4.9 WEATHER DATA COMPRESSION

Christopher E. Steffen* and Ning Wang
NOAA Research - Forecast Systems Laboratory
Boulder, Colorado

1. INTRODUCTION

Significant reductions in weather data size without loss of any useful information can be accomplished through the use of data compression and rounding to the least significant digit. The source of the weather data's compression can be traced to the data's smoothness and rounding. Rounding off the insignificant digits and extracting the smoothness results in increased symbol frequency. Common applications such as gzip and bzip2 encode frequent symbols with less bits. Hence, more data may be sent over a fixed bandwidth channel if it is first compressed. This paper compares several compression techniques with respect to 12 kilometer weather data from the ETA model.

This issue has become more important with the advent of higher resolution model data, ensemble model data, and with the commissioning of the construction of new weather satellites using twelve bands instead of three. In addition, the fixed bandwidth of the Satellite Broadcasting Network (SBN) and limited disk space are also driving the interest in data compression. The primary products to be compressed include model data, radar data, and satellite image data.

Some ramifications of doubling the resolution of the model data are: 1) computation time increases by an order of magnitude, 2) the data volume quadruples, 3) the number of significant digits needed doubles, and 4) the data smoothness quality improves.

This paper addresses the application of common compression techniques to weather data. It begins with a short description of the important characteristics of gzip and bzip2. It continues with a description of the benefits of rounding prior to using gzip or bzip2. Next, it describes the benefits of applying a difference filter to extract the smoothness quality and how it improves the compression ratio when using gzip and bzip2. Next, there is a brief discussion of the benefits of decoupling the file specification from the compression technique and the existing file formats. The combined compression technique is applied to the ETA model data and the results are summarized in Table 3. A high level discussion of Wavelet compression with an error grid is then given along with the results which are summarized in Table 3. The reduction in "bandwidth used" is then summarized in Table 2 demonstrating a significant improvement in the compressibility of the ETA model data.

* Corresponding author address:
NOAA/FSL, 325 Broadway,
Boulder, CO 80305-3328
Email:steffen@fsl.noaa.gov

2. GZIP / BZIP2

The algorithm in GZIP tries to map long strings into short symbols. The algorithm in BZIP2 try to represent the most frequently occurring symbols with the least number of bits. Both algorithms are lossless. GZIP uses LZ77, (Ziv et. al.1977). BZIP2 uses a Burrows-Wheeler Transform, (Burrows et. al. 1994) and (Nelson 1996), along with adaptive Arithmetic encoding, (Bell et. al. 1990) and (Nelson 1991).

The compression ratios achieved on weather data by both algorithms can be significantly improved by preprocessing the data. The two preprocessors used were rounding and difference filtering.

3. ROUNDING / QUANTIZING

The term "rounding data" refers to truncating the precision of the data. In essence, throwing away information which may only be conveying to the end user the amount of noise in the data. Care must be taken to remove this information correctly. In order improve the compression by GZIP and BZIP2, the rounding must occur in a way that increases the frequency of some symbols. For instance, "rounding" sets more binary digits on the right side of the mantissa of a floating point number to zero. Since this occurs in every floating point number, it reduces the number of symbols possible.

Simply multiplying by a power of 10 and rounding to decimal place and then dividing by the power of 10 will not optimize the compressibility of the data. The reason is that the data are stored in a binary format which is a base two notation and it must be rounded in base two. One method to eliminate this problem is to multiply by powers of two. However, if it is desirable to maintain the rounding as a power of 10, one can calculate the number of binary digits required to maintain this precision and mask off the unnecessary binary digits. During reconstruction multiply by the power of ten, round, and divide by the power of 10 to reconstruct the power of 10 precision accurately. In Table 3, "z + 1" was used in place of z to give a little extra precision. Below the z+1 bit, one additional bit was used to specify whether to round up or down. In the future, these two extra bits may be able to be eliminated.

x = Original Number,

y = Decimal Digits of Precision,

p = Number of Binary Bits of Precision,

q = Rounded number,

r = Reconstructed number,

$$p = \lceil \log_2(10^y) \rceil,$$

$$q = \lfloor x * 2^p \rfloor / 2^p,$$

Reconstruct with,

$$r = \lfloor q * 10^y + 0.5 \rfloor / 10^y,$$

$$\text{Error} = x - r.$$

Currently, GRIB1 rounds the data to some precision. It is preferable to avoid losing information GRIB1 currently contains since some of the weather model output may be used to initialize local weather models. Weather models tend to be overwhelmed by the amplification of small errors. Hence, only algorithms that lose less information than GRIB1 are acceptable.

