An intelligent data model for the storage of structured grids

John Clyne (clyne@ucar.edu), AlanNorton (alan@ucar.edu)

Problem

A consequence of the extraordinary computing capability afforded to numerical modelers is the production of vast amounts of numerical data. Our ability to manage, analyze, and gain insight from these numerical outputs, however, has not kept pace with our ability to generate them. One of the largest contributing factors is the disparate rate of advancement of different computing technologies. Microprocessor performance has doubled roughly every 18 months in accordance with "Moore's law". I/O bandwidths and secondary storage capacities have historically been on more modest improvement curves, a trend that is unlikely to abate for the foreseeable future as evidenced by the U.S. DOE's plans for an *exascale* computer[1]. The impact of this disparity in capability is twofold:

- insufficient long-term storage capacity necessary for analyses that may be conducted over months or even years; and
- inadequate I/O bandwidth to process data at interactive rates.

Intelligent data storage

Current storage practices for gridded simulation data are simplistic: we save memory location contents for every sample, at every grid point location, at some prescribed frequency. This straightforward approach fails to take advantage of the coherency in neighboring grid points that invariably exists in most numerical simulations. Such coherency is the foundation of lossy and lossless compression strategies that make DVD-Video, digital cameras, streaming movies and audio all possible today.

While losing information that has been computed at great cost may at first appear anathema to computational scientists, it must be noted that some forms of lossy compression are already commonly used today. For example, modelers may regulate the frequency of their outputs to fit the available storage. And, while computations are generally performed at double precision outputs used for analysis are typically truncated to single.

These ad hoc approaches are driven by their simplicity, not by any methodology that would ensure the highest possible information retention for a given reduction in data volume. Here we report on a lossy data compression strategy that is based on the information compaction capabilities of *Wavelets*[2,3].

This work is supported by NCAR's Computational & Information Systems Lab. (CISL), and NSF Grants # 03-25934 and 09-06379) and through a TeraGrid GIG award.

Discrete wavelet transform

Similar to Fourier transforms a Wavelet transform expresses a signal *f(t)* as linear expansion:

$$f(t) = \sum_l a_l \psi_l(t)$$

where a_l are real-valued coefficients, and ψ_l are basis functions. The basis function ψ_l is constructed from a *scaling* function, ϕ:

$$\psi(t) = \sum_k h_\psi(k)\sqrt{2}\phi(2t-k), \quad k \in Z \quad \text{wavelet function}$$

which is recursively constructed from scaled, dyadic translates of itself:

$$\phi(t) = \sum_k h_\phi(k)\sqrt{2}\phi(2t-k), \quad k \in Z \quad \text{scaling function}$$

which leads to a more general two-parameter representation of a wavelet expansion of *f*:

$$f(t) = \underbrace{\sum_k c(k)\phi_k(t)}_{\textit{Approximation}} + \overbrace{\sum_k \sum_{j=0}^{\log_2 N} d_j(k)\,\psi_{j,k}(t)}^{\textit{Detail}}$$

Key wavelet properties:

- Compact support: wavelets are zero outside of narrow interval
- Transforms can localize signal details (frequencies) in time (space). This tells us not just *what* frequencies are present but *when* they occur
- Computationally efficient: *O(n)*, compared with *O(N*LogN)* for Fourier

Compression with wavelets

We want to prioritize coefficients based on contribution to signal

$$f(t) = \sum_{n=0}^{N-1} a_n u(t), \quad \text{original } f(t) \qquad \hat{f}(t) = \sum_{m=0}^{M-1} a_{\pi(m)} u(t), \quad (M < N), \quad \text{compressed } f(t)$$

L^2 error given by:

$$L^2 = \left\| f(t) - \hat{f}(t) \right\|_2^2$$

If *u(t)* are *orthonormal*, then:

$$L^2 = \sum_{i=M}^{N-1} \left(a_{\pi(i)}\right)^2 = \left\| f(t) - \hat{f}(t) \right\|_2^2, \text{ where } a_{\pi(i)} \text{ are discarded coefficients}$$

- The L^2 error is the sum of the squares of the coefficients left out!
- To minimize the L^2 error, simply **discard** (or **delay** transfer of) the smallest coefficients!
- If discarded coefficients are zero, there is no information loss!

Results: particle-turbulence interactions

Here we show results of visualizing data arising from high resolution numerical simulations after compression by discarding the smallest wavelet coefficients. Figures 1 and 2 depict results from a 2048^3 simulation that explores particle-turbulence interactions in conditions which mimic cumulus cloud cores[4]. Compression rates are indicated in the top-left corner of each sub-figure. Uncompressed variables occupy 32GBs/time step.

