A fast compression algorithm for seismic data from non-cable seismographs

Fan Zheng
College of Instrumentation and Electrical Engineering
Jilin University
Changchun, China
zhengfan@jlu.edu.cn

Shufen Liu
College of Computer Science and Technology
Jilin University
Changchun, China
liusf@jlu.edu.cn

***Abstract*—Considering data storage characteristics of non-cable seismograph, accommodating to the requirement of further processing, a new parallel seismic data compression algorithm is developed based on the integer wavelet transform. First, separate the valid file header and extract the valid data to 4 one-dimensional matrixes. Then compress the four matrixes in parallel: the time signals are converted into transform domain by using the integer wavelet transform, where low frequency wavelet coefficients are retained and high frequency ones are scalar quantized. Lossless encoding is then conducted to deliver compression output. Experiments have shown that, at the same compression ratio, the proposed algorithm compresses 30% faster than traditional approach which directly uses two-dimensional wavelet transform to process two-dimensional matrixes, and yields good compression results.**

Keywords-non-cable seismograph; parrallel; wavelet; seismic data compression

I. INTRODUCTION

Seismic exploration exploits natural shocks or artificual shocks to collect seismic waves that reach the ground, using ground-mounted seismographs. It employs seismic data processing and analysis techniques to inverse underground geological structures. As the most effective geophysical exploration method, seismic exploration is widely used. In recently years, with the rapid development of mechanical, electronic, computer and telecommunication technologies, seismic exploration is shifting from traditional two-dimensional, small-scope operations towards three-dimensional, high-resolution, large-scale projects. This has directly led to rapidly increasing magnitude of seismic data storage. Network data transmission, data storage and backup, among other issues, have garnered more and more attentions in the development of seismic instruments and seismic exploration. Seismic data compression is an effective answer to that issue, and researchers have made some remarkable progress in this area [1-4].

Currently there are two popular types of seismographs: cabled telemetric seismographs and non-cable, memory-based seismographs. Telemetric seismographs transmit data simultaneously as it collects data, with a relatively small volume of data transmitted and stored. Researches in seismic data compression algorithm now focus mainly on seismic data collected by telemetric seismographs, and seismic data with standard format that have undergone through further processing. Telemetric seismographs are widely used; however, when it comes to the exploration with more than 10,000 tracks, cabled seismographs become very difficult to manage in terms of cables and equipment maintenance, and incur too high a cost. Therefore it is predicted by many experts that non-cable, memory-based seismograph is the way forward for the seismic instruments development [5]. Non-cable seismic instruments first store the data while continuing with data collection, then at a later time retrieve all the stored data. This creates a substantial amount of data that need to be stored and transmitted through network, therefore setting a tough requirement for compression quality and compression efficiency. The proposed algorithm is based on the non-cable, self-positioning seismograph developed by the author's laboratory, and the accompanying self-defined RAW file data memory format. A four-channel, parallel processing mode was adopted to ensure high quality of compression while boosting the compression efficiency.

II. ANALYSIS OF SEISMIC DATA IN NON-CABLE SEISMOGRAPH

The original seismic data files are typically very large given the long collection time and high sampling rate of non-cable seismographs. Moreover, when using hundreds or even thousands of seismographs to retrieve data, the data volume could be monumental. For example, if 250 seismographs work at the same time at a sampling rate of 2k for 24 hours, the data volume can exceed 643 GB. Therefore it is a great burden on both the storage side and network transmission side, consuming substantial resources on both sides.

Each lab-developed, memory non-cable seismograph is connected to four detectors, thus is able to conduct four-channel data acquisition. Data files can be saved in the self-defined format of RAW. As shown in Fig. 1, RAW file comprises a 512-byte file header and 4-track seismic data. The 512-byte file header consists of 32-byte effective part and 480-byte reserved part. The effective header contains the following information: file version, filter type, phase type, number of tracks, sampling length, sampling rate, time base, acquisition date, acquisition time, longitude and latitude, and more. Seismic data are original sampling

978-1-4673-4805-8/12/$31.00 ©2012 IEEE

values outputted by A/D converters, with a range of -6102081 ~ 6102081.

Our target of compression is the 32 byte valid file header and the seismic data in the RAW. The preserved header should be discarded, and the seismic data are to be decomposed into 4 one-dimensional matrixes. Transform coding is performed on each matrix to finalize the compression.