The seventh and eigth rows of Table 3 demonstrate the advantage of using a rounding in addition to a difference filter prior to compression with BZIP2. For instance, in the case of temperature, the rounding algorithm improves the compressibility by a factor of 5.7. (ie. 163.4% / 28.7% = 5.7)

The term "quantizing" is typically used in conjunction with a continuous signal which must be represented with a discrete signal. The quantization process maps a range of the continuous signal to a single level in the discrete signal essentially rounding the infinite precision continuous values into a discrete value.

4. WEATHER DATA SMOOTHNESS

"Weather data" typically refers to a parameter defined over a geographical area at a particular time. More often than not, two nearby points have similar values, so there is a high positive correlation between parameter values which are both near geographically and near in time. This would imply that some of the information is redundant, which means that it can be extracted while increasing the frequency of occurrence of some of the symbols. This results in improved compression once GZIP or BZIP2 are applied.

One method of extracting the smoothness is by using the difference filter specified by Paeth (1991). The Paeth filter uses the data points to the left, to the upper-left, and above the current data point to estimate the value of the current data point. The estimate is then subtracted from the current data point to remove the smoothness. If labels a, b, c, and d are applied to left, upper-left, above, and current data points, respectively, the algorithm estimate equals "a + b - c". It is also possible to use the value above or to the left of the current value as an estimate. The pattern used in the difference filter algorithm used here is shown in Table 1. To reconstruct the original, the estimate must be added back to the values.

Table 1. Estimates subtracted from array element.

	Left	Left	Left	Left
Above	Paeth	Paeth	Paeth	Paeth
Above	Paeth	Paeth	Paeth	Paeth
Above	Paeth	Paeth	Paeth	Paeth
Above	Paeth	Paeth	Paeth	Paeth

The sixth and seventh rows of Table 3 demonstrate the advantage of using a difference filter to remove the smoothness of the data prior to compression with BZIP2. For instance, in the case of temperature, the difference filter algorithm with BZIP2 improves the compressibility by a factor of 1.28. (ie. 209.3% / 163.4% = 1.28)

5. DECOUPLING FILE FORMAT FROM COMPRESSION ALGORITHM

One of the major impediments to implementing a new compression algorithm is the existing file formats. Ideally, the file would consist of a meta-data block and a data block. The meta-data block would contain parameters describing the data. The data block would contain an array of data and the meta-data describing the array format.

This file architecture is attractive for the following reasons: 1) GZIP or BZIP2 could be applied to the meta-data block. 2) An algorithm which takes into account the rounding and smoothness could be applied to the data block. 3) The meta-data format is decoupled from the compression algorithm used. 4) The compression code can be removed from the file format encoder/decoder code and placed in a post/preprocessor module. This would also make it easy to change compression algorithms if the algorithm used violates a patent.

6. EXISTING FILE FORMATS

The National Weather Service (NWS) RADAR (NWS 2002) and GRIB1 (Grid in Binary) (NWS 1997) specifications define complex fielded file formats. Both define the compression algorithm to be used as an integral component of the file format. This means that the World Meteorological Organization (WMO) and NWS Radar Group must support compression standards along with the file format standard even though there are other organizations which specialize in supporting compression standards. This approach is justified if the compression ratio of the internal standard is better than the compression ratios of the external standards, or if the data to be compressed are significantly different from the data compressed by existing compression standards. However, since this situation can change over time, it is still preferable to separate the file format standard from the compression standard.

7. TIME CONSTRAINTS

Compression of the data must be done in real time and as close to the source of the data as possible for maximal benefit. For instance, the satellite data must be compressed and disseminated within one minute receipt.

8. 2-DIM / ROUND / DIFFERENCE / BZIP2

In the Round/DifferenceFilter/BZIP2 tests, the binary rounding algorithm from section 2 was applied to eliminate the unnecessary information. Next, the Paeth filter (Paeth 1991) was used to extract the smoothness of the data while increasing the frequency of some symbols. Then BZIP2 was applied to encode frequently occurring symbols with fewer bits.

