Data Compression for MingantU SpEctral Radioheliograpy

Long Xu, Yihua Yan, Jun Cheng
Chinese Academy of Sciences
Beijing 100012, China
Email: {lxu,yyh, chengjun}@nao.cas.cn

Lin Ma
Tencent AI Lab
Shenzhen, China
Email: lma@ee.cuhk.edu.hk

***Abstract*—In recent years, the high resolution, high precision and high sensitivity telescopes have been widely developed for astronomical observation in the world, such as SKA, Arecibo, ALMA and MingantU SpEctral Radioheliograpy (Muser). Along with these newly developed modern telescopes, a “big data” challenge was raised for data storage and processing. For example, Muser produces about 4.3TB raw data for daily solar radio observation, which presents a great challenge for archiving, analysing, transmitting and storage of data. In this paper, for the first time, we investigate data compression of Muser. We seek a lossless compressor, meanwhile, with good compression efficiency and affordable computational complexity**

I. Introduction

There has been a long history of studying image and video compression. Due to very strong continuity of image/video signal in spatio-temporal domain, image/video can be dramatically compressed to alleviate the burden of transmission and storage of data. For astronomical data, there is no general methods for compression/processing, so specific compression/processing method was developed for each device regarding the specific techniques of imaging and and specific characteristics of data. In radio astronomy field, aperture synthesis (AS) technique was widely used to deal with resolution limitation of single-dish antenna by synthesising a large number of small antennas. In AS, each pair of antennas composes an interferometer to record one Fourier component of the observed object. Multiple antennas would record a bunch of Fourier components, which could give an spatial object by recurring to inverse Fourier transform.

In theory, the correlation of two radio waves from the same radio source is the same as the Fourier coefficient of the radio source in spatial domain [1], which is the theoretical foundation of AS system. AS records the Fourier component instead of pixel intensity of a spatial object by using interferometry principle. These Fourier coefficients are then reconstruct a spatial image by using inverse Fourier transform. In AS, we also call the Fourier domain the UV-domain. The signal in UV-domain is called visibility function, and its spatial form is named brightness function.

The flexible image transport system (FITS) has been widely applied to astronomical data exchange and storage. It provides a general format for lots of astronomical data processing. However, it does not explore compression in depth. Recently, Hierarchical Data Format (HDF) [2][3] which was developed by the National Centre for Supercomputing Applications (NCSA) is becoming a popular format for storing astronomical data due to its well-defined and adaptable structure. It was designed to store and organize large amounts of numerical data, as well as compression, data access rate. HDF file format provides the options of data compression, where a number of compressors are provided as optional filter in HDF. In [9], different lossless data compression techniques were tested to find an optimal one to compress astronomical data obtained by the Square Kilometer Array (SKA). In [10], the authors presented a high-throughput data compression scheme for astronomical radio data that obtains a very high compression ratio. This compression algorithm was used in conjunction with the HDF5 to achieve 28% compression (lossy) ratio of its original size on CHIME Pathfinder telescope. In the context of radio astronomy, Peters and Kitae [11], [12] showed that the data format prescribed by the JPEG2000 standard can deliver high compression ratios with limited error on analysis of observational data. In [13], several data compression techniques were evaluated to seek minimal bandwidth usage, file transfer time, and storage space. The results suggested that JPEG2000 could be suitable for numerical datasets stored as gridded data or volumetric data.

The MingantU SpEctral Radioheliograpy (Muser) is a solar-dedicated radio telescope with the highest spatial, frequency and time resolutions in the world. It was developed by National Astronomical Observatories, Chinese Academy of Sciences (NAOC), located in Inner Mongolia and finalized in 2016. It consists of 100 antennas, arranging them in a three-arm spiral layout to form an AS system. Thus, Muser images the solar disk indirectly by using AS principle rather than directly capture spatial image like optical telescopes. The system framework of Muser is shown in Fig. 1. As mentioned above, the correlations of radio waves are collected from the interferometers of antenna array. This correlation data is remarkably different from spatial signal. The later is usually stored as gridded data or volumetric data and can be represented by image/video. From mathematical aspect, the correlation data is the Fourier coefficients of the spatial image of an object. Due to the limited number of antennas, the Fourier coefficients are very sparse, resulting in very blur image. At this moment, there is no study and well-developed system for handling archiving, analyzing, transmitting and storage of Muser data. In our previous efforts [14][15], we had investigated the imaging of Muser data. In this work, we contribute our efforts on Muser raw data compression for relieving the chanllege of transmitting and storage of scientific data. Referring to Fig. 1, raw data is the output of the correlator preceded by the digital receiver and followed by imaging