III. LIFTING INTEGER WAVELET TRANSFORM

To streamline the construction of wavelet and accelerate its computation, Sweldens at his bell laboratory designed a lifting framework to construct wavelet filters [6-7]. In his lifting scheme, the method of factorization was adopted to convert existing wavelet transform into several steps. Experiments showed that all wavelet transform constructed by finite-step filters are achievable by the finite-step lifting scheme. The fundamental of using lifting scheme for wavelet construction is the construction of predict operator and update operator. There are three steps in the lifting scheme: split, predict and update, as shown in Fig. 2. Integer wavelet transform [8] is done by rounding the product of the integer and the floating point number at every link of lifting.

IV. COMPRESSION ALGORITHM

By analyzing seismic data from non-cable seismographs we can see that each original seismic data file consists of 4 tracks of information, and the data have to be decomposed into 4 individual tracks later so that they can be properly analyzed and converted into other standard SEG formats. Therefore, an efficient way to do this is to process them individually. Thanks to the advanced computer technology, especially revolutionary innovation in CPU and the wide application of multi-CPU and multi-core technologies, now it is possible to design parallel compression algorithm based on multi-core technology to boost processing speed.

Integer wavelet decomposition was applied on the seismic data, after which we found that major information was concentrated in the low frequency band, while on the high frequency component a lot of useless noises resided. Low frequency coefficients then underwent several rounds of wavelet transform. A certain level of scalar quantization was applied on high frequency information so as to control the compression ratio. The resulting low frequency wavelet coefficients contained the majority of seismic reflective energies, and were directly coded in a lossless manner to preserve full accuracy for data reconstruction later. High frequency wavelet coefficients of all levels were treated with various scalar quantization, and then coded in a lossless way, hence wrapping up the compression process.

Detailed compression procedures for seismic data are as follows:

1. Separate the file header of the raw file; retain effective information in the header, cut off reserved bytes.

2. Extract seismic data into 4 tracks; store them in 4 one-dimension matrixes: T_1, T_2, T_3 and T_4.

3. Use the following steps to process the 4 matrixes in parallel:

3.1. Apply integer wavelet transform on matrix T, yielding low frequency wavelet coefficient matrix L and high frequency wavelet coefficient matrix H.

3.2. Perform another round of integer wavelet transform on L, obtaining low frequency matrix LL and high frequency matrix LH.

3.3. Retain LL, conduct scalar quantization on LH and H, obtaining LH_1 and H_1, respectively.

3.4. Losslessly code on LL, LH_1 and H_1, output compressed data T'.

4. Compress and integrate effective header.

Decompression is a reverse process of compression; therefore it can also be achieved in a parallel fashion. On each of the 4 tracks of compressed data, applying the following processes: decoding, de-quantization and inverse wavelet reconstruction. Flow charts of compression and decompression processes are shown in Fig. 3.

V. NUMERICAL SIMULATION

First, use lifting integer wavelet to extract one track of actual seismic data for experiment. Fig.4 shows the result of lossless compression. We can see that, by using integer wavelet transform and entirely lossless compression, we obtained reconstructed signals that are the same as original ones. We then scalar quantized the high frequency wavelet coefficients, increased compression ratio, yielding results as shown in Fig. 5. Now the waveform of reconstructed signals is almost identical with that of original signals, with only marginal errors.

A 4-track non-cable seismograph was adopted for field experiment. Sample at a rate of 2k, for a continuous duration of 100 seconds, obtaining 200,000 sampling points per track. To ensure best display quality of images, we present the first 4,000 points in a waveform as shown in Fig. 6, and compress them using the proposed algorithm. Fig. 6 shows the result at a compression ratio of 10:1. It is obvious from the graphs that signals before and after compression are basically the same. This indicates that our compression approach is feasible.

There are three ways to realize wavelet transform in seismic data compression: one-dimensional wavelet transform, two-dimensional wavelet transform, and three-dimensional wavelet transform. From the perspective of mean square error, at same compression ratio, the higher the dimension goes, the better the compression results are. However, on the computational side, higher dimensions will affect the efficiency of transform and coding. To illustrate the robust efficiency of the proposed algorithm, we compare it with a commonly used algorithm which uses two-dimensional wavelet to transform two-dimensional data matrix. A test was conducted using 8 groups of data, each group containing 10 seismic data files. Therefore a total of 80 data files were selected from real world seismic exploration. Each file was processed through 4 channels,

with 3,276,800 sample points in each channel. Sampling rate was 1k, all data files were 50mB in size. Under the same computing condition the above data were compressed in batches by both algorithms. Table 1 shows the time consumption of compression. We can see that the proposed algorithm performed remarkably better in compression efficiency: at the same compression ratio, it saves more than 30% processing time.