9. WAVELET COMPRESSION/ ERROR GRID

Wavelet compression typically consists of three steps. The first step is to wavelet transform the data, (Daubechies 1988 and Cohen et. al.1992), which applies an orthogonal (or biorthogonal) transform to the data to compact the information into fewer coefficients. The second step is to quantize the transformed coefficients, (Shapiro 1993) which finds the optimal bit allocation that minimizes the mean squared error of the quantized coefficients. The third step is to encode the data with some sort of entropy encoder, (Bell et. al. 1990). This, in essence, is a lossless data compression, driven by a finite context model that further reduces the size of the compressed data file.

Wavelet compression typically gives higher compression ratios. However, coefficient domain quantization makes it difficult to directly control the errors in the L^∞ norm. The L^∞ norm acts as a measure of the peak error. How to efficiently minimize the L^∞ error within the coefficient domain is still an ongoing research topic. Here we use a rather simple scheme for our experiment which is similar to "lossy plus lossless error residual coding", (Rabbani et. al. 1991). We first minimize the L^2 error using a coefficient domain quantization. This is equivalent to minimizing the mean squared error. We then reconstruct the dataset from the compressed data stream and compare the reconstructed dataset with the original dataset to produce an error grid. Since this error grid is very sparse, we can simply entropy encode it and append it to the compressed data stream. The decoder receives this compressed stream, reconstructs the dataset and uses the information contained in the error grid to eliminate the errors that exceeds the maximum error.

Like most compression techniques, wavelet compression compresses smooth/ correlated data better than random data. In order to achieve optimal compression, different compression ratios are needed to control the size of the error grid for different datasets. In addition, a different compression ratio may be needed for the same dataset at a different time of day or different time of year. We can predetermine a set of compression ratios for the datasets, or generate the ratios automatically using some statistical methods.

10. EVALUATING THE RESULTS

A problem in evaluating different compression algorithms is that they all generally compress one type of data better than the others. Therefore, each dataset must be compressed to see which one does the best on average. Consistency of the compression ratio is also important, since it indicates that the algorithm is more likely to work with a larger dataset and that the test case is less likely to be attributable to coincidence. Notice in Table 3 that the GRIB1/BZIP2 method is not as consistent as 2D / Round / Difference / BZIP2. This could indicate that the dataset is not large enough.

11. BANDWIDTH USAGE

Compression can have a positive effect on the bandwidth necessary to transmit a given amount of data. Table 2 summarizes the reduction in bandwidth needed to transmit the same data that are currently transmitted via GRIB1.

Table 2. Bandwidth Usage
* "lossless is lossless with respect to GRIB1's current rounding level

Compression Technique	Bandwidth Usage (% of GRIB1)
GRIB1	**100%**
2D / Round / Difference / BZIP2	**31%**
Wavelet ("lossless"*, quantization, error grid)	**8% to 15%**

12. SUMMARY

Significant improvements in bandwidth utilization can be realized through the use of common compression techniques such as Wavelet/ErrorGrid or Round / Difference / BZIP2 compression. Significant improvements in the ease with which updates to the latest compression algorithm occur can be realized by decoupling the file format from the compression algorithm.

An area of future investigation is to address the problem of setting the compression ratio in the wavelet compression algorithm automatically for different data types. Unfortunately, the size of the error grid may vary dramatically with small changes in the compression ratio. Another area of investigation is to extend the algorithms to three dimensions.

13. ACKNOWLEGMENTS

The authors wish to thank the other members of the Integrated Work Team on Data Compression. Especially, Mary Erickson, Steve Gilbert, Ralph Peterson, Howard Carney, and Barbara Hickman.

Table 3. Comparison of compression algorithms with respect to GRIB1's compresssion.