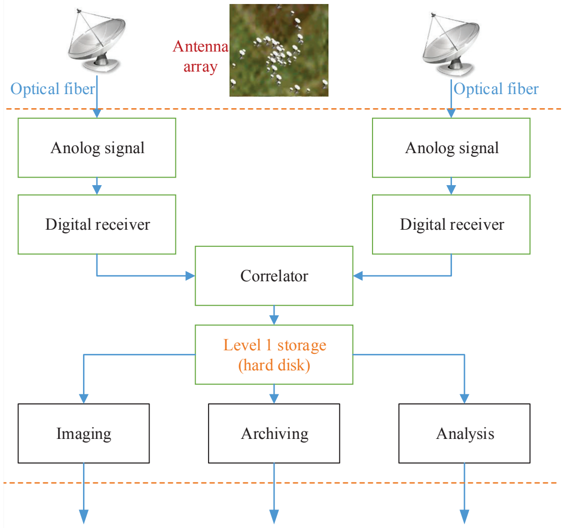


Fig. 1. System framework of Muser

system. In addition, we only investigate lossless compression in this work.

Muser monitors the Sun over more than 500 frequency channels and with time resolution of less than 200ms. At each frequency channel and each time slot, one time of observation records $n \times (n-1)$ sampling points of UV/Fourier domain. Expanding these sampling points into one dimension, along with two other dimensions, the raw data of Muser can be represented by a three dimensional cube, containing axes representing spectral frequency, correlation product, and time. The correlation product refers to the correlation of all antenna pairs, including auto-correlations from the same antenna and cross-correlations between different antennas. Muser has high spatial, time and frequency resolutions. Its data output rate is extremely high, about 4.3TB data for daily observation. Therefore, compression is highly desirable. Muser was developed by the team that I am affiliated with, and it is still in its infancy stage. It's the first time that we investigate data compression of Muser. Both project and research merits associated with this work are listed as the following.

a) Testing and evaluating the state-of-the-art compression algorithms on Muser data compression;
b) Multi-thread compression and programmings on Muser data are investigated and associated softwares are developed;
c) Statistical analysis of data: byte-based data are collected to compute the probabilities of all entries (0∼255), and then variable length coding (VLC) is implemented.

II. Muser

Muser is a radio telescope of AS. It was developed by National Astronomical Observatories, Chinese Academy of Sciences, located at Mingantu Station, Inner Mongolia, China. Muser uses aperture synthesis (AS) technique to image the Sun. AS synthesizes a group of small antennas to achieve the resolution of a big antenna. For a single-dish antenna system, assumed the diameter D, its resolution is given by $R = \lambda/D$, For an AS system, its resolution is still given by the formula for single-antenna system, while D is no longer the diameter but the maximum baseline length termed by the largest distance of two antennas. The maximum baseline length of Muser is 3km, so it can achieve a good resolution. Muser consists of 100 reflector antennas, which are grouped into two antenna arrays (Muser-I and Muser-II) for low and high frequency bands respectively. Muser-I is composed by 40 antennas with the diameter of 4.5 meter, and operated at 0.4-2GHz frequency band. Muser-II is composed by 60 antennas with the diameter of 2 meter, and operated at 2-15GHz frequency band. The frequency resolution of both Muser-I and Muser-II is up to 25MHz, so there are 64 channels for Muser-I and 520 channels for Muser-II, which is greatly superior to other existing radio heliographes in the world. Meanwhile, time sampling rate is up to 25ms for Muser-I and 200ms for Muser-II. This is also very competitive among contemporary radio telescopes. The detail specifications of Muser are tabulated in Table I.

AS devices record the Fourier components of observed objects instead of spatial images, where each two antennas compose of an interferometer to capture one Fourier component each time. Given n antennas, there would be $n \times (n-1)/2$ interferometers, which can record $n \times (n-1)/2$ Fourier components for each time of observation. By making use of the Earth's rotation, more Fourier components can be obtained. Nevertheless, only a small percent of Fourier components are recorded by AS, which results in blur synthesized images usually. A big challenge companying with AS is image reconstruction. Since very sparse Fourier components are captured by AS devices, an ill-posed problem is associated with image reconstruction of AS. We have investigated image reconstruction of Muser in [14][15]. In this work, our efforts are to compress data for saving transmission and storage costs of Muser, and possibly allow faster data access during data processing.

