed Optimization

In order to avoid compression ratio deterioration by applying TDE-DPCM on a non-periodic signal, conditions must be defined whether to use it or to revert to coding differences between consecutive samples. Sufficient relative autocorrelation integral value (normalized autocorrelation) for the detected time shift (period) is a conspicuous parameter, since it incorporates the information about signal regularity and signal to noise level. A universal threshold is difficult to establish, but the practice has shown that it is usually around 25 % so this value was set as default. It is closely connected to the characteristics of the applied sensor and signals recorded, so different values might be better suited for other measuring systems. In addition, if the signal is low magnitude or has good compression ratio, TDE-DPCM preprocessing should be cancelled. Mean deviation threshold of three quanta was set empirically. A threshold for the compression ratio was not established because it is highly correlated with the mean deviation.

TDE analysis may be performed with the step smaller than the sampling interval, so that one instance of the signal is interpolated [15], but this is not feasible on embedded systems.

Performing single cycle autocorrelation integration requires considerable number (hundreds or thousands) of 12-bit signed multiplications (on the processor with a machine instruction for 8 bit by 8 bit unsigned multiplication only), and this needs to be repeated for different time shifts,

so there is a two-dimensional loop that needs to be executed hundred thousands or millions of times. Optimizing this process was performed by a number of techniques. A 20 % sliding window from the middle of the recording was chosen to perform the correlation and it was shifted 20 % of the total time in one direction. This includes about 400,000 cycles per axis. However, full calculation of the integral for a specific time shift is not always necessary. It can be halted if an arbitrary number of samples first processed (1 % total or 5 % of the sliding window was chosen) yields a negative result. Furthermore, shortest periods detected in hundreds of performed experiments were equal to 8 sampling periods, so it is safe to skip the next 4 shifts before performing another integral calculation. By doing this, execution time decreases to 2.5 seconds per axis. In comparison, total Huffman coding time ranges from under a second (noise) to 20 seconds (most complex signals). Further improvements can be achieved if all DPCM data with absolute value larger than 8 bits (9-bit signed) is treated as maximum 8-bit integer and multiplications are performed with a single instruction. Civil engineering structures vibrations are seldom high frequency and high amplitude so this is usually justified (higher differences are very rare) and the speed increase is about 25 %. In addition, if the period is expected to be less than 20 % of the full recording, a smaller value can be set for the maximal time shift. A property of vibrations of big structures is taken into account to speed up the process here. Main axis oscillations are sometimes high period (over a second), but lateral oscillations of big structures usually do not have significant low frequency components. Therefore, a smaller number of shifts can be performed to determine optimal transverse axes periods. Fine-tuning the algorithm to optimize for speed or compression ratio is possible at this point. The current algorithm uses the same 20 % window for transverse axes but shifts it by 2 % total time only to determine optimal periods. Main axis period (if outside this range) is processed too and included in the transverse periods ruling.

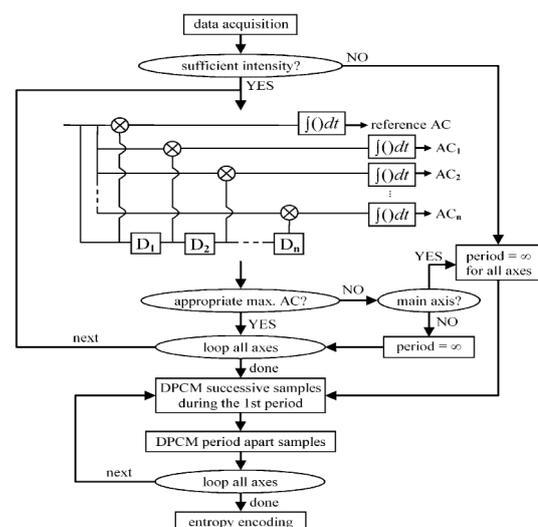


Fig. 4. Approximate compression flowchart; AC denotes autocorrelation.