VI. CONCLUSIONS

The article proposes a parallel seismic data compression algorithm for non-cable seismographs based on integer wavelet transform. Unlike traditional compression approaches which rely on two-dimensional wavelet transform and two-dimensional data matrix, the proposed algorithm decomposes seismic data into 4 one-dimensional matrixes and compresses them in parallel. Experiment results have shown that, the proposed algorithm can compress seismic data from non-cable seismographs effectively; the parallel processing mode serves to significantly increase compression speed. The proposed algorithm adopts a multi-core, parallel processing approach. Works in the future may consider using GPU to process data at a greater scale in parallel, and the advantage may be even more manifested when it comes to mass seismic data processing.

REFERENCES

[1] Feng Z L et al, "Some practical aspects of seismic data compression based on wavelet transform", J Tsinghua Univ (Sci & Tech), vol. 41, 2001, pp. 170-173.

[2] Zhao L P, Xiao D G, "An efficient algorithm for large-scale volume data compression and its application in seismic data processing", Journal of Computer-Aided Design and Computer Graphics, vol. 21, 2009, pp. 1606-1611.

[3] Liu C, Wang P M, "Application of United data compression technique in seismic prospecting, Global Geology, vol. 25, 2006, pp. 434-439.

[4] Liu C, Wang P M, "The application of discrete cosine transform coding to seismic data compression, Journal of Jilin University (Earth Science Edition), vol. 34, 2004, pp. 277-282.

[5] Guo J, Liu G D, "Current situation and expectation of cable-less seismic acquisition system, Progress in Geophys, vol. 24, 2009, pp. 1540-1549.

[6] Sweldens W, "The lifting Scheme: A Construction of Second Generation Wavelets, SIAM Journal on Mathematical Analysis, vol. 29, 1998, pp. 511-546.

[7] Daubechies I, Sweldens W, "Factoring wavelet transforms into lifting steps, Journal of Fourier Analysis, vol. 4, 1998, pp. 247-269.

[8] Mallat S G, "A Theroy for multiresolution signal decomposition: the wavelet representation, IEEE Trans Pattern Analysisi and Machine Intelligence, vol. 11, 1989, pp. 674-693.

TABLE I. COMPARISON OF COMPRESSION EFFICIENCY OF PROPOSED AND TRADITIONAL ALGORITHMS

	1	2	3	4	5	6	7	8
Proposed	39.41	40.13	39.46	38.91	39.44	39.17	40.01	39.29
Traditional	56.59	60.91	59.84	54.41	59.68	57.28	59.94	58.65
Increase	30.36%	34.12%	34.06%	24.49%	33.91%	31.62%	33.25%	33.01%

