NOAA Technical Memorandum OAR PMEL-134

COMPRESSION OF MOST PROPAGATION DATABASE

Elena Tolkova[1]

[1] Joint Institute for the Study of the Atmosphere and Ocean
Seattle, WA

Pacific Marine Environmental Laboratory
Seattle, WA
March 2007

UNITED STATES
DEPARTMENT OF COMMERCE

Carlos M. Gutierrez
Secretary

NATIONAL OCEANIC AND
ATMOSPHERIC ADMINISTRATION

VADM Conrad C. Lautenbacher, Jr.
Under Secretary for Oceans
and Atmosphere/Administrator

Office of Oceanic and
Atmospheric Research

Richard W. Spinrad
Assistant Administrator

NOTICE

Mention of a commercial company or product does not constitute an endorsement by NOAA/OAR. Use of information from this publication concerning proprietary products or the tests of such products for publicity or advertising purposes is not authorized.

Contribution No. 3051 from NOAA/Pacific Marine Environmental Laboratory

Also available from the National Technical Information Service (NTIS)
(http://www.ntis.gov)

Contents

List of Figures

Compression of MOST Propagation Database

Elena Tolkova[1]

Abstract. The MOST Propagation Database consists of approximately 1000 file triplets, representing time series for wave height, meridional currents, and zonal currents in a modeled tsunami caused by each of 804 unit earthquakes (tsunami sources) in the Pacific and 194 in the Atlantic Ocean. The data represents a 24-hour-long evolution of a tsunami with 1-minute time resolution and 16 angular minute space resolution in both directions. These data comprise three 646 × 516 × 1441 blocks of individual floating-point values for each Pacific source file (the space grid size is different for the Atlantic). The size of each of those data blocks is 1832 Mbyte, while the whole database (tsunami data and accompanying information) for the Pacific region only is 4.2 TB (tera $= 10^{12} = 2^{40}$). This volume of data presents problems with access, storage, and distribution, and hence employing some compression technique to reduce its size is desirable. Donald Denbo reduced the database size to about one half by rearranging data in time series of variable length. In his compression scheme the only data retained are from the moment the data values became greater than some threshold value and these data files are then supplemented by 2D arrays of starting indexes, ending indexes, and starting times (Venturato *et al.*, 2005).

To reduce data volume further, individual time series have been quantized and compressed using Differential Pulse Code Modulation. Currently, data are kept with precision 0.001 cm for water height and 0.0001 cm/sec for velocities. The total size of the entire Pacific database has been reduced to 266 GB, or 6% of its original size, while no visible changes have occurred in either the database time series or in results of MOST calculations that utilize the quantized time series as input. The compression algorithm used is described in the present paper in the following sections:

1. How the data are encoded
2. How the data are stored
3. How much the data can be compressed
4. How the precision of quantization in MOST input affects MOST output

1. How the data are encoded

In the compression algorithm, floating-point type data are converted into integers via uniform quantization. That is, if d is a quantization step, then the quantized value of continuous amplitude u is u/d rounded to the nearest integer. Thus amplitudes, decoded from quantized values, can differ from the original amplitudes by as much as $d/2$ (the round off error).

The utility here is that, while original continuous amplitudes are kept in 4 bytes per value floating-point numbers, quantized values can be stored in a fewer number of bytes, which depends on the range of the values in the data set being compressed. It is typical in waveforms that neighboring samples are correlated. Therefore, the expectation is that the difference between any arbitrary sample amplitude and its neighbor will have smaller values than amplitude values themselves, and the difference between an amplitude and its linear estimate using two neighbor samples will be even smaller. In the case of MOST signals, the quantized input used for compression was the difference between a sample amplitude and its linear estimate made with the two preceding decoded samples. Such a scheme is known as Differential

[1] Joint Institute for the Study of the Atmosphere and Ocean (JISAO), University of Washington, Box 357941, Seattle, WA 98195-4235, USA

Pulse Code Modulation with Quantizer Feeedback (Jayant and Noll, 1984; Jayant, 1974). Fortunately, its implementation looks a lot easier than its name:

Encoder:

```
$u1 = 0 \triangleright (i-1)$th decoded sample
$u2 = 0 \triangleright (i-2)$th decoded sample

    read next ($i-$th) sample $u$
    $\Delta u = u - (2u1 - u2) \triangleright$ quantizer input
    $n = round(\Delta u/d) \triangleright$ quantizer output
    $u0 = d * n + 2u1 - u2 \triangleright$ decode $i-$th sample
    $u2 = u1 \triangleright$ update $u2$
    $u1 = u0 \triangleright$ update $u1$
    goto read next sample
```

Decoder:

```
$u1 = 0 \triangleright (i-1)$th decoded sample
$u2 = 0 \triangleright (i-2)$th decoded sample

    read next ($i-$th) encoded value $n$
    $u = d * n + 2u1 - u2 \triangleright$ decode $i-$th sample
    $u2 = u1 \triangleright$ update $u2$
    $u1 = u \triangleright$ update $u1$
    goto read next encoded value
```

Figure 1 is an example of a waveform given by floating-point values in the range $[-3.04\ 4.20]$ with no two amplitudes the same, encoded and decoded as described above. This signal encoded with quantization steps $d = 0.1$, can take only one out of 11 possible values: integers in the range $[-5\ 5]$, but the reconstructed wave after decoding looks almost the same as the original. The same signal, encoded with quantization steps $d = 0.8$, is represented with only 3 values: -1, 0, and 1, and now the step-like structure in the reconstructed wave is visible.

2. How the data are stored

For this compression scheme, encoded values are recoded in segments beginning with a 3-byte long header containing 2-byte and 1-byte integers. The first header integer is the number of values in that segment, the second one designates a bit rate (32, 16, 8, or 4 bits per sample) used to record that segment. Let us assume that in the data flow there are a consecutive values in the range $[-8\ 7]$, that can be written with 4 bits/sample rate, followed by b larger values in the range $[-128\ 127]$, that can be written with 8 bits/sample rate. Then the entire data stream can be written as two segments of a half-byte values and b 1-byte values, which would take