This way TDE execution time is around 3 seconds (with 12-bit multiplications). Approximate compression flowchart is presented in Fig. 4.

All loops and recursive procedures in TDE, DPCM and Huffman coding are written in assembler to maximize speed. Speed increase compared to the embedded C code (for the identical algorithm), already optimized for speed by the compiler, is 1.6-2.2 times. Better accelerations are achieved for high entropy signals, which compress more slowly.

C. Huffman Tables and Memory Optimization

Primary component of the signal is usually present on only one, main axis. In addition, LIS3LV02DL itself has higher noise on z-axis (most commonly used as main) when it is exposed to $\pm g$. This is explained by the fact that the chip is actually comprised of two separate sensors, one for x and y, and one for z acceleration measurements. Construction of their springs and electrodes is not identical and only z-axis noise increases considerably when corresponding proof mass is away from its equilibrium position [16].

Three variations of Huffman table alignment were considered: single frequency and Huffman codes table for all axes, two tables (for main and transverse axes), and a table for each axis. It was realized that recording two tables is usually better than one, while three tables variation is always inferior. This conclusion cannot be generalized and the best solution depends on the particular signal characteristics.

While the measurement is in progress, 12-bit data is recorded in common 16-bit integer form. Average value calculation (rounded integer) is performed on the fly so the signal is stripped of the direct component as soon as the measurement is over. TDE analysis is then performed to determine if there is periodicity in the signal. If the conditions of sufficient mean deviation and normalized autocorrelation are met, DPCM starts by coding the differences between successive samples during the first period, and continues by coding the differences between samples one period apart afterwards. Otherwise, successive samples DPCM is applied on all data. Finally, in order to increase available memory for tables of frequencies and Huffman codes, the record is packed, decreasing the size from 19200 to 14400 bytes (4 unused bits per word are eliminated). The 13th bit is omitted in this procedure. Theoretically, this might cause losses, because DPCM is not able to code the differences with absolute value larger than half of the full scale using the original number of bits, but this condition was never encountered in practice. Packing the record increases execution time a bit, due to more complex access to the data that is not byte aligned, but freeing the memory is necessary.

There is still not enough room for the complete table of frequencies and Huffman codes. Therefore, the differences (DPCM symbols) larger than 9-bit signed (± 255) share a single entry in the table, reducing its size to 512 records. On any occurrence of such a symbol, the designated code is written into the output stream followed by the raw 12-bit value. In most experiments, large differences did not occur at all. Only the experiments with high frequency vibrating platform and empty trains produced this condition. Large differences probability can be further reduced by turning the on-chip low-pass filter on.

Huffman tables are written to the output stream using pre-order traversal [17] so that their packed sizes equal number

of symbols occurring in the input data times their size (9 bits) increased by approximately 1.5 bits. Compared to tree structure traversal in general, Huffman tree is especially convenient for the application of this method since it is full binary (all internal nodes have exactly two child branches). Pre-order traversal solution is not always optimal, but it offers substantial savings in cases of high data entropy (poor compression ratio).

To conserve space, variables not used simultaneously share the same memory locations, which is known as data overlaying. For example, output buffer start is located 2 KB below the input (raw) data buffer, and the raw data gets overwritten as it is compressed.

V. EXPERIMENTAL RESULTS AND DISCUSSION

The more uniform mechanical excitation of the monitored structure is, the better are the results obtained by TDE-DPCM preprocessing. Vibrations induced by a vibrating platform are the best example. Compression ratios were improved significantly in over 50 experiments with sine excitation in the 5 Hz–750 Hz range, sampled using 160 Hz and 2560 Hz sampling rates (compared to sequential samples DPCM preprocessing). They were cut by up to 36 % original value (which means they were cut more than in half in some cases). Best gains are achieved when vibration frequency reaches a significant portion of the sampling frequency, because sequential samples DPCM is not appropriate for fast alternating signals.

Normalized autocorrelation for the main axis is good in all experiments, as expected, and ranges from 78 % to over 100 % (values higher than one occur when the amplitude increases in the direction of sliding window shift during TDE). Satisfying compression ratio gains also coincide with higher transverse axes autocorrelations.

