On Data Latency and Compression

Joseph M. Steim, Edelvays N. Spassov,

Kinemetrics, Inc.

Abstract

Because of interest in the capability of digital seismic data systems to provide low-latency data for "Early Warning" applications, we have examined the effect of data compression on the ability of systems to deliver data with low latency, and on the efficiency of data storage and telemetry systems. Quanterra Q330 systems are widely used in telemetered networks, and are considered in particular.

Q330 data acquisition systems transmit data compressed in Steim2-format packets. Some studies have inappropriately associated the use of data compression with necessarily increased system latency. When the amount of time duration represented per packet is fixed and the physical length of the packet is variable (as in the Q330), rather than a fixed physical size of the packet, latency is defined and can be arbitrarily small. Latency is a function of the number of samples represented in a data packet, not whether the data volume used to represent those samples is compressed. A robust method[5] to measure latencies arising in the Q330 family of data loggers, shows a typical mean latency <0.82s over the public internet using cellular connections, and <0.65s on an Ethernet LAN. For a given number of samples in a data packet, compression *reduces* delay because fewer bits are transmitted to represent a fixed time duration. Higher levels of compression produce further improvements in system and communications efficiency. A figure of merit is illustrated representing the performance of several compression techniques.

Background

Level 1, 2 ("Steim1" and "Steim2")

The Level 1 method, introduced in 1984, codes first differences as groups of 8, 16, or 32-bits. A `bit map' preceding each group of fifteen 32-bit words gives the decoder the information necessary to reconstruct the original samples. The detailed description of the Level 1 and 2 coding appears in the SEED Manual[4]. Level 1 and 2 have become de facto standards for much of the data in IRIS DMS and FDSN archives.

Level 3

In 1990, a third, increased level of compression was proposed[3] that further improved compression efficiency of the Level 1 and 2 techniques by including:

- spanning two 32-bit groups to allow a representation of 5 consecutive 12-bit differences and 3 20-bit differences; representing up to 9 consecutive 3-bit differences; and using the previously-reserved bit map code 0 to represent 2 16-bit differences.
- Adaptively using either 1st or 2nd differences (by frame if desired), coding the difference in the sign bit of bit-map word 0, which maps to itself, and was therefore not used.

- replacement of common compression bit-map patterns with one-byte codes, liberating up to about 5% additional space, called "flag squeezing".

11	32				1 32-bit difference
10	16	16			2 16-bit differences
01	8	8	8	8	4 8-bit differences
	32 bits				

map	LEVEL 2 & 3									
10	01	30								1 30-bit difference
10	10	15	15							2 15-bit differences
10	11	10	10	10						3 10-bit differences
01	8	8	8	8						4 8-bit differences
11	01	5	5	5	5	5				6 5-bit differences
11	00	6	6	6	6	6	6			5 6-bit differences
11	10	00	4	4	4	4	4	4	4	7 4-bit differences
	32 bits									

map	LEVEL 3											
00	16	16										2 16-bit diff
11	11	000	3	3	3	3	3	3	3	3	3	9 3-bit diff
11	11	001	NOT A DIFFERENCE (NAD)									NAD
11	11	1	29-bit SAMPLE									29-bit sample
10	00	12	12	6 hi	00	6 lo	12	12				5 12-bit diff
10	00	20	10 hi	01	10 lo	20						3 20-bit diff
	64 bits											

Code Available

A public-domain C library has been available at the Quanterra user's group archive[3] to document and illustrate the compression and decompression of all Levels.

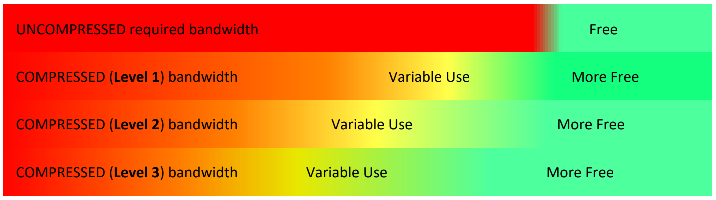
Framing

While the compression technique is often associated with 64-byte "frames" as used in MSEED, the methods are not limited to fixed 64-byte frames. Q330 data loggers, for example, employ Level 2 coding in variable-length packets with a constant time delay.