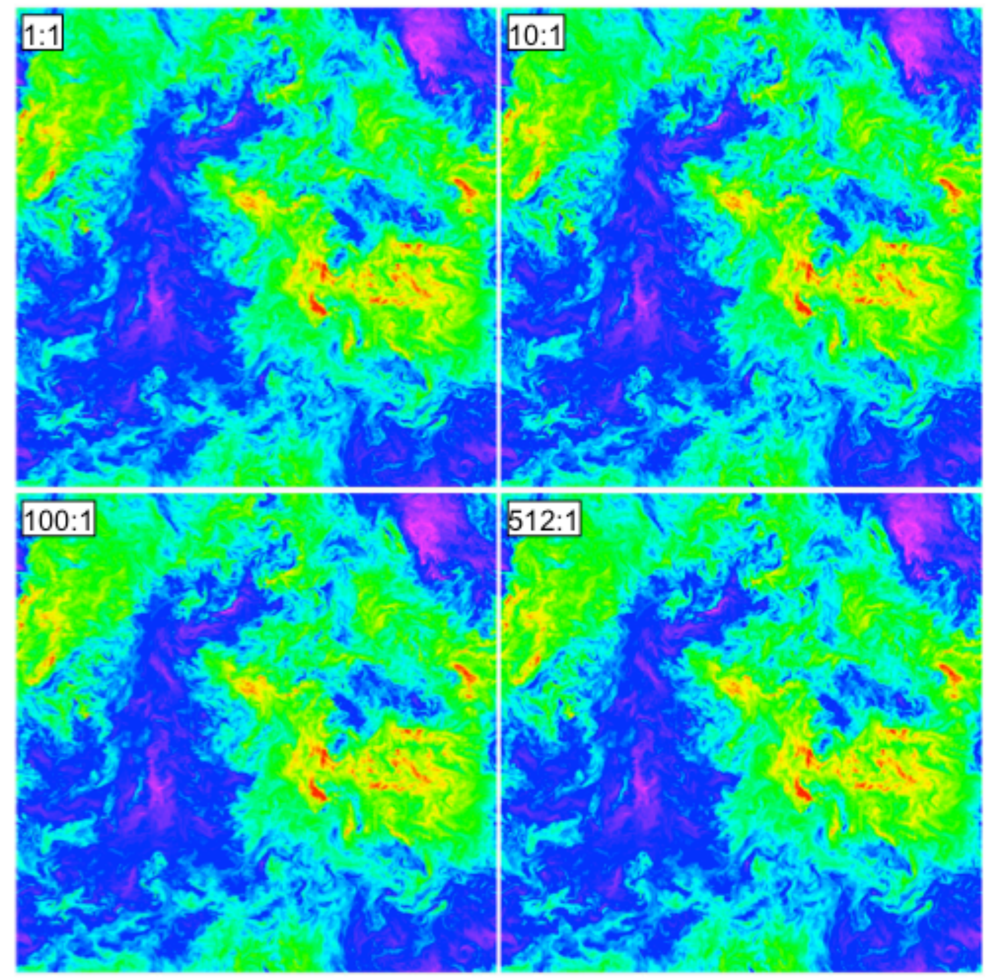

Figure 1: A 2D slice of the *W* component of velocity field

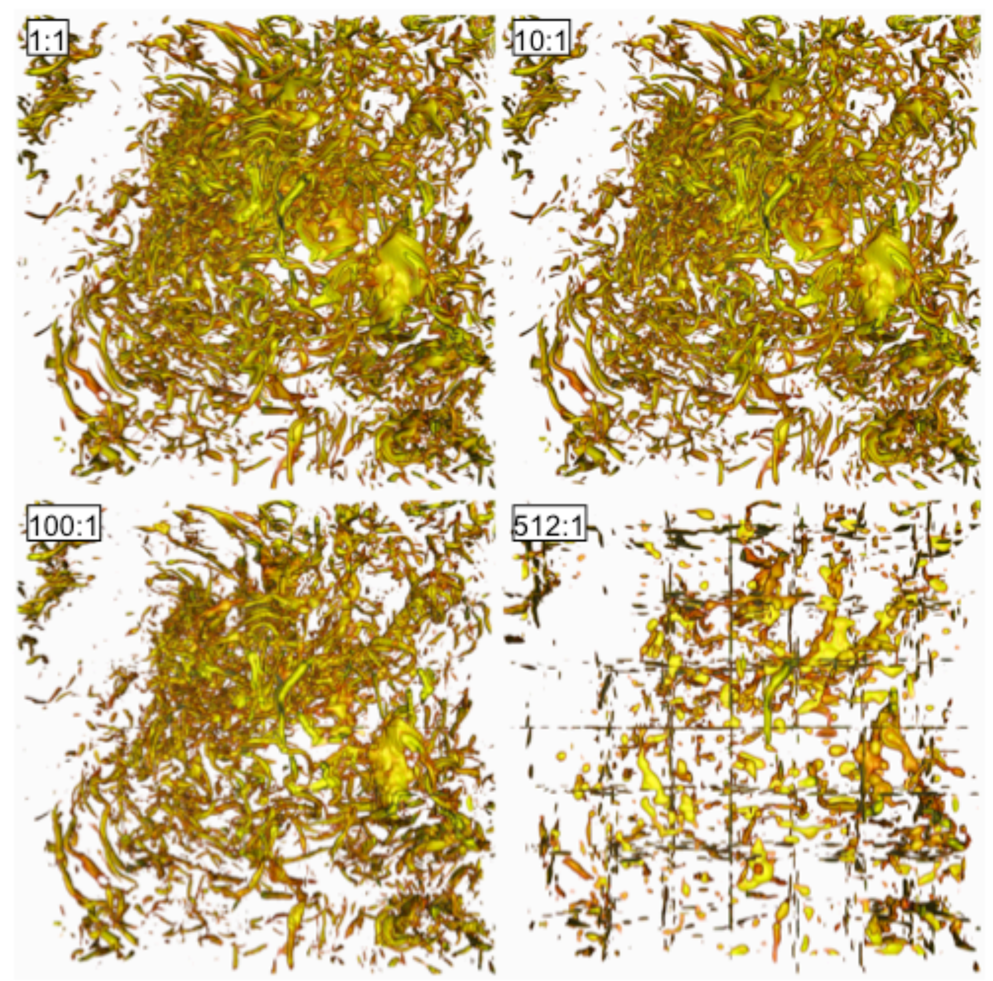

Figure 2: Volume rendering of enstrophy field derived by taking the curl of compressed velocity field components using 6th-order finite differences. Uncompressed data, top left.