The results are presented in Table I. Gains better by 1 % on average can be obtained if the full-shift (20 % of the measurement window) TDE algorithm is applied on all axes.

TABLE I. ORIGINAL COMPRESSION RATIOS (SEQUENTIAL DPCM) AND RATIO GAINS (BY TDE-DPCM) IN VIBRATING PLATFORM EXPERIMENTS.

Signal type	Original CR [%]	CR gain [% original val.]		
		min.	typ.	max.
Vibrating platform, $f_s = 160$ Hz	61-73	7	24	36
Vibrating platform, $f_s = 2560$ Hz	48-79	-1	11	32

Ten experiments were performed to monitor wooden sleeper vibrations on the railroad for coal transport to the thermal power plant of Obrenovac. Empty trains produced high amplitude vibrations that caused two undesired effects: accelerometers experienced saturation, and high samples differences (over 9-bit signed) occurred. The latter worsened the compression due to the described Huffman table reduction algorithm. Well tamped sleeper features more regular vibrations (shown in Fig. 5) where wagon pass pattern can be visually recognized on the diagram, unlike in the case of the poorly tamped one. Still, TDE detects correct wagon pass periods in both cases. The only exception are occasional multiples of the period, detected when oscillations produced by non-sequential groups of wagons

match better than those of neighbouring ones.

Normalized autocorrelation for the main axis is about 75 % for the well tamped sleeper and 30 %–50 % for the poorly tamped sleeper. Railroad engineers can use this parameter to determine structural health of a sleeper (in combination with its substructure). Other applications of sleeper acceleration measurements include determination of train speed if wagon geometry is known and estimation of sleeper dynamic deflection by double integration [18].

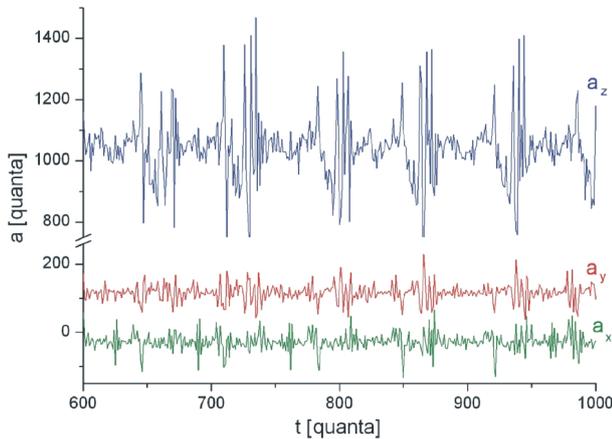


Fig. 5. A regular signal produced by a loaded train on a well tamped sleeper; a_y was shifted up to improve clarity.

Compression ratio gains are shown in Table II. If the main axis period is applied on transverse axes, all results improve by 0.5 % (so that the gains obtained are 2 %–4 % and 0 %–1 %).

The conclusion from the first two sets of experiments is that in cases excitation is periodic, the main axis time shift should be somewhat prioritized over shorter periods established for transverse axes by the time-saving reduced shift TDE procedure.

TABLE II. ORIGINAL COMPRESSION RATIOS (SEQUENTIAL DPCM) AND RATIO GAINS (BY TDE-DPCM) IN THE RAIL TRACK EXPERIMENTS.

Signal type	Original CR [%]	CR gain [% original val.]		
		min.	typ.	max.
Well tamped sleeper, loaded train	71-79	1.5	2.5	3.5
Poorly tamped sleeper, empty train	78-81	-0.5	0	0.5

Tram bridge is the only Belgrade bridge that survived World War II. It has a single lane with tram rails in each direction, used by both trams and cars. Total length is 430 m whereas the distance between two main concrete columns (embedded into the river bottom) is 160 m.