Discussion

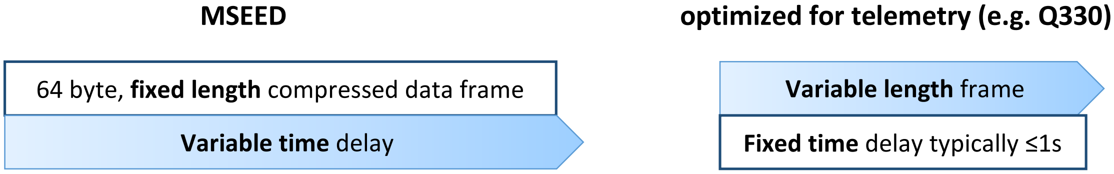
Compression and Latency related to Origin Queuing[5]

Plausible data compression methods reduce the volume (number of bits) required to represent each data sample. Compression of data therefore always reduces the minimum required bandwidth (bits/sec) for transmission compared with transmission of uncompressed data. The volume required to transmit compressed data is variable, depending on characteristics of the data. Higher levels of compression require less minimum bandwidth and volume, and maximize the available unused ("free") bandwidth for transmission of data queued at the source ("Origin Queuing" described by [5]). In general, latency caused by queuing is therefore *reduced* by compression because more bandwidth is available to transmit queued data.

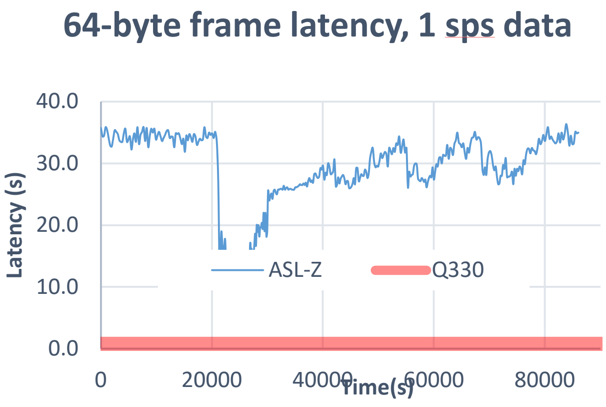

UNCOMPRESSED required bandwidth
Free
COMPRESSED (**Level 1**) bandwidth
Variable Use
More Free
COMPRESSED (**Level 2**) bandwidth
Variable Use
More Free
COMPRESSED (**Level 3**) bandwidth
Variable Use
More Free

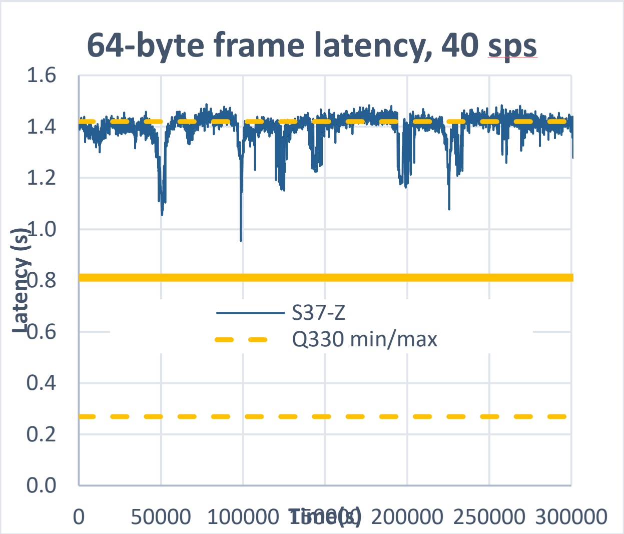
Packetization Aperture[5] Latency

Where compressed data are transmitted in frames of a fixed number of bytes, compression can increase the latency of transmitted data because a variable number of samples occupy a given volume. A data packet representing a fixed number of samples, however, is transmitted with a constant, arbitrarily small latency. Compression reduces latency in this case, because fewer bits/sec are transmitted.


MSEED
64 byte, **fixed length** compressed data frame
Variable time delay
optimized for telemetry (e.g. Q330)
Variable length frame
Fixed time delay typically ≤1s