Results: Hurricane Sandy

Here we show results for a very high resolution simulation of Hurricane Sandy, computed on a 0.5km grid (~5000x5000x150 grid points)[5].

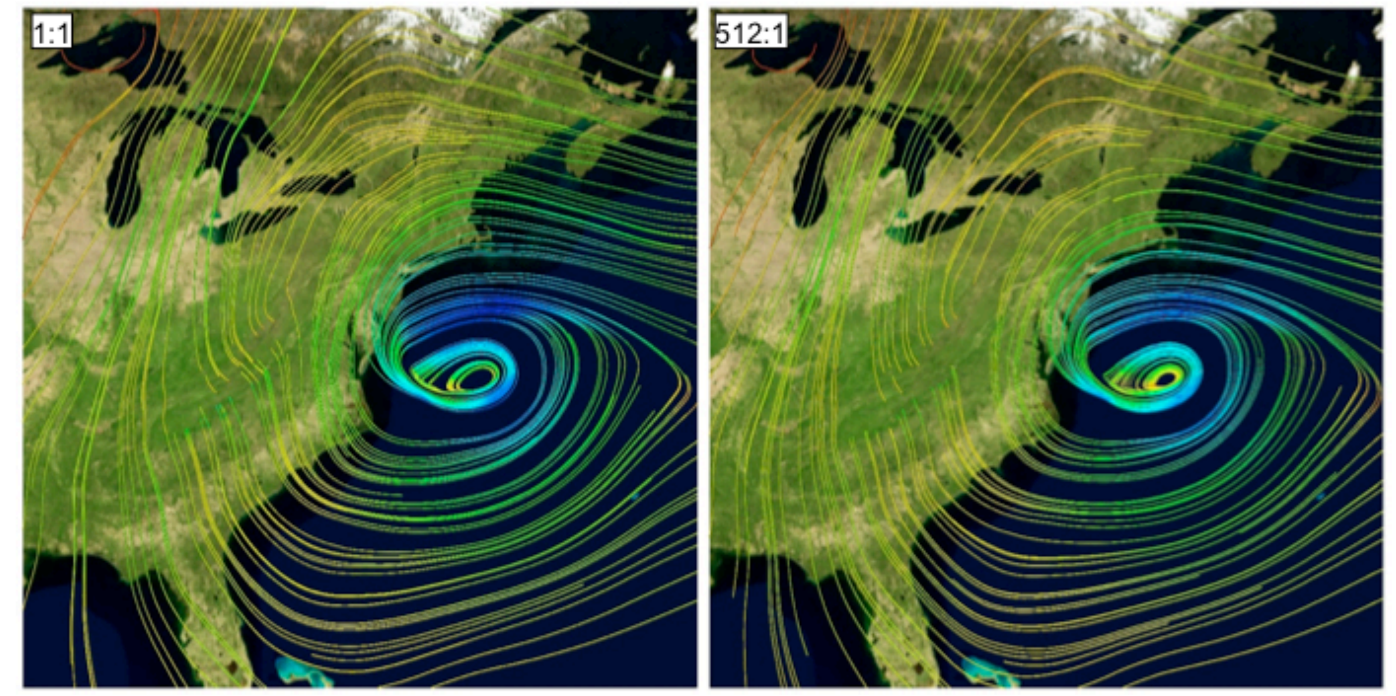

Figure 3: Streamlines of velocity computed using 4th order Runge-Kutta approximation for original(left) and compressed 512:1 (right) data.