TDE for all axes was mandatory, since signals feature different resonant frequencies on different axes. Twelve experiments were performed for each tram and car excitation. Vibrations produced by trams have higher amplitude and frequency, but they seem more regular, and TDE-DPCM method yields improvements of 2 %–5 % in all cases. Cars, on the other hand, introduce oscillations that are more variable. Only four experiments with cars passing by relatively uniformly during the measurement period triggered the TDE-DPCM. Average 1 % loss was observed. The experiments were not performed during the rush hour, so the traffic was not only sparse (most drivers choose other,

wider bridges, without tram rails) but clustered too, due to traffic lights. Therefore, signals lacked sufficient periodicity throughout the entire measurement period, but some exhibited enough in the investigated section to pass the periodicity test.

Normalized autocorrelation for all axes varies in the 30 %–70 % range and little connection with the compression ratio gain can be observed. This is, also, because it was calculated for the sliding window only, whereas the compression ratio describes the entire measurement, and variations in intensity were considerable between signal sections.

TABLE III. ORIGINAL COMPRESSION RATIOS (SEQUENTIAL DPCM) AND RATIO GAINS (BY TDE-DPCM) IN THE TRAM BRIDGE EXPERIMENTS.

Signal type	Original CR [%]	CR gain [% original val.]		
		min.	typ.	max.
Tram bridge, tram excitation	49-71	2	4	5
Tram bridge, cars excitation	42-55	-6	-1	3

The Gazela bridge is 330 m long with the main span of 250 m and has three highway lanes in each direction. It was built in the late 1960's as the combination of beam and arch with shallow frame beam and diagonal steel-concrete abutments. Over 160,000 vehicles cross it on an average day, which means it is overloaded and often congested.

The vibrations were recorded on the middle of the bridge and on a pier. A sample mid-bridge signal is shown in Fig. 6. Existence of a low frequency harmonic, characteristic for the oscillations of the entire structure, can be spotted here. This large period is ruled by the main axis TDE in some cases, but higher harmonics are singled out sometimes as well. Twelve recordings with sufficient intensity all exhibit 1 %–4 % improvement, except one with ratio depreciation of 0.1 %.

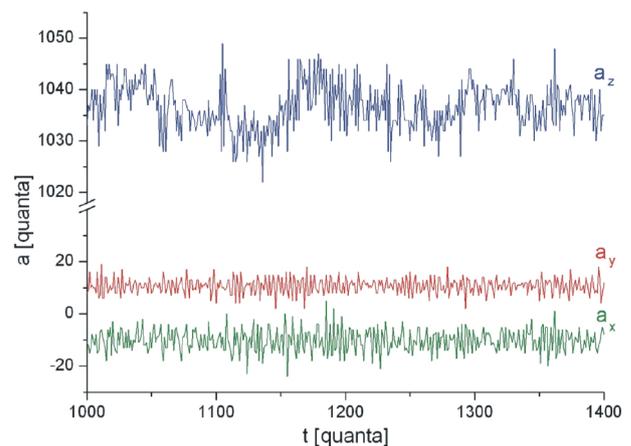


Fig. 6. Vibrations recorded on the middle of the Gazela bridge.

Oscillations of the pier, on the other hand, are largely response to impacts between cars and the bridge expansion joint they run over, since the pier is a node for low frequency oscillations of the entire bridge structure. Cars exciting the pier do not produce uniform responses, so there are cases when compression ratio is worsened (similar to the tram bridge excitation by cars), but on average there is a 2 % improvement from over 30 measurements performed at different points on and inside the pier.

Average main axis normalized autocorrelation is around 25 % in Gazela bridge experiments. Some measurements on mid-bridge did not trigger on-spot TDE, so empirical normalized autocorrelation threshold of 25 % for the main axis, established earlier, proved inappropriate here. These signals were processed later on the PC, considering just the mean deviation to decide whether to perform TDE-DPCM preprocessing or not. This implies that further investigation of the conditions for TDE triggering is needed.

TABLE IV. ORIGINAL COMPRESSION RATIOS (SEQUENTIAL DPCM) AND RATIO GAINS (BY TDE-DPCM) IN THE GAZELA BRIDGE EXPERIMENTS.