TABLE I. The Specifications of Muser

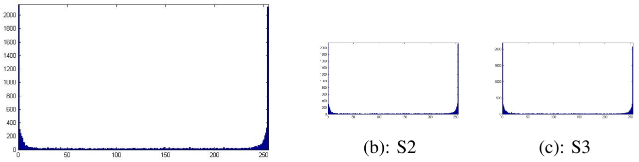
Frequency range	0.4-15GHz (I: 0.4-2GHz; II: 2-15GHz)
Frequency resolution	I: 64 channels; II: 520 channels
Spatial resolution	$1.4'' - 10.3''$; II: $10.3'' - 51.6''$
Temporal resolution	I: 25ms, II: 200ms
Dynamic range	25dB
Polarizations	Dual circular L, R
Number of antennas	I: 40 × 4.5m; II: 60 × 2.0m
Max baseline	3km
Field of view	$0.6^\circ - 7^\circ$

III. Data compression algorithm

Compression can greatly ease the burden of storing and handling large data sets. In case data is compressed in volume, the time required to load data from hard disk into memory also can be reduced. This is very beneficial for Muser data processing since the I/O of hard drive is a bottleneck for it. A single hard drive can typically be read at a rate of 100MB/s. Since of powerful computing ability of model computers, the speed of data processing, including compression, is sufficiently fast, even a modest number of processors is able to keep up with I/O rate. Even if this is not the case, data could be processed in parallel. A multi-threaded implementation of data processing can thereby result in a significant speed up on multi-core systems. Regarding decompression, one might wish to load long time of acquired data at once for analysis. Thus, it is desirable to perform decomposition as fast as possible. The speeding up of decomposition can also recur to multi-threaded implementation.

In this section, a quick overview of the state-of-the-art lossless compressors is first given as follows.

i) **Szip**: Szip [18] is an implementation of the extended-Rice lossless compression algorithm. It is reported to provide fast and effective compression, specifically for the NASA Earth Observatory System (EOS)[1];
ii) **Gzip**: GZIP [17] is a combination of Lz77 and Huffman coding and is based on the DEFLATE algorithm;
iii) **Lz4**: Lz4 [25] algorithm was developed by Yann Collet and belongs to the Lz77 family of compression algorithms. Its most important design criterion is simplicity and speed;
iv) **Lz77**: Lz77 and Lz78 [16] are the two lossless data compression algorithms. They are both theoretically dictionary coders. Lz77 maintains a sliding window during compression. The later input looks up already encoded symbols in the sliding window. And, offset-length pairs of new input are encoded. These two algorithms form the basis for many variations including LZW, LZSS, LZMA and others. Besides they formed the basis of several ubiquitous compression schemes, including GIF and the DEFLATE algorithm used in PNG;
v) **LZMA**: LZMA [19][20] uses a dictionary compression algorithm. It was first used in the 7z format of the 7-Zip archiver [21]. It is somewhat similar to the Lz77 and features a high compression ratio and a variable compression-dictionary size (up to 4 GB). LZMA2 is a simple container format that can include both uncompressed data and LZMA data, possibly with multiple different LZMA encoding parameters. LZMA2 supports arbitrarily scalable multi-threaded compression and decompression.
vi) **Huffman**: a Huffman code [22][23] is a particular type of optimal prefix code that is commonly used for lossless data compression. Statistic the probabilities of all symbols, the two nodes with the lowest probabilities are taken as the leaf nodes, sharing a parent node, and then the probability of the parent node is added into statistics. Repeat these steps until all the element has been added to the binary tree. Once Huffman tree is established, each leaf node can be represented by a series of "0" and "1" which indicate left and right path respectively. In this way, the leaf node with smaller probability would have longer path length, and vice versa;
vii) **Golomb**: Golomb coding [24] is a lossless data compression method using a family of data compression codes invented by Solomon W. Golomb in the 1960s. Golomb coding uses a tunable parameter M to divide an input value N into two parts: q, the result of a division by M, and r, the remainder. The quotient is sent in unary coding, followed by the remainder in truncated binary encoding. When M=1, Golomb coding is equivalent to unary coding. An exponential-Golomb (Exp-Golombcode) is a type of universal code. To encode any nonnegative integer x using the Exp-Golomb code: Write down x+1 in binary; Count the bits written, subtract one, and write that number of starting zero bits preceding the previous bit string.