$$V1 = 6 + a/2 + b$$

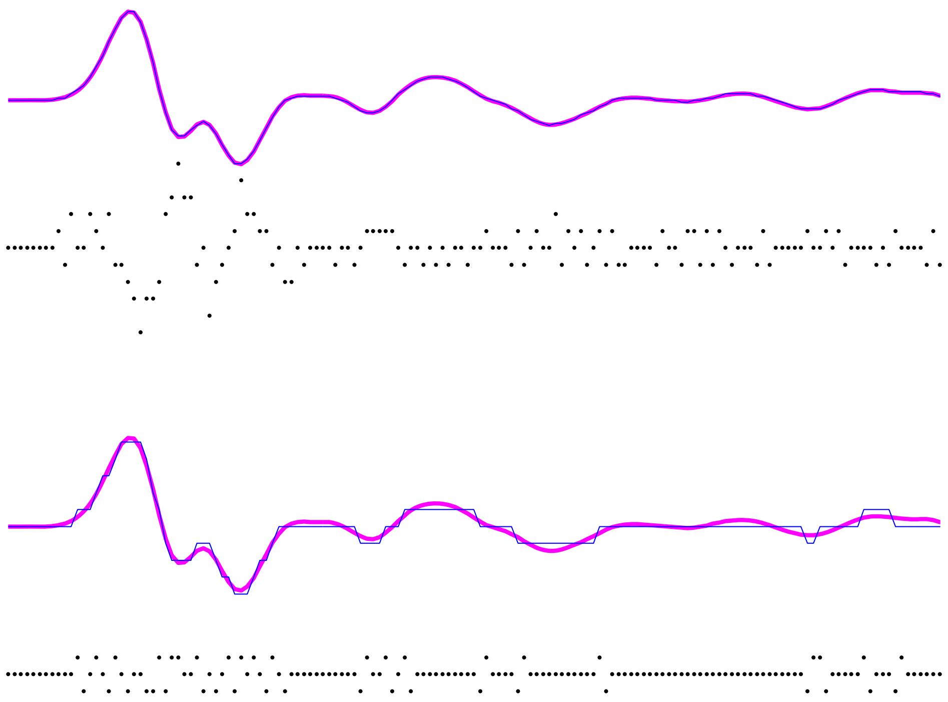


Figure 1: Example wave height data, encoded with quantization step 0.1 (top) and 0.8 (bottom). Magenta: original waveform, blue: decoded waveform, black dots: encoded data.

bytes, including the two headers; or as one segment of $a + b$ 1-byte values, which would take

$$V2 = 3 + a + b$$

bytes. It makes sense to use two segments, if $V2 > V1$ or $a > 6$, which yields

$$MinLen = 6$$

as the minimal length of a segment.

The encoder scans the data twice before writing a compressed version of each time series. During the first pass, it quantizes the data and divides the output flow into segments of a determined bit rate, either 32, 16, 8, or 4, by calculating the bit rate for each quantized value and comparing it with the bit rate for the current segment. If the new bit rate is the same as the current one, it increases the segment value counter by 1; if it is smaller than the current one, it starts a new segment; if it is bigger than the current rate and the current segment already has at least $MinLen$ values, it starts a new segment; if the segment does not yet have this many values, the encoder continues the current segment, but updates its bit rate to the new (larger) one. Also, the encoder only counts, but does not output any zeros at the beginning of a time series, and excludes all the zeros at the end of the last segment. The number of zeros in the beginning of each time series is saved as the start time array, as it has been done in Donald Denbo's format (Venturato *et al.*, 2005). During the second pass, before writing down the next segment the encoder checks to determine if following segments have the same bit rate and connects them, if necessary, into one segment. This technique thus implements an adaptive bit rate as an alternative to an adaptive quantization step (Jayant and Noll, 1984; Atal and Schroeder, 1979).

The decoder restores each time series to the same length (typically 1441 samples) by inserting the recorded number of initial zeros, then by reading the data by segments, and finally adding ending zeros, if necessary, until the required length of the time series is reached.

3. How much the data can be compressed

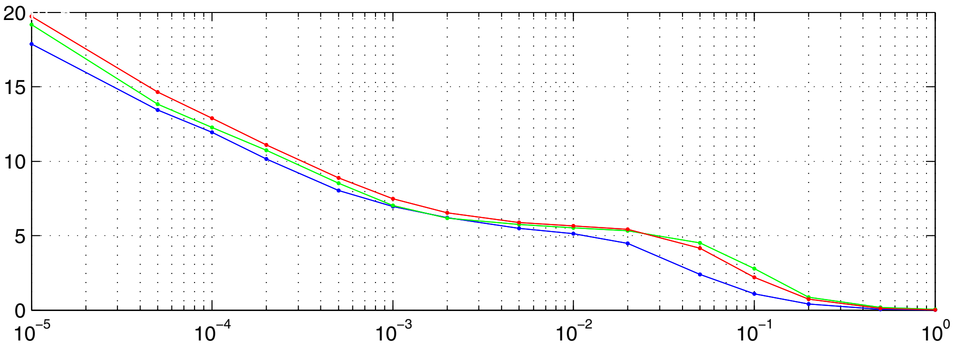
MOST has been run with an input containing continuous wave heights and quantized velocities, and an input containing quantized wave heights and continuous velocities. Quantization in each type of input resulted in the same level of distortions in MOST output, when the quantization step (in cm/sec) for velocities was 1/10 that for water height (in cm). This agrees with the range of input velocity data (in cm/sec) being 1/10 the range of height data (in cm).