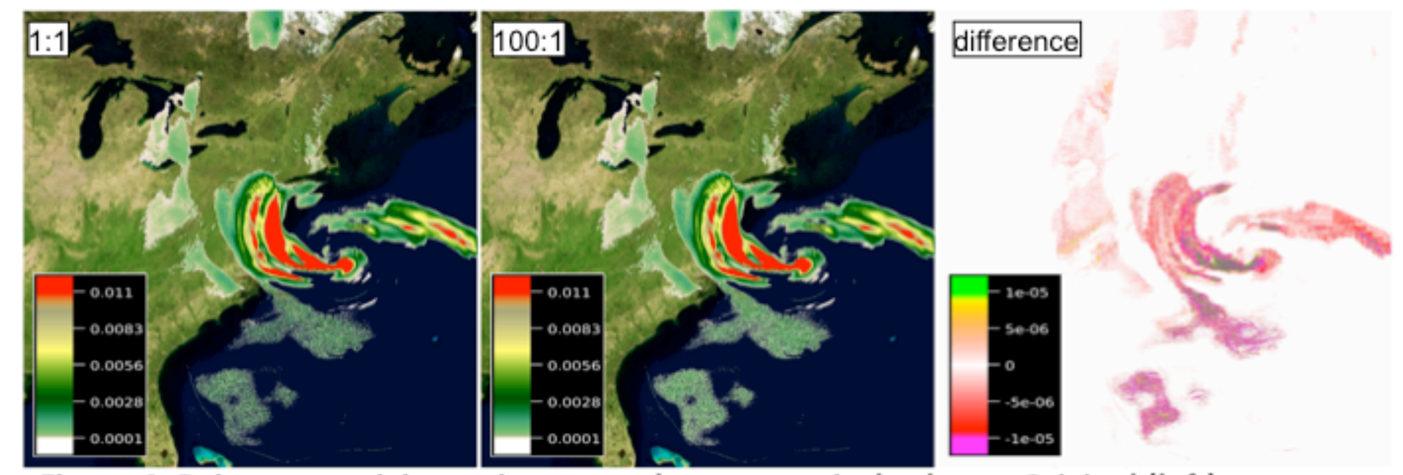

Figure 4: Rain water mixing ratio summed over a vertical column. Original (left), compressed 100:1 (middle), and difference between fields (right).

	1:1	2:1	4:1	8:1	16:1	32:1	64:1	128:1	256:1	512:1
LMax	0.0	0.02	0.09	0.2	0.3	0.6	1.0	1.9	1.8	3.6
NRMSE	0.0	6e-5	0.0002	0.0004	0.0009	0.002	0.003	0.004	0.006	0.01
Read (sec)	850	211	107	58	32	19	11	7	4	3
Xform (sec)	11	12	11	11	12	15	16	16	16	16

Table 1: Error (lmax, NRMSE), read and transform times for the U component of velocity for the 2048^3 turbulence data.

	1:1	2:1	4:1	8:1	16:1	32:1	64:1	128:1	256:1	512:1
LMax	0.0	5e-6	5e-5	0.0002	0.0003	0.0008	0.001	0.003	0.004	0.005
NRMSE	0.0	2e-6	2e-5	6e-5	0.0002	0.0003	0.0005	0.0008	0.001	0.002
Read (sec)	571	131	177	56	41	27	12	16	5	3
Xform (sec)	7	7	7	7	8	8	9	10	10	10

Table 2: Error (lmax, NRMSE), read and transform times for the rain mixing ratio field of the 5147x5199x149 Hurricane Sandy simulation data set.

References

1. Stevens, R. and White, A. *Architectures and technology for extreme scale computing*, 2009.
2. J. Clyne and A. Norton. Progressive Data Access for Regular Grids, in High Performance Visualization, 2012
3. S. Mallat. A Wavelet Tour of Signal Processing, Third Edition: The Sparse Way, 2008.
4. Data source: Peter J. Ireland and Lance R. Collins (Cornell University)
5. Data source: Peter Johnsen (Cray), Mel Shapiro (NCAR), and Tom Galerno (NCAR)

NCAR
NATIONAL CENTER FOR ATMOSPHERIC RESEARCH