Signal type	Original CR [%]	CR gain [% original val.]		
		min.	typ.	max.
Gazela bridge, middle	36-40	0	3	4
Gazela bridge, the pier	29-58	-5	2	5

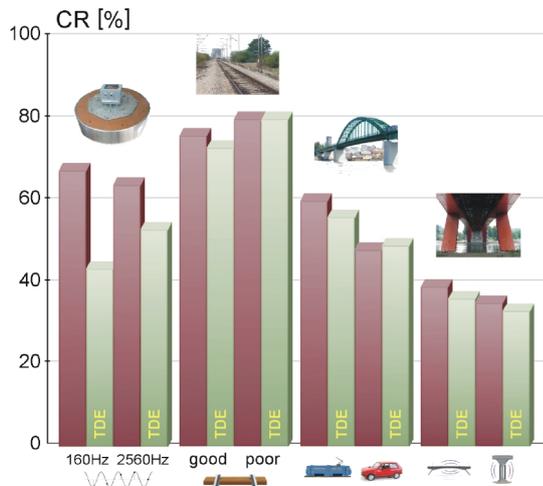


Fig. 7. The comparison of compression ratios (CR) using sequential samples DPCM and TDE-DPCM.

Average compression ratios (in two Huffman tables mode) without and with the application of TDE-DPCM are presented graphically in Fig. 7.

VI. CONCLUSIONS

A wireless sensor system for vibration measurements was developed at the Faculty of Civil engineering in Belgrade. Original embedded RTOS was developed for the sensor devices. One of its features is high-speed (assembler) lossless compression, using TDE-DPCM and Huffman entropy coding. This method is based on periodicity detection using autocorrelation and coding of differences between successive signal periods.

The practice has shown that the described algorithm can yield savings whenever mechanical oscillations of the structure are relatively uniform (quasiperiodic). Experiments on a vibrating platform, railroad structure, and two major bridges were described. Average compression ratio changes, compared to sequential samples DPCM, vary from 1 % deprecation for cars induced oscillations of the tram bridge to 24 % gain on the vibrating platform. Some parameters of the process, important for fast and accurate real time execution, are still manually tuned, and there is room for algorithm improvement.

Additional conclusion derived from the experiments with LIS3LV02DL accelerometer is that it should be used with its

z-axis in horizontal plane, because only z-axis noise increases significantly when it is exposed to $\pm g$ offset. Not only does noise increase impair the signal quality, it also worsens the compression ratio by increasing its entropy.