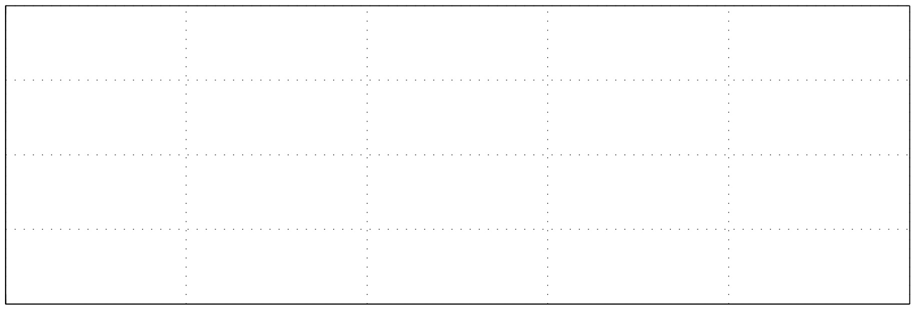
Figure 2 shows the size of encoded data as a percent of the original, which was 640 × 546 × 1441 × 4 bytes, vs. quantization step size, for three randomly selected Pacific sources. The top plot shows a reduction in size in each of three wave height data files; the bottom plot shows similar results for the six velocity data files. The quantization step was varied from 10^{-5} cm to 1.0 cm for wave height data files, and from 10^{-6} cm/sec to 0.1 cm/sec for velocity data files.

All the plots for the same variable in Fig. 2 closely follow each other. They all have a plateau where increasing a quantization step results in very little reduction of data size. To the left of the plateau, the reduction is due both to excluding the zero values at the beginning and the end of each time series, and also to recording quantized values with a lower bit rate. The plateau starts when almost all the values are encoded with the smallest (4 bits/sample) bit rate. After that, increasing a quantization step results in dropping the signal level to the level of quantization noise. To the right of the plateau, data size reduction is entirely due to cutting away larger and larger chunks of data at the beginning and at the end of each time series, due to the larger values that round to zero. All these are reasons to expect the highest compression in the data for the least loss in precision, when the quantization steps are selected at the beginning or slightly to the left of the plateau.

Quantization steps presently used for database compression are 0.001 cm for wave height and 0.0001 cm/sec for velocities.

Quantization with 0.001 cm and 0.0001 cm/sec will be referred to as fine quantization, while quantization with 100 times bigger steps, that is, 0.1 cm and 0.01 cm/sec, respectively, will be referred to as coarse quantization. With fine quantization, the size of the data files for each source gets reduced to about 5–7% of its original size as a *space* × *time* block of floating-point


