FAULT AND DISTURBANCE ANALYSIS CONFERENCE
Arlington VA Nov. 5-8, 1996

An Efficient Zero-Loss Technique for Data Compression of Long Fault Records

R.V. Jackson, G.W. Swift
Alpha Power Technologies
Winnipeg, Canada

Background

The value of data compression ranges from the obvious (reduction of storage capacity required) to the not so obvious (reductions in transmission time, reductions in response time). Typically, the extra computation time associated with the reconstruction of original data from a compressed format is small in comparison with the reductions in communication time due to compression.

Lossy compression is appropriate where the difference between pre-compression and post-compression data is acceptably small (i.e. imperceptible), or is a consequence of imposed bandwidth or storage limitations, and is frequently applied to visual image data and sound representations. Lossless compression allows the exact reconstruction of the original data; advantage is taken of some predictable characteristic(s) of the data to increase the likelihood that successful compression will occur.

In the case of transient data recording, it is imperative that the original data be reproduced exactly; thus a lossless algorithm is appropriate. One of the characteristics of transient power-system data that can be used to advantage is its generally large component in the power system fundamental frequency.

Binary Batching

A typical digitizer (analog to digital converter) used in power system applications delivers a binary value somewhat shorter than 16 bits. To facilitate computation within the attached computer, the sign bit is

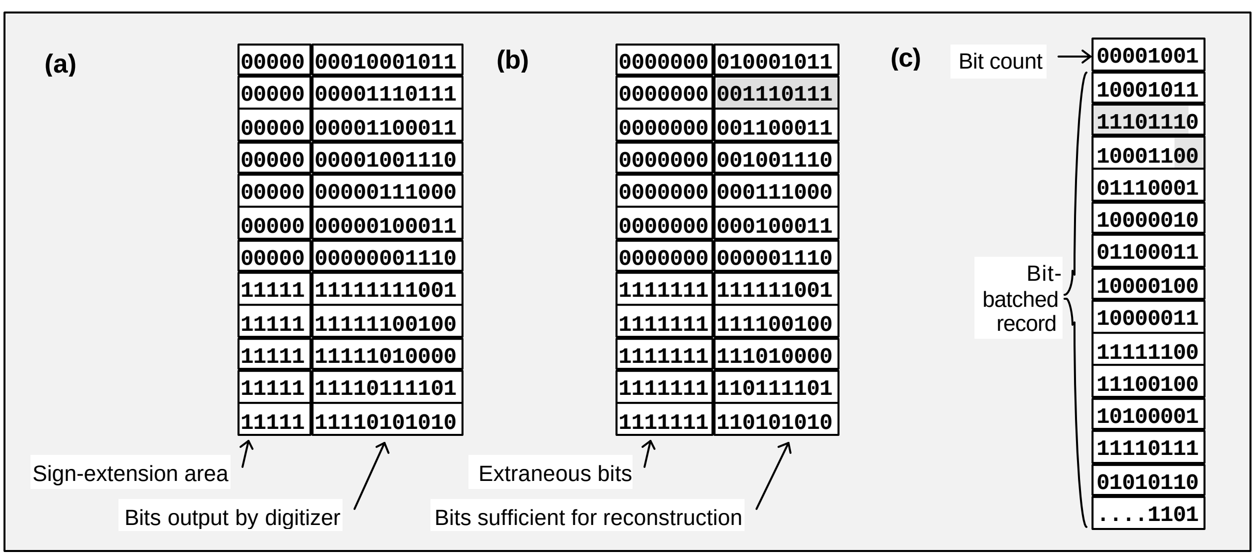

Figure 1: *Sample record fragment: (a) original bits from digitizer; (b) processor prescans the data, determines that nine bits are sufficient for reconstruction; (c) resulting recording, arranged bytewise.*

propagated towards the most significant end to a full word, which these days is almost always 16 bits wide as indicated in Figure 1(a). The extra bits generated in this manner represent no information content. One sign bit (in addition to the data bits) contains sufficient information for the purposes of reconstruction.

The recorder accumulates the sample values for each channel over a time interval, then scans (prescans) the sample data to determine the number of bits actually required to represent the data from the positive and negative maxima for the interval. See Figure 1(b). The recorder removes the extraneous bits by placing the sample values end to end bitwise (ignoring byte and word boundaries), then transmits the encoded data preceded by the bit-count. See Figure 1(c). Such a procedure ensures that the messages thus produced contain sufficient information to reconstitute the original sample values.