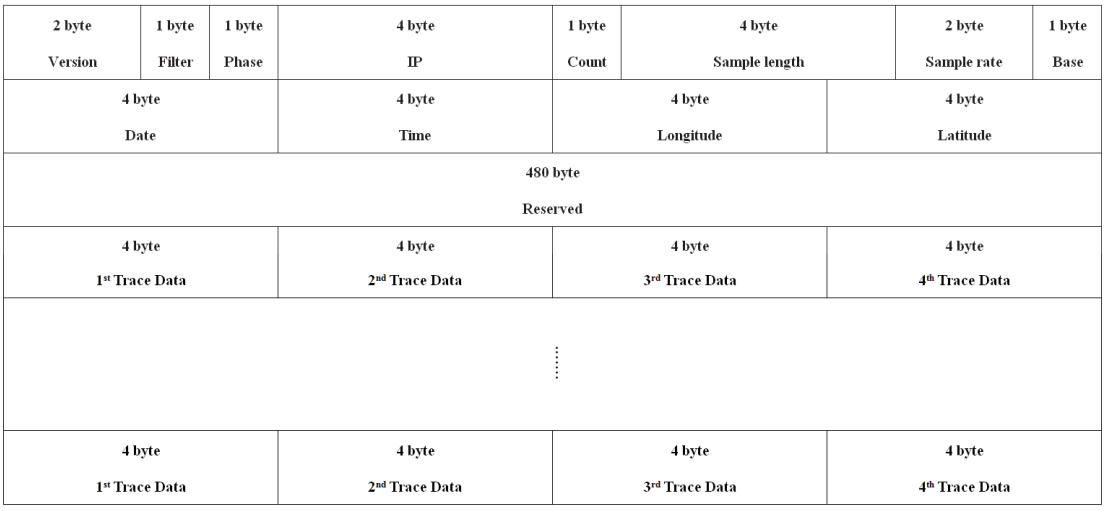

Figure 1. Raw structure

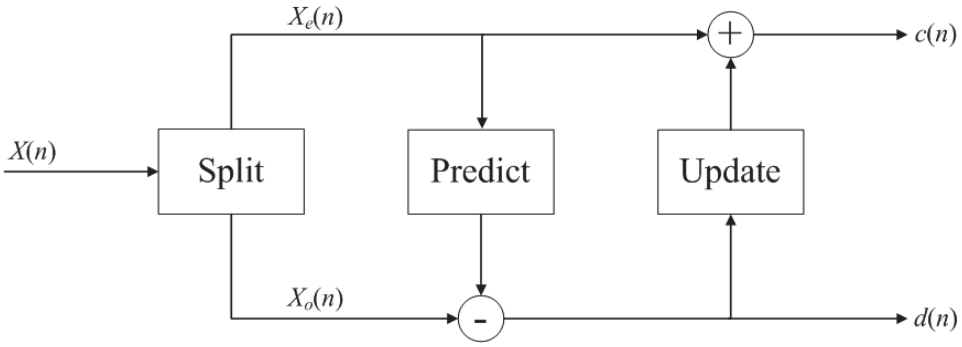

Figure 2. Wavelet lifting transform diagram

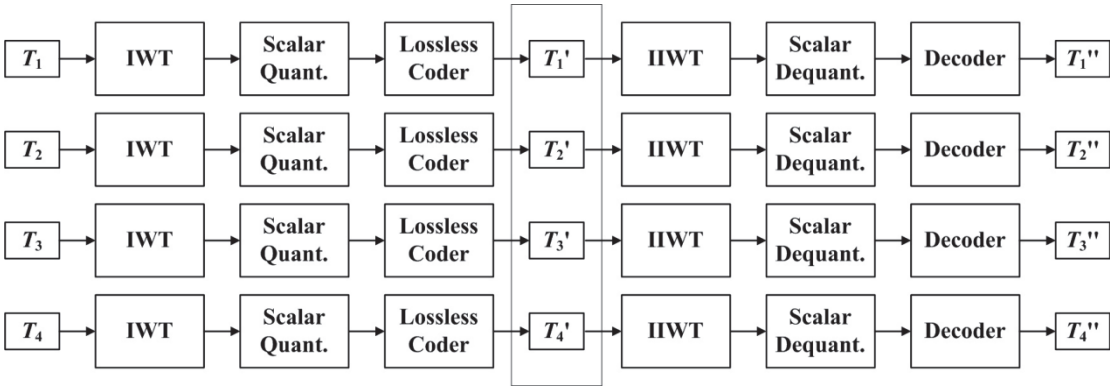

Figure 3. Algorithm structure

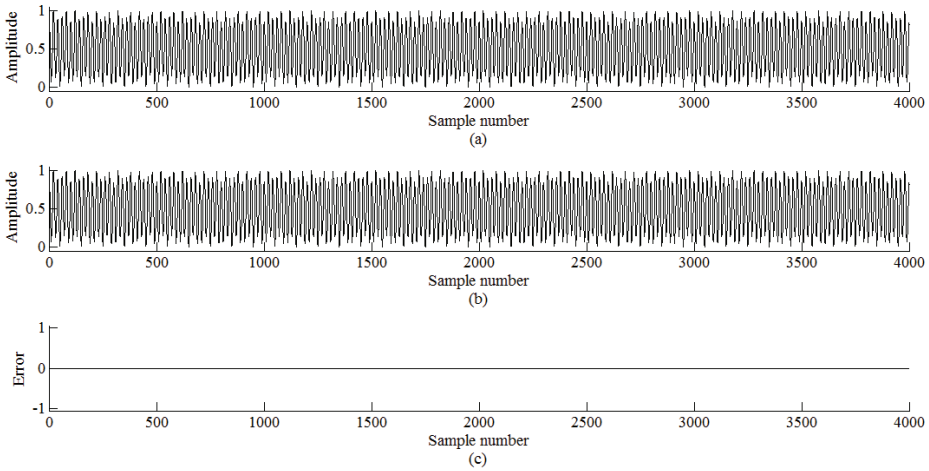

Figure 4. Original signal(a), reconstructed signal from lossless compression(b) and error(c)