(b): S2

(c): S3

(a): S1

Fig. 2. Histogram of byte-based data (16 frames are included for computing histogram; "S1": sequence "20130122", "S2": sequence "20140122", "S3": sequence "20151101")

Second, the statistical analysis of data is performed to result in a better VLC coding. Since VLC is dependent on the probabilities of coding symbols, we get statistics about the byte-based integers (0∼255) for each raw data file as shown in Fig. 2. As the file name suggests, (a) was recorded on Jan. 22, 2013, (b) was recorded on Jan. 22, 2014 and (c) was recorded on Nov. 1, 2015. In Muser, one-minute data are stored in a binary file, where all recorded syntax elements are byte-based. By statistical analysis, we found that there is a very sharp peak around 0 and 255 in the histograms. Therefore, VLC is highly superior to fixed-length coding. Without in-depth comparing and analyzing, Exp-Golomb coding is used to accomplish VLC in this work. Meanwhile, we noticed that the numbers going far away from 0 and 255 are distributed uniformly, so fixed-length coding is used on them. By this proposed method, both larger compression ratio and less computational complexity can be achieved.

IV. Experimental results

Since data format of radio telescope is much different from that of optical telescope which carries out imaging of an object directly in spatial domain, very successful visual signal (image/video) compression techniques are not applicable. In addition, usually lossless compression is mandatory for keeping raw data of observations. Therefore, lossless compression was widely investigated for radio telescopes [9]-[13]. We checked the compression performance of the popular lossless compression algorithms and the proposed one on Muser. The compression ratios are comparable, ranging from 50% to 70%.

The syntax elements of Muser are organized as shown in Table II. They are all saved in bytes in a data file. Since there is only less than 1% percent of header bits in a data file, we do not handle header bits and data bits separately.

TABLE II. Major syntax elements of Muser recorded data

Syntax	Length (Bytes)	Explain
Sequence header	8	GPS time
Cross-correlation	11352 (44 × 43 × 6)	44 antennas (including 4 redundancy antennas), each two antennas form an interferometer giving one Fourier component each time. One Fourier component is represented by a complex number, both real and imaginary parts are 3-bytes numbers.
Auto-correlation	176 (44 × 4)	44 antennas, each one has one auto-correlation coefficient.

Besides compression, data reorganization would benefit fast data access. Two adjacent channels are interleaved in current data format. Since ordering data with time/channel axis is most efficient for the majority of Muser data processing,

TABLE III. Compression performance comparison among all tested compression algorithms

Sequence	Szip	Gzip	Lz4	7z	Huffman	Golomb	Lz77
S1	60.69	67.28	69.46	57.24	66.59	67.17	77.23
S2	56.28	60.07	62.68	55.38	67.13	67.40	73.68
S3	57.40	60.83	63.26	58.29	69.06	69.21	72.20
Average	58.12	62.73	65.13	**56.97**	67.59	67.93	74.37

we deinterleave data of two adjacent channels to reorganize them along each channel. To accelerate encoding/decoding, we employ CWinThread of MFC to realize multi-thread programming. The implementing time is dramatically reduced comparing with single-thread procedure.

The compression efficiency is listed and compared in Table III. Among all compared compression algorithms, “Szip”, “Gzip”, “Lz4” and “7z” are all downloaded from Internet. “Huffman”, “Golomb” and “Lz77” are implemented by ourselves. These algorithms are all variable length coding by using short codeword representing symbol with large probability and long codeword representing the one with small probability. From Table III, it can be concluded that the proposed compression method is comparable with the state-of-the-art compressors.

The associated compression algorithms are all lossless. They can be categorized into two classes: VLC and dictionary based coding. VLC achieved such a big success before emergency of dictionary based methods, and was very popular for decades before dictionary based coding was developed. VLC is on the basis of information theory, namely Shannon information theory which claimed that the minimum coding length is bounded by Shannon entropy. The dictionary based methods compress a sequence by looking up dictionary which was built online or offline, consisting of already encoded words. With dictionary, we just need to indicate the index of an input word in compressed bitstream. It has been witnessed that dictionary-based algorithm took place of VLC, becoming mainstream of lossless compression. It should be pointed that such a big achievement partially comes from relevance, continuity and predictability of natural text, visual signal and language. In case of randomly distributed symbols for coding, dictionary-based algorithm is no longer competitive relative to VLC. That’s why we achieve the comparable compression efficiency (in Table III) for both VLC and dictionary-based method on Muser data.

Besides testing and evaluations on existing compression algorithm, we go further to explore the specific characteristics of Muser data so that data-specific encoder is proposed in this work. We analyze the byte-based distribution of Muser data, and observed that very high probability is for numbers around 0 and 255. Therefore, only a part of data are encoded by Golomb code, but others are still are encoded by fixed length encoding. This strategy speeds up encoding/decoding time dramatically. Loosely speaking, more than 50% computational complexity is saved comparing with purely Golomb-based encoding.