The compression advantage of binary batching is especially noticeable when the digitized samples are small. In typical installations, a significant number of channels are scaled so that the digitizer rarely exceeds a somewhat narrow range of values around zero. This is especially true of current inputs, which are generally scaled so that the digitizer will not saturate for expected fault current levels. The fact that the recording system is expected to reproduce fault current levels faithfully does not preclude it from taking advantage when the sample values are small compared to the range of the digitizer.

Prediction

The foregoing description takes advantage of relatively small numbers to reduce the bulk of recorded data without sacrificing precision. A prediction technique reduces the size of the generated data even further.

In this technique two consecutive (known) sample values for a channel are used to "predict" a third sample value. See Figure 2. The first two values are presumed to lie on a sinusoid at the nominal power system frequency. An estimate of the third sample can be computed as

$$s_m = 2\, s_{m-1} \cos \omega T - s_{m-2},$$

where ω is the power system frequency in radians per second,
and T is the sample interval in seconds.

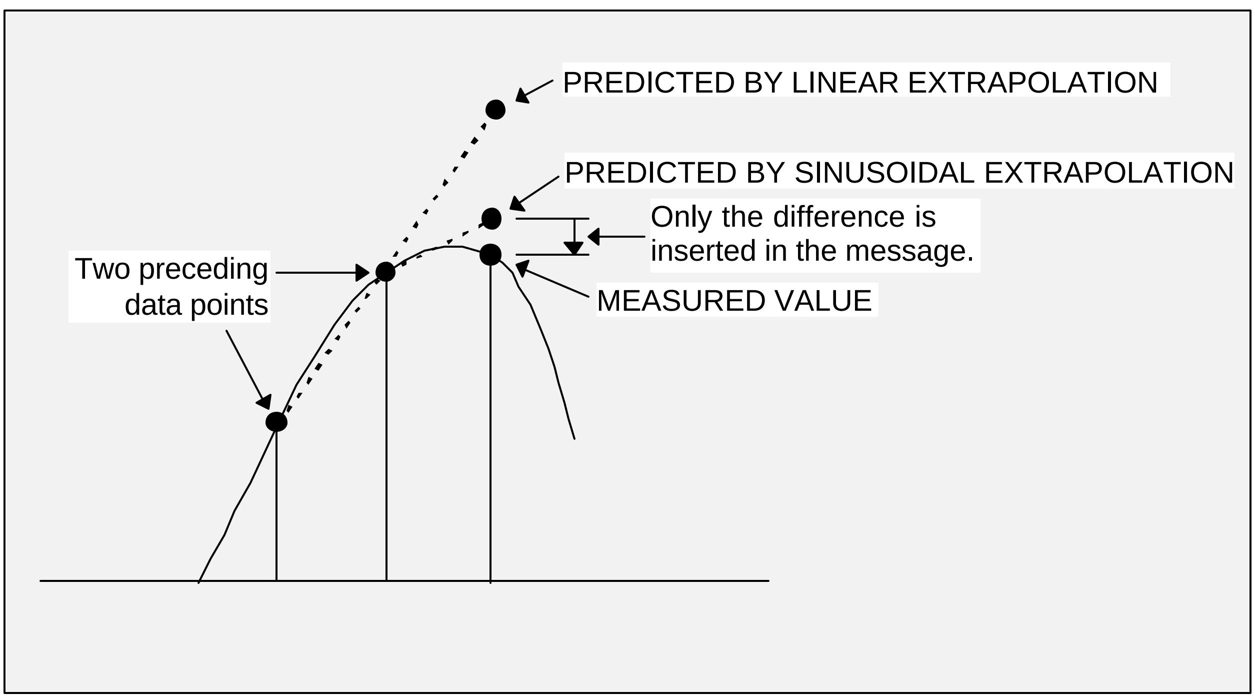

Figure 2. *Illustration of the linear and sinusoidal algorithm methods of data point prediction.*

Sample values (decimal)	Predicted values	Differences	Data inserted into message
139			0000000010001011
119			0000000001110111
			00000010 (bit count)
99	99	0	00
78	77	-1	11
56	57	-1	11
35	34	1	01
14	14	0	00
-7	-7	0	00
-28	-27	-1	11
-48	-48	0	00
-67	-67	0	00
-86	-85	-1	11

Table 1: *The sample fragment of Figure 1 compressed using the sinusoidal prediction method. The differences are inserted as twos complement binary format.*

(The derivation of this formula is given in the Appendix.)