Method \ Data Precision	height 3 bits 39 grids	precip 2 bits 1 grid	relh 0 bits 39 grids	temp 3 bits 39 grids	uwind 3 bits 39 grids	vertvel 3 bits 39 grids	vwind 3 bits 39 grids
GRIB1 Size	43808 Kbytes	889 Kbytes	24571 Kbytes	30795 Kbytes	32735 Kbytes	21823 Kbytes	32654 Kbytes
Wavelet/ Error Grid			14.4% of GRIB1	7.3% of GRIB1			
GRIB1 / BZIP2	60.3 % of GRIB1	11.9 % of GRIB1	26.3 % of GRIB1	22.7 % of GRIB1	31.8 % of GRIB1	16.2 % of GRIB1	33.4 % of GRIB1
Floats	230.3% of GRIB1	291.0% of GRIB1	410.6% of GRIB1	327.6% of GRIB1	308.2% of GRIB1	462.3% of GRIB1	309.0% of GRIB1
2D/Bzip2	145.8% of GRIB1	118.0% of GRIB1	353.3% of GRIB1	209.3% of GRIB1	288.8% of GRIB1	436.7% of GRIB1	292.5% of GRIB1
2D/Paeth /Bzip2	104.8% of GRIB1	175.8% of GRIB1	333.9% of GRIB1	163.4% of GRIB1	261.8% of GRIB1	436.7% of GRIB1	272.0% of GRIB1
2D/Rnd/ Paeth/Bzip2	30.3% of GRIB1	19.2% of GRIB1	39.8% of GRIB1	28.7% of GRIB1	33.8% of GRIB1	31.7% of GRIB1	34.1% of GRIB1
2D/Gzip	172.6% of GRIB1	117.6% of GRIB1	352.5% of GRIB1	245.6% of GRIB1	280.3% of GRIB1	426.7% of GRIB1	285.4% of GRIB1
2D/Paeth /Gzip	145.3% of GRIB1	174.9% of GRIB1	348.4% of GRIB1	220.3% of GRIB1	269.5% of GRIB1	427.1% of GRIB1	274.9% of GRIB1
2D/Rnd/ Paeth/Gzip	52.6% of GRIB1	29.4% of GRIB1	62.9% of GRIB1	44.3% of GRIB1	53.3% of GRIB1	49.5% of GRIB1	55.7% of GRIB1

14. REFERENCES

Bell, T. C., J. G. Cleary, I. H. Witten, 1990: *Text Compression*, Prentice Hall Inc, NJ, ISBN 0-13-911991-4

Burrows, M., Wheeler, D.J., 1994: A Block-sorting Lossless Data Compression Algorithm, *Digital Systems Research Center Research Report 124*,

Cohen, A., I. Daubechies, J. C. Feauveau, 1992: Biorthogonal Bases of Compactly Supported Wavelets, *Comm. PAM*, **XLV**, 485-560.

Daubechies, I. 1988: Orthonormal Bases of Compactly Supported Wavelets, *Comm. PAM*, **XLI**, 909-996.

National Weather Service 1997: *GRIB1* Edition 1. http://weather.unisys.com/wxp/ Appendices /Formats/GRIB.html

National Weather Service 2002: Radar File format. *2620001C* Interface Control Document for the RPG to Class 1 User. Formerly called *2620001B* Interface Control Document for the RPG/Associated PUP Open Build 1.0

Nelson, M., 1996: Data Compression with the Burrows-Wheeler Transform, *Dr. Dobb's Journal*, September.

Nelson, M., 1991: Arithmetic Coding + Statistical Modeling = Data Compression Part 1 - Arithmetic Coding, *Dr. Dobb's Journal*, February.

Nelson, M., 1991: Arithmetic Coding + Statistical Modeling = Data Compression Part 2 - Statistical Modeling, *Dr. Dobb's Journal*, February.

Paeth, A.W., 1991: Image File Compression Made Easy in *Graphics Gems II*, J. Arvo, Ed. Academic Press, San Diego. ISBN 0-12-064480-0. See Filters in PNG Technical Specification.

Rabbani, M., P. W. Jones, 1991: Digital Image Compression Techniques, Bellingham, WA: *SPIE*,

Shapiro J. M, 1993: Embedded Image Coding Using Zerotrees of Wavelet Coefficients. *IEEE Trans. Signal Processing*, 41, 12, 3445-3462.

Ziv, J., A. Lempel, 1977: A Universal Algorithm for Sequential Data Compression, *IEEE Trans. on Information Theory*, May.