V. Conclusions

This paper investigated data compression of Muser. The state-of-the-art entropy encoders were tested and evaluated on Muser data. In addition, by analyzing probability distribution of Muser data, a new encoder specifically designed for Muser data is developed. It can achieve better encoding efficiency regarding both compression ratio and computational complexity. In the future, we will further study spatio-temporal prediction, transform and quantization coding techniques for Muser data to improve compression radio.

Acknowledgment

This work was supported by the National Natural Science Foundation of China under grants 61572461 and 11433006, CAS 100-Talents (Dr. Xu Long).

References

[1] J. A. Hőgbom, “Aperture Synthesis with a Non-Regular Distribution of Interferometer Baselines,” Astron. & Astrophys. Suppl., vol. 15, p. 417, Jun. 1974.

[2] www.hdfgroup.org/hdf5/whatishdf5.html

[3] J. Portell, E. Garcia Berroad, C. E. Sanchez, J. Castaneda, and M. Clotet, “Efficient Data Storage of Astronomical Data Using HDF5 and PEC Compression,” SPIE High- Performance Computing in Remote Sensing, Vol. 8183, 2011, Article ID: 818305.

[4] SKA Science Working Group, “The Square Kilometre Array Design Reference Mission: SKA-mid and SKA-lo”, report, v1.0, February 2010.

[5] http://www.naic.edu/general/index.php?option=com_content&view=article&id=151&Itemid=649

[6] http://www.almaobservatory.org/en/publications

[7] Y. Yan, “Radio imaging spectroscopy observations of the Sun in decimetric and centimetric wavelengths,” 2013 (IAUS).

[8] J. Du, Y. Yan, W. Wang, & D. Liu, “Image simulation for Mingantu Ultrawide Spectral Radioheliograph in the decimeter wave range,” Publications of the Astronomical Society of Australia, 2015, 32, 1-13

[9] Karthik Rajeswaran, Simon Winberg, “Lossless compression of SKA data sets,” Communications and Networks, vol. 5, pp. 369-378, 2013.

[10] K. Masui, M. Amiri, L. Connor, and et.al., “A compression scheme for radio data in high performance computing,” Astronomy and Computing, vol. 12, pp. 181-190, 2015.

[11] S.M. Peters, V.V. Kitaeff, “The impact of JPEG2000 lossy compression on the scientific quality of radio astronomy imagery,” Astronomy and Computing, vol. 6, pp. 41-51, Oct. 2014.

[12] V. V. Kitaeff, A. Cannon, A. Wicenec, D. Taubman, “Astronomical imagery: Considerations for a contemporary approach with JPEG2000,” Astronomy and Computing, vol. 12, pp. 229-239, Sept. 2015.

[13] D. Vohl, C. J. Fluke, G. Vernardos, “Data compression in the petascale astronomy era: A GERLUMPH case study,” Astronomy and Computing, vol. 12, pp. 200-211, Sept. 2015.

[14] L. Xu, L. Ma, Z. Chen, Y. Yan and J. Wu, “Perceptual Quality Improvement for Synthesis Imaging of Chinese Spectral Radioheliograph,” PCM2015, South Korea, Sept. 2015. DOI: 10.1007/978-3-319-24078-7_10, pp. 94-105

[15] L. Xu, L. Ma, Z. Chen, X. Zeng, and Y. Yan, “Perceptual Image Quality Enhancement for Solar Radio Image,” International Conference on Quality of Multimedia Experience (QoMex 2016), Lisbon, Portugal, Jun. 6-8, 2016.

[16] Ziv, Jacob; Lempel, Abraham, “A Universal Algorithm for Sequential Data Compression,” IEEE Transactions on Information Theory. 23 (3): 337C343. doi:10.1109/TIT.1977.1055714, May 1977.

[17] http://www.gzip.org/

[18] http://www.compressconsult.com/szip/

[19] http://7z.sparanoid.com/sdk.html

[20] http://zh.wikipedia.org/wiki/LZMA

[21] http://www.7-zip.org/

[22] http://blog.csdn.net/abcjennifer/article/details/8020695

[23] https://en.wikipedia.org/wiki/Huffman_coding

[24] https://en.wikipedia.org/wiki/Exponential-Golomb_coding

[25] http://fastcompression.blogspot.com/2011/05/lz4-explained.html