The recorder copies the first two sample values to the recording, then computes and transmits the difference between actual and predicted sample values for the remainder of the samples in the time interval. For each sample a new prediction is made from the preceding actual sample values. If the signal has a strong component close to the fundamental frequency, the computed difference values are small in comparison to the actual sample values. By performing a pre-scan and transmitting bit-batched difference values (instead of bit-batched sample values as before), even more effective compression can be achieved, with the cost of a minor increase in processing at the receiving end (Table 1). Even ignoring the possibility of compressing the first two sample values and the bit count, the reduction in bits generated is remarkable (60 vs. 116).

An interesting sidelight is the case where zero is substituted for ω, the assumed power-system frequency, in which case the predictor formula reduces to

$$s_m = 2\, s_{m-1} - s_{m-2},$$

which some may recognize as the formula for linear extrapolation. Such a predictor could be employed for input channels which are known to be non-sinusoidal.

Example

A typical fault transient recording is shown in Figure 3. This could be, for example, the fault current for the phase in which a single-line-to-ground fault has occurred.

The signal has been sampled and quantized, i.e. rounded off to the nearest integer value.

Note the three sections of this recording:

- Pre-fault: a low amplitude near-sinusoid.
- Transient: a high amplitude ‘noisy’ signal (here represented arbitrarily as damped fifth harmonic, and a damped dc exponential decay).

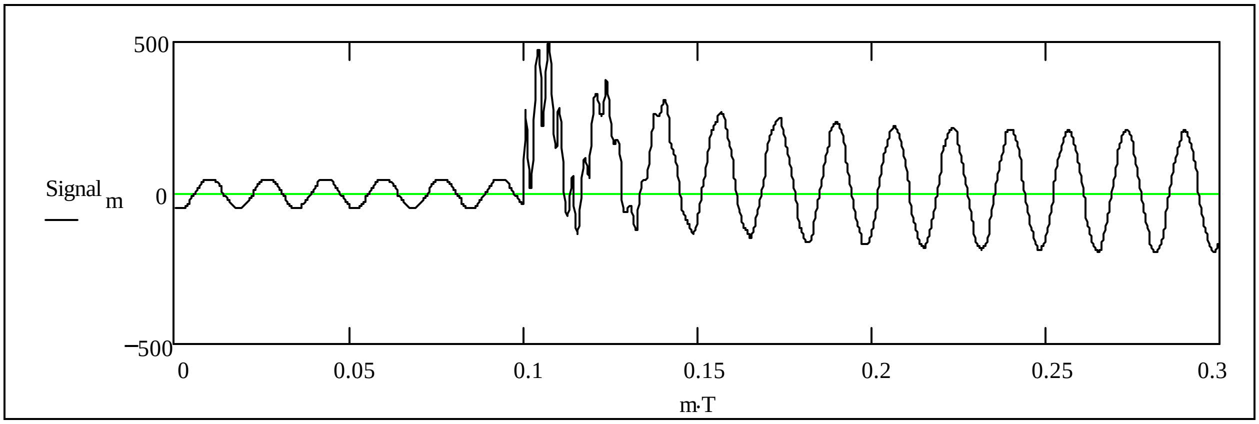

Figure 3. *Typical fault transient record, the basis for the investigation of various alternative compression strategies.*

- Post-transient: a high amplitude near-sinusoid.

It is apparent that the first and third sections are especially amenable to data compression on the basis of sinusoidal prediction, because of the relatively large component in the fundamental frequency.

Results

Variables considered:

- Sampling rate: 8 samples/cycle and 96 samples/cycle (based on 60 Hz)
- Signal frequency: At nominal: 60 Hz and off-nominal: 70 Hz.
- Linear and sinusoidal prediction algorithms.
- Compression of the signal itself and compression of the 'delta signal.'

The compression is performed on blocks of data, in quarter-cycle blocks. In other words, in any block of data, the largest (absolute value) data sample is found, and compression applied equally to all samples in the block. This is to reduce 'message overhead.'