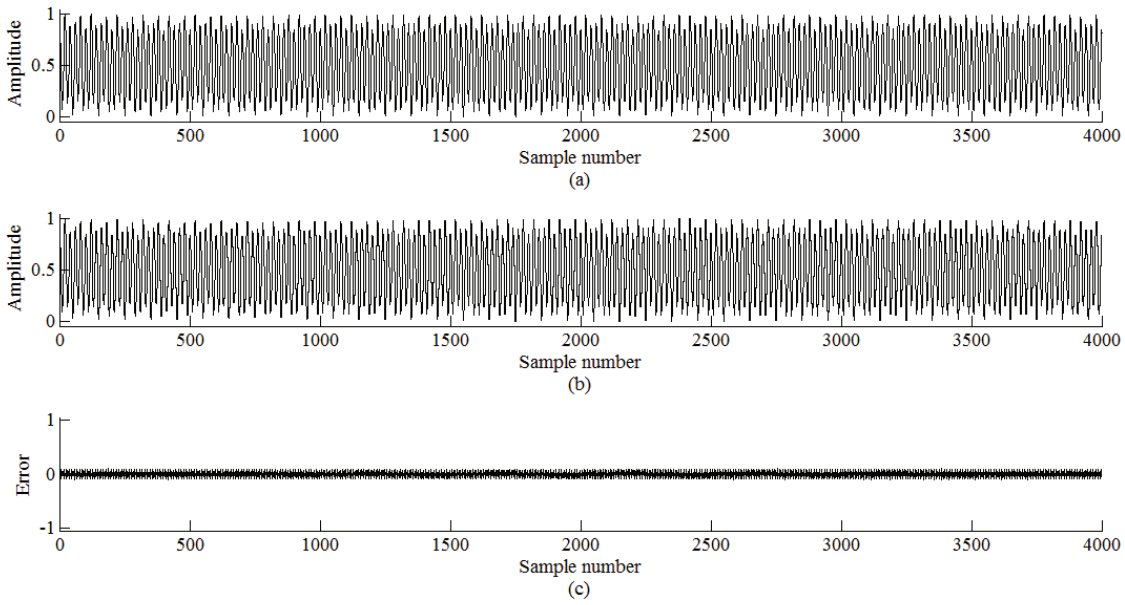

Figure 5. Original signal(a), reconstructed signal from loss compression at 6:1(b) and error(c)

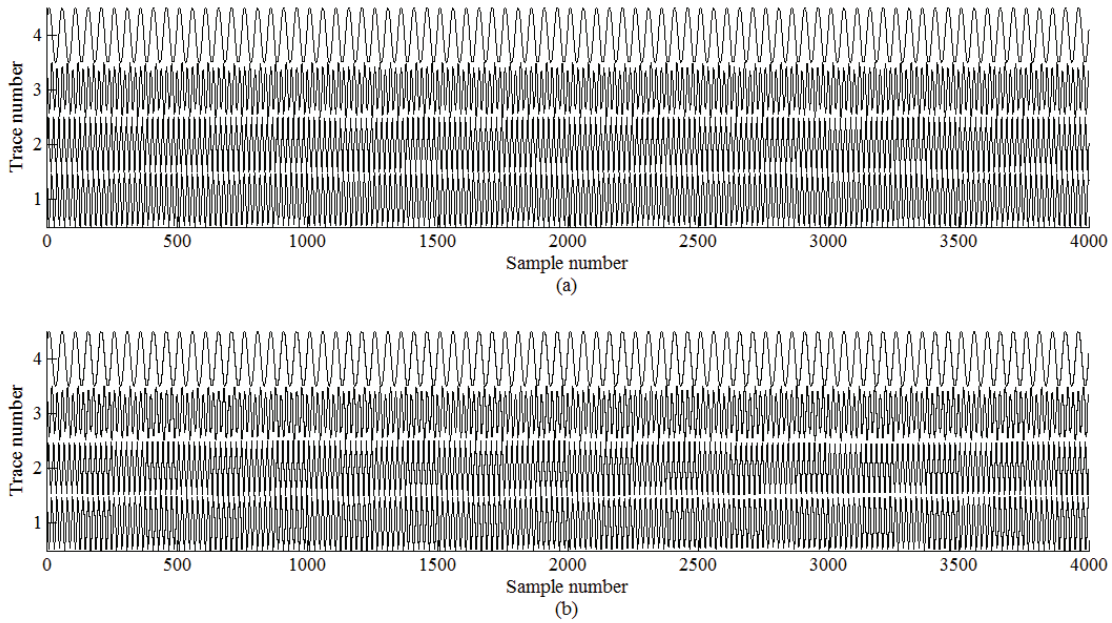

Figure 6. Original waveform(a) and reconstructed waveform at 10:1(b)