20
15
10
5
0
10^{-5}
10^{-4}
10^{-3}
10^{-2}
10^{-1}
10^{0}



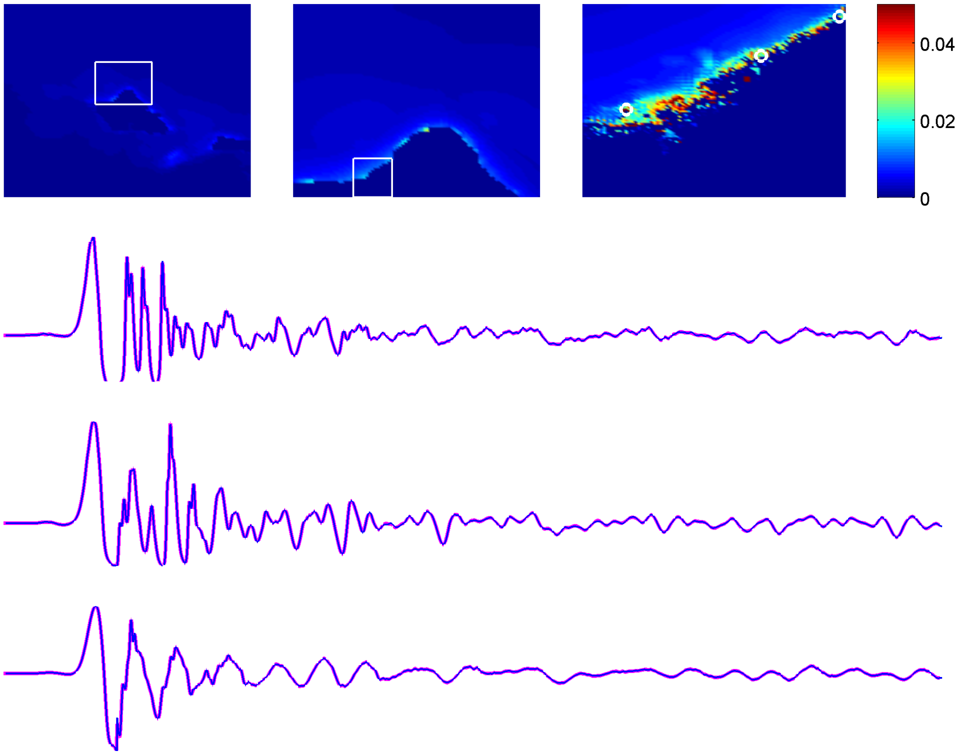


Figure 3: Maximum difference in wave heights calculated with original and finely quantized data, in every point in the A, B, and C grids (top plots, left to right), as a fraction of maximum wave height in that grid. Combined source AASZ 30(a17 + b17 + a18 + b18 + a19 + b19) corresponds to 9.0 Mw. Bottom plots: time series in grid C (in locations marked with "o"); blue: quantized input, magenta: continuous input.

not visible on any of the time series plots, even though all three observation points have been selected to represent a "worst case," where this difference is relatively big, according to surface plot.

By combining the six input data sets with a factor of 30 increases the quantization noise $30\sqrt{6} \approx 73$ times, and still introduces rather negligible distortions in MOST output. To illustrate how coarse we can make the MOST input and still get practically the same output, MOST was run with an input quantized at 100 times as coarse, that is, with 0.1 cm step for water height and 0.01 cm/sec step for velocities. Data file sizes are now 3.1%, 0.6%, 0.7% for heights and both currents, respectively, for AASZ a19 source; and 1.5%, 0.4%, 0.4% for heights and both currents, for KKSZ b3 source.

Figures 4 and 5 show the maximum difference in wave heights calculated with continuous and coarsely quantized input, as a fraction of the maximum height in grids A, B, and C, respectively, and selected near-shore time series. The difference in the plan view plots is mostly due to high frequency waves that had been generated by step-like structure in roughly truncated input. A major part of these waves dies out while they propagate ashore. In the points where, due to bathymetry features, short waves attenuate least, larger