For 96 samples/cycle this implies 24-sample blocks, and for 8 samples/cycle, 2-sample blocks are implied (perhaps too small a block size for efficiency, in this case).

Figure 4 shows the number of bits necessary to store the data in each block **h** for the fault record.

Upper curve: Bit-batched without prescan: 12 bits (including sign bit) for all data samples.

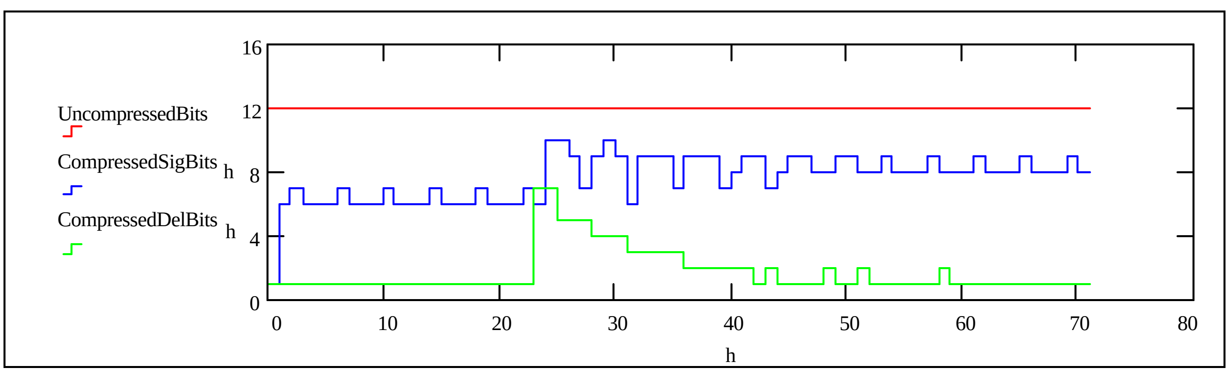

Figure 4: *Compression comparison, bits required for data storage, at 96 samples per cycle. Each horizontal step represents a block of 24 samples. Upper curve: Bits required for bit-batched sample data without prescan; Middle curve: Bits required to represent bit-batched sample data with prescan; Lower curve: Bits required for compressed 'delta' data, using prediction algorithm.*

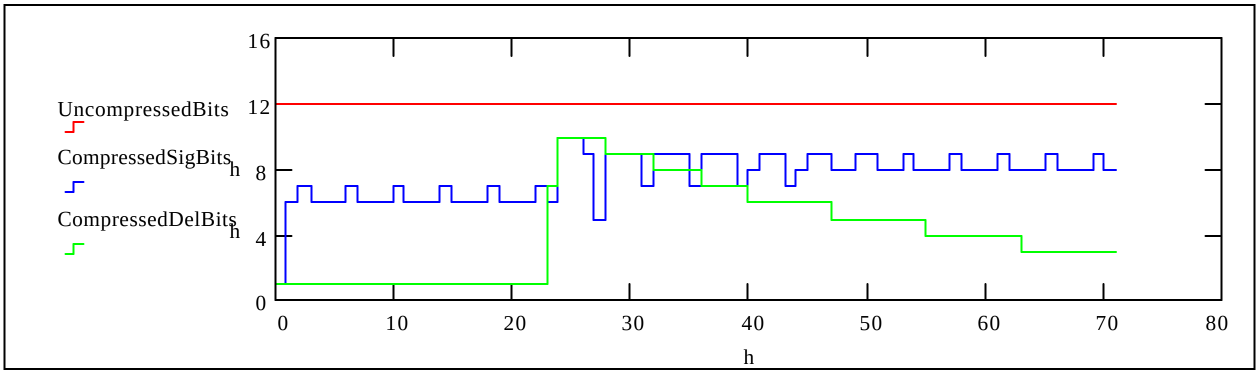

Figure. 5: *Compression comparison, bits required for data storage, at 8 samples per cycle. Each step represents a block of 2 samples. Upper curve: Bits required to represent bit-batched sample data without prescan; Middle curve: Bits required to represent bit-batched sample data with prescan; Lower curve: Bits required for compressed 'delta' data, using prediction algorithm.*

Middle curve: Bit-batched with prescan: the sample values in the message are represented using the number of bits necessary to represent the largest numerical sample in the message.

Lower curve: Difference data only (actual minus predicted), bit-batched, with prescan.