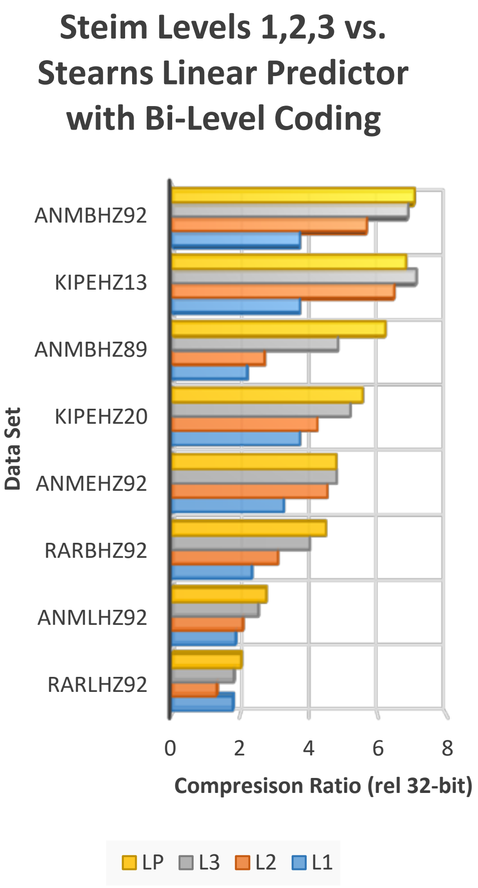
Figures below show examples of latency variation using 64-byte data frames over time for time series at 1sps and 40sps compared with the constant min, max, and 0.81s mean latency of a Q330S. Note that events result in *lower* latency because of reduced compressibility.



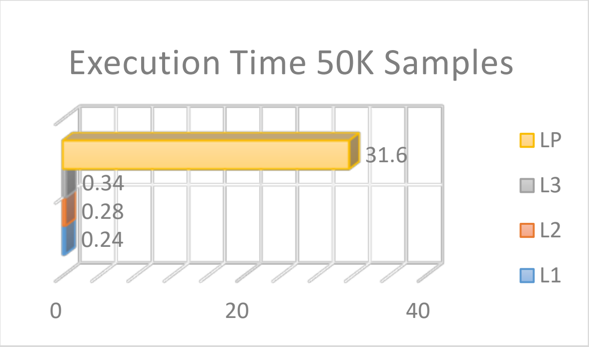


Compression

Steim1 and Steim2 compression methods are designed to exploit the redundancy of low-entropy 1D time series data (such as seismic, acoustic). The object of these compression methods is to remove as much redundancy as possible, leaving a maximally entropic residual. A technique was developed by Stearns[1] in the 1990's to maximally compress seismic data using linear prediction (LP) and efficient residual coding. The Level 1,2 (and 3) methods are compared using various data samples to the LP method. While LP compression is generally always more effective, L3 compares favorably, with *execution time roughly 100x less*, and much simpler all-integer arithmetic.

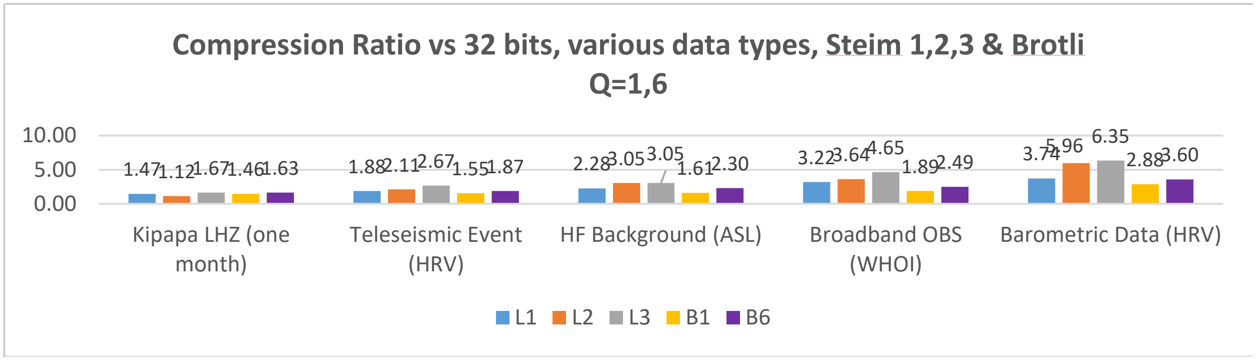


Comparison of compression efficiency and execution times of Level 1,2, and 3 Steim compression with the Stearns[1] linear predictor for various samples of data representing varying compressibility.