REFERENCES

- [1] T. Srisooksai, K. Keamarungsi, P. Lamsrichan, K. Araki, "Practical data compression in wireless sensor networks: a survey", *J. Netw. Comput. Appl.*, vol. 35, no. 1, pp. 37–59, 2012. [Online]. Available: <http://dx.doi.org/10.1016/j.jnca.2011.03.001>
- [2] L. Wang, J. Manner, "Evaluation of data compression for energy-aware communication in mobile networks", in *Proc. Int. Conf. on Cyber-Enabled Distributed Computing and Knowledge Discovery*, Zhangjiajie, 2009, pp. 69–76. [Online]. Available: <http://dx.doi.org/10.1109/cyberc.2009.5342172>
- [3] P. Prabu, N. G. Bhuvaneshwari Amma, G. Annapoorani, "Interfacing visible light communication with GSM networks to prevent the theft of the vehicle", *ICTACT J. Commun. Technol.*, vol. 6, no. 3, pp. 1136–1140, 2015.
- [4] O. Ordentlich, U. Erez, "Performance analysis and optimal filter design for sigma-delta modulation via duality with DPCM", in *Proc. IEEE Int. Symp. on Information Theory*, Hong Kong, 2015, pp. 321–325. [Online]. Available: <http://dx.doi.org/10.1109/isit.2015.7282469>
- [5] L. Robert, R. Nadarajan, "Simple lossless preprocessing algorithms for text compression", *IET Softw.*, vol. 3, no. 1, pp. 37–45, 2009. [Online]. Available: <http://dx.doi.org/10.1049/iet-sen20070106>
- [6] A. Said, "Comparative analysis of arithmetic coding computational complexity", in *Proc. Data Compression Conf.*, Snowbird, 2004, p. 562. [Online]. Available: <http://dx.doi.org/10.1109/dcc.2004.1281538>
- [7] R. J. Kozick, B. M. Sadler, "Source localization with distributed sensor arrays and partial spatial coherence", *IEEE Trans. Signal Process.*, vol. 52, no. 3, pp. 601–616, 2004. [Online]. Available: <http://dx.doi.org/10.1109/TSP.2003.822354>
- [8] S. Bjorklund, L. Ljung, "A review of time-delay estimation techniques", in *Proc. 42nd IEEE Conf. on Decision and Control*, Maui, 2003, vol. 3, pp. 2502–2507. [Online]. Available: <http://dx.doi.org/10.1109/CDC.2003.1272997>
- [9] C. I. Puryear, O. N. Portniaguine, C. M. Cobos, J. P. Castagna, "Constrained least-squares spectral analysis: application to seismic data", *Geophys.*, vol. 77, no. 5, pp. V143–V167, 2012. [Online]. Available: <http://dx.doi.org/10.1190/geo2011-0210.1>
- [10] J. T. VanderPlas, Z. Ivezić, "Periodograms for multiband astronomical time series", *Astrophys. J.*, vol. 812, no. 1, p. 18, 2015. [Online]. Available: <http://dx.doi.org/10.1088/0004-637X/812/1/18>
- [11] T. Zielinski, A. M. Moore, E. Troup, K. J. Halliday, A. J. Millar, "Strengths and limitations of period estimation methods for circadian data", *PLoS one*, vol. 9, no. 5, 2014. [Online]. Available: <http://dx.doi.org/10.1371/journal.pone.0096462>
- [12] B. S. Atal, M. R. Schroeder, "Adaptive predictive coding of speech signals", *Bell Syst. Tech. J.*, vol. 49, no. 8, pp. 1973–1986, 1970. [Online]. Available: <http://dx.doi.org/10.1002/j.1538-7305.1970.tb04297.x>
- [13] M. Malovic, Lj. Brajovic, Z. Miskovic, G. Todorovic, "Vibration measurements using a wireless sensors network", *Technics*, vol. 68, pp. 19–26, 2013.
- [14] J. Seo, J. W. Hu, J. Lee, "Summary review of structural health monitoring applications for highway bridges", *J. Perform. Constr. Facil.*, 2015. [Online]. Available: [http://dx.doi.org/10.1061/\(ASCE\)CF.1943-5509.0000824](http://dx.doi.org/10.1061/(ASCE)CF.1943-5509.0000824)
- [15] L. Svilainis, K. Lukoseviciute, V. Dumbrava, A. Chaziachmetovas, "Subsample interpolation bias error in time of flight estimation by direct correlation in digital domain", *Meas.*, vol. 46, no. 10, pp. 3950–3958, 2013. [Online]. Available: <http://dx.doi.org/10.1016/j.measurement.2013.07.038>
- [16] St. J. Dixon-Warren, "Motion sensing in the iPhone 4: MEMS accelerometer", *MEMS J.*, 2010, [Online]. Available: <http://www.memsjournal.com/2010/12/motion-sensing-in-the-iphone-4-mems-accelerometer.html>
- [17] I. Goel, S. Agarwal, R. Prasad, "A linear representation of Huffman tree", in *Proc. Int. MultiConf. of Engineers and Computer Scientists*, Hong Kong, 2012, vol. 1, pp. 317–319.
- [18] S. J. Cox, "Deflection of sleeper in ballast", *Vehicle Syst. Dyn.*, vol. 24, pp. 146–153, 1995. [Online]. Available: <http://dx.doi.org/10.1080/00423119508969621>