The parameters for Figure 4 are: 96 samples/cycle; sinusoidal predictor; 60 Hz signal.

Note that since normal data word storage is 16 bits, and the assumed analog-to-digital converter range is 12 bits, bit-batching alone has created one-quarter saving in data storage space. The compressions listed below do not include this factor.

Figure 5 shows a second example, at a much-reduced sampling rate: 8 samples/cycle; sinusoidal predictor; 60 Hz signal.

Tabulated results:

The "Compression Ratio" is defined here as the number of bits required to store the compressed data in binary form, divided by the number of bits required for a fixed storage size, namely 12 bits including sign for the whole record. The message overhead is assumed small and is ignored for comparison purposes.

This amounts to a comparison of the areas under the curves in Figures 4 and 5.

Sample Rate (samples/cycle)	System Frequency, (Hz)	Compression Ratio		
		Original Samples, Bit Batching with Prescan	Prediction Method, Linear Extrapolation	Prediction Method, Sinusoidal Extrapolation
96	60	1.6	6.0	6.9
96	70	1.6	6.0	6.1
8	60	1.6	1.9	2.8
8	70	1.6	1.8	2.2

Table 2: *Tabulated results for various scenarios. The middle column identifies the value of the pre-scan in determining the bit count pre sample point; the fourth column evaluates the use of linear extrapolation, the fifth evaluates sinusoidal extrapolation as a predictor.*

The results are shown in Table 2.

Conclusions

1. Compression ratios of the order of 6 or 7 are achievable for fault transient records of the nature studied in this document. The ratio would be higher, of course, for records with longer near-sinusoidal content, as for example channels recorded simultaneously on unfaulted phases.
2. A prediction/difference method appears to have a significant advantage over simple reproduction of the samples from the digitizer.
3. Although the sinusoidal algorithm is frequency-dependent, it performs better than the linear extrapolation method in all cases considered, even at an extreme of 70 Hz. The slight edge shown by the sinusoidal algorithm in compressing actual fault data will be improved in the accompanying data recorded for channels less affected by the fault.
4. For high sampling rates, say 96 samples per cycle, the advantage in using the sinusoidal algorithm rather than the linear one is reduced in the presence of a strong random component. The dynamic choice of prediction/compression method can also be considered.

Appendix : Derivation of the Sinusoidal Predictor Algorithm

It is assumed that the bulk of the data to be stored is *nearly* sinusoidal, and 60 Hz. If sinusoidal, any value at point m can be predicted from prior samples at points m-1 and m-2, where the spacing between the samples (in the time domain) is uniform.

Let m = index of sample value being predicted.
X_m = m^{th} sample.
Y_m = estimate of m^{th} sample.
ω = radian frequency, e.g. $2\pi 60$ radians per second.
P = arbitrary phase angle.
T = sample interval.

The compression algorithm is

$$y_m = 2 \cos \omega T \, x_{m-1} - x_{m-2}$$

<u>PROOF</u>:

The left-hand-side is

$$LHS = y_m = \cos(m\omega T+p),$$

representing a sinusoid of arbitrary phase and magnitude unity without affecting the generality of the result.

The right-hand-side is

$$\begin{aligned} RHS &= 2 \cos \omega T \, x_{m-1} - x_{m-2} \\ &= 2 \cos \omega T \cos[(m-1)\omega T + p] - \cos[(m-2)\omega T + p] \end{aligned}$$

Expanding, using $\cos(a+b) = \cos a \cos b - \sin a \sin b$, we get

$$\begin{aligned} RHS &= 2\cos \omega T \cos(m\omega T-\omega T+p) - \cos(m\omega T-2\omega T+p) \\ &= 2\cos \omega T \cos(m\omega T-\omega T+p) - \cos(m\omega T-\omega T+p-\omega T) \\ &= 2\cos \omega T \cos(m\omega T-\omega T+p) - \cos(m\omega T-\omega T+p) \cos \omega T + \sin(m\omega T-\omega T+p) \sin \omega T \end{aligned}$$

$= \cos \omega T \cos (m\omega T - \omega T + p) - \sin \omega T \sin (m\omega T - \omega T + p)$

$= \cos (\omega T + m\omega T - \omega T + p)$

$= \cos (m\omega T + p)$

= LHS.

Q.E.D.