More Compression

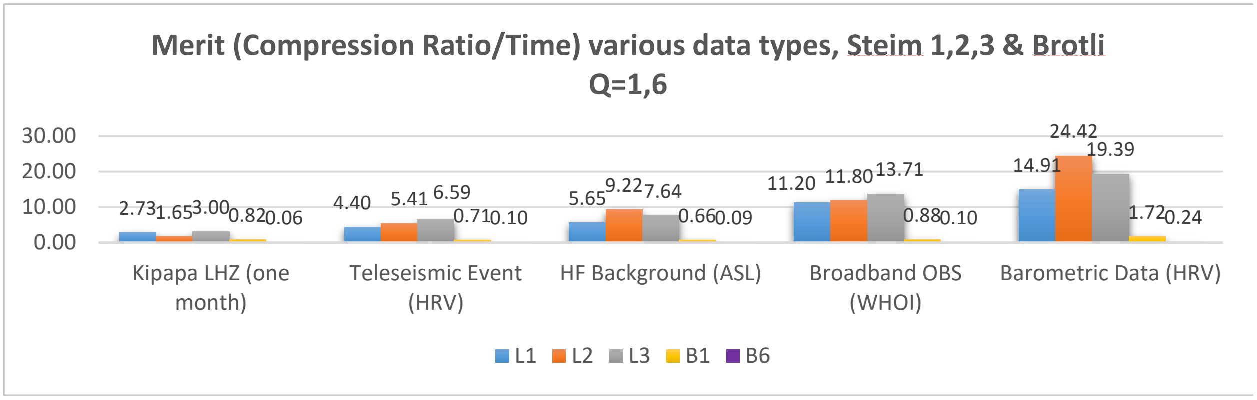
Lossless Lempel-Ziv (LZ) compression methods (e.g. gzip, 7zip, and more recently "brotli" introduced by Google for web-page compression) are often used for lossless compression of binary and text files. How does the best example of modern LZ methods compare with FDSN Level 1 and 2 for waveform data? Segments of data representing up to one month of various levels of compressibility were compressed using Levels 1,2 and 3, and brotli using "quality" 1 and 6. The compression ratio (relative to the original 32 bits) shows that in all cases the compression using L1,L2, or L3 is significantly greater than brotli.



An important metric of compression is resource (e.g. cpu, memory) usage. A figure of Merit (on a 200MHz ARM processor) can be defined to combine compression efficiency and resource usage as

Merit = 32/(average bits per compressed sample)/(execution time per 100k samples)

The Merit values for the same five different types of data in the figure above, in increasing (L-R) order of compressibility is shown below. Note that the generic brotli compressor is up to 100x less efficient in this measure. Resource usage is particularly important in a data center where simultaneous operations multiply resource demands. When memory usage is considered, LZ compression merit is further reduced.



Conclusions

- In a packetized data system, a component of data latency is defined by the number of samples per data packet. A fixed number of samples per packet results in a fixed packetization aperture[5] latency.
- Compression may *reduce* latency in a system that transmits a defined number of samples per data packet, because the number of bits required to represent those samples is reduced by compression.

- Compression increases available unused ("free") bandwidth in telemetry applications, *reducing* latency caused by Origin Queuing[5]. Higher levels of compression demand less minimum bandwidth.
- Low-complexity compression methods such as Level 1,2,3 are designed to exploit the redundancy of low-entropy 1D time series data (such as seismic, acoustic). A "Merit" value combining compression efficiency and resource utilization can be defined. The Merit of these low-complexity methods for 1D time series data is orders of magnitude greater compared to generic lossless compute-intensive dictionary-based compression methods (e.g. gzip, brotli, and other LZ algorithms).
- Level 2 compression offers high efficiency and low complexity, suitable for most 1D low-entropy time series. Additional efficiencies are available in a further Level 3 compression extending Level 2.

References

1. S. D. Stearns, L. Z. Tan and N. Magotra, "Lossless compression of waveform data for efficient storage and transmission," in IEEE Transactions on Geoscience and Remote Sensing, vol. 31, no. 3, pp. 645-654, May 1993.
2. J. Alakuijala, Z. Szabadka, "Brotli Compressed Data Format, draft-alakuijala-brotli-11",2016, http://www.ietf.org/id/draft-alakuijala-brotli-11.txt (this Internet Draft is a "work in progress" and does not represent a specification)
3. Steim, J. M., "Steim Compression", 1994, http://www.ncedc.org/ftp/pub/quanterra/steim123.ps.Z, http://www.ncedc.org/ftp/pub/quanterra/steim123.tar.Z
4. IRIS, 2012, "SEED Reference Manual, SEED Format Version 2.4"
5. Steim, J. M., and Reimiller, R. D. (2014). Timeliness of data delivery from Q330 systems, Seismol. Res. Lett. 85, no. 4, 844–851, doi: 10.1785/0220120170.