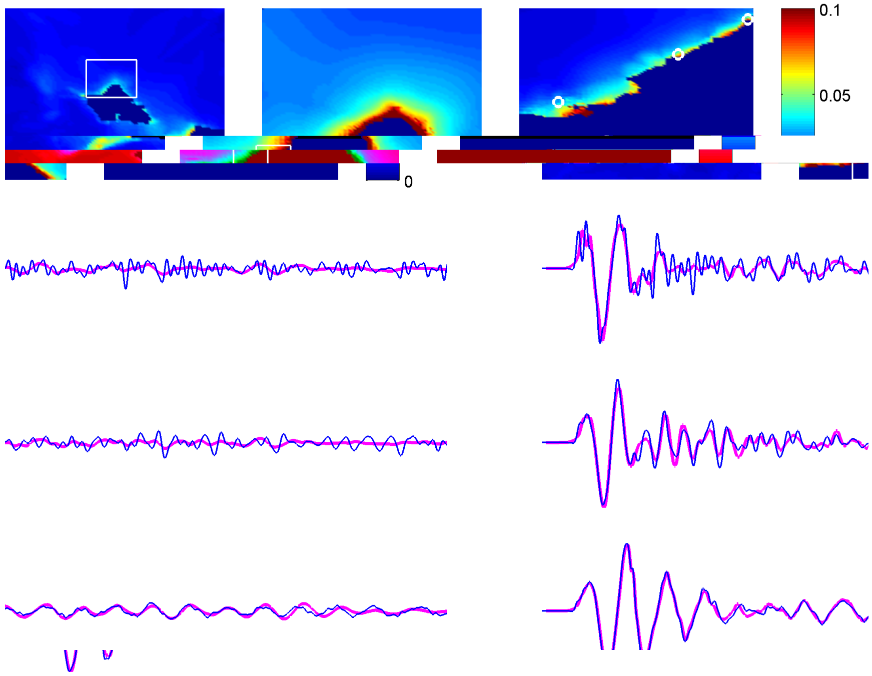


Figure 4: Maximum difference in wave heights calculated with original and coarsely quantized data, in every point in A, B, and C grids (top plots, left to right), as a fraction of maximum wave height in that grid. Source AASZ a19. Bottom plots: time series in grid C (in locations marked with "o"); blue: quantized input, magenta: continuous input.

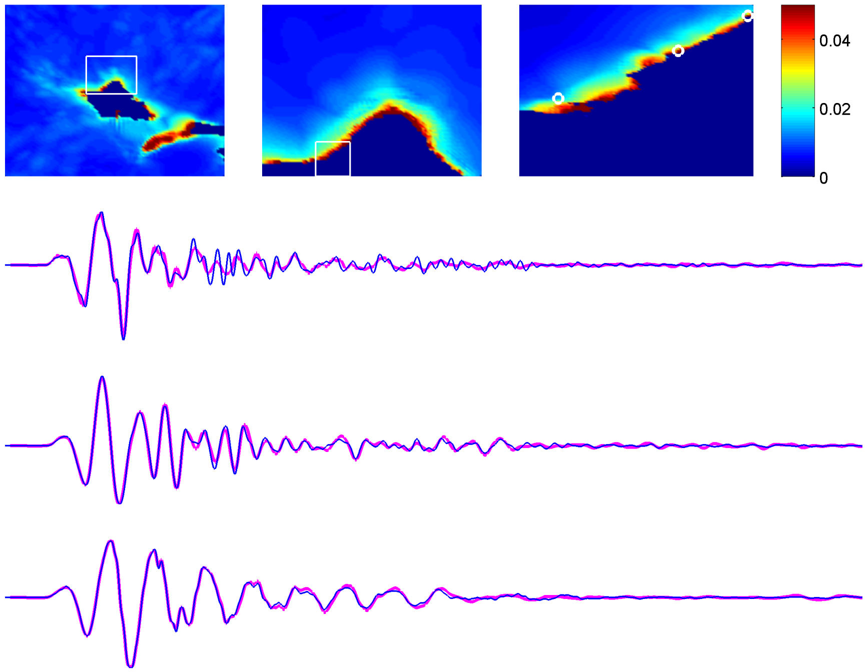


Figure 5: Maximum difference in wave heights calculated with original and coarsely quantized data, in every point in A, B, and C grids (top plots, left to right), as a fraction of maximum wave height in that grid. Source KKSZ b3. Bottom plots: time series in grid C (in locations marked with "o"); blue: quantized input, magenta: continuous input.

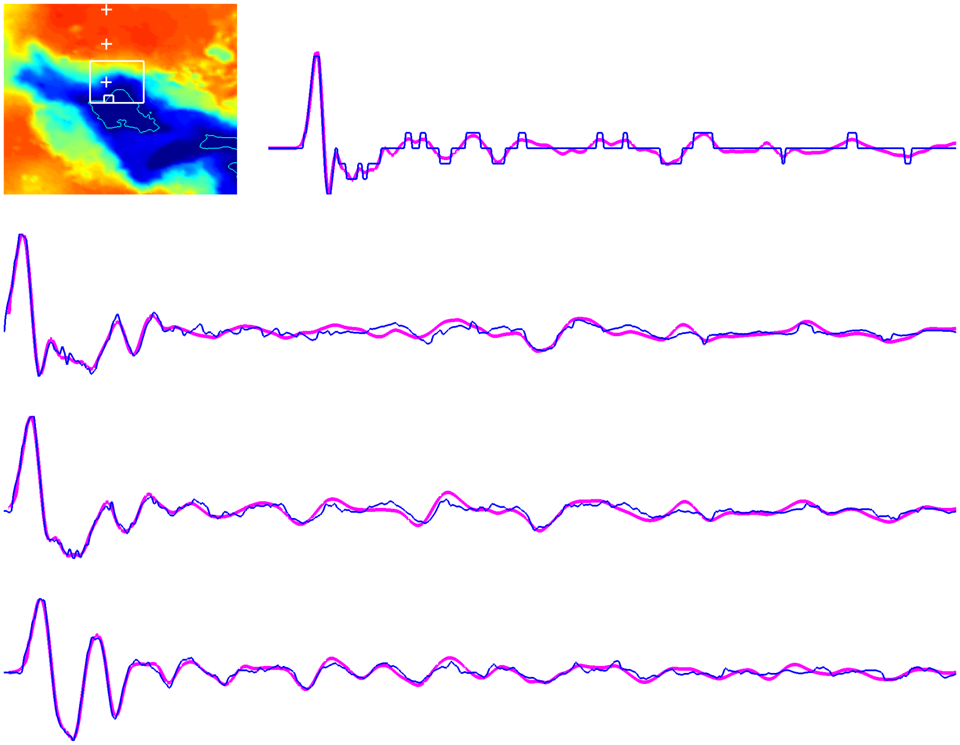


Figure 6: Evolution of coarsely quantized wave as it propagates toward the shore. A sample time series entering grid A (top), and time series calculated as MOST output in points marked with "+" on grid A bathymetry plot; blue: quantized input, magenta: continuous input.

differences are observed in waveforms calculated with original and quantized input. However, for 99% of all points in grid C, this difference does not exceed 6.4% of max wave height in C for AA a19 source and 4.5% for the KK b3 source. In the absence of those short waves that are generated, the signals calculated with continuous and even roughly quantized data appear almost identical.

Very high precision in MOST input does not seem at all to be a necessary condition of achieving high precision in MOST output. On the contrary, these tests show that both the exact wave, entering grid A, and its rough estimate, result in almost the same wave coming ashore.

Figure 6 shows the evolution of a coarsely quantized wave (0.1 cm and 0.01 cm/sec) from the AASZ a19 source as it propagates through grids A and B. We see that the computational scheme acts as a low pass filter both in space and time, leading to smoothing the quantization steps and literally restoring a close approximation to the original wave while it propagates through inner grids, wherever the signal level was sufficient to produce non-zero quantizer output.

5. Summary

Quantizing the surface elevation with a quantization step of 0.001 cm and velocities with a step of 0.0001 cm/sec does not introduce noticeable changes either in MOST output (calculated wave heights in grids A, B, and C) or input, that is, time series in the propagation database. The quantization used with second-order Differential Pulse Code Modulation with Quantizer Feedback and adaptive bit/sample rate, as described here, allows for reducing the Propagation Database size to 6% of its original size as a *space* × *time* block of floating-point data. Also, a quantization even 100 times coarser changes MOST output only within a few per-cent, which points to a possibility for more compression.

6. Acknowledgments

This publication was funded by NOAA and the Joint Institute for the Study of the Atmosphere and Ocean (JISAO) under NOAA Cooperative Agreement No. NA17RJ1232, Contribution #1391.

7. References

Atal, B.S., and M.R. Schroeder (1979): Predictive coding of speech signals and subjective error criteria. IEEE Transactions on Acoustics, Speech and Signal Processing, 247–254.

Venturato, A.J., D.W. Denbo, V.V. Titov, K.T. McHugh, and P. Sorvik (2005): Tsunami Forecasting System Design and Development. *Eos Trans. AGU, 86*(52), Fall Meet. Suppl., Abstract IN23B-1211.

Jayant, N.S., and P. Noll (1984): *Digital Coding of Waveforms.* New Jersey, Prentice-Hall, 688 pp.

Jayant, N.S. (1974): Digital coding of speech waveforms: PCM, DPCM, and DM quantizers. *Proceedings IEEE, 62*(5), 611–632.