LA-UR-12-1352

Approved for public release;
distribution is unlimited.

Title: Subband coding for large-scale scientific simulation data using JPEG 2000

Author(s): Christopher M. Brislawn, Jonathan L. Woodring, Susan M. Mniszewski, David E. DeMarle, and James P. Ahrens

Intended for: Southwest Symposium on Image Analysis & Interpretation
IEEE Computer Society
Santa Fe, NM, April 22-24, 2012
pages 201-204

Los Alamos National Laboratory, an affirmative action/equal opportunity employer, is operated by the Los Alamos National Security, LLC for the National Nuclear Security Administration of the U.S. Department of Energy under contract DE-AC52-06NA25396. By acceptance of this article, the publisher recognizes that the U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or to allow others to do so, for U.S. Government purposes. Los Alamos National Laboratory requests that the publisher identify this article as work performed under the auspices of the U.S. Department of Energy. Los Alamos National Laboratory strongly supports academic freedom and a researcher's right to publish; as an institution, however, the Laboratory does not endorse the viewpoint of a publication or guarantee its technical correctness.

Form 836 (7/06)

Subband coding for large-scale scientific simulation data using JPEG 2000

(Invited Paper)

Christopher M. Brislawn*, Jonathan L. Woodring*, Susan M. Mniszewski*,
David E. DeMarle†, and James P. Ahrens*
*Los Alamos National Laboratory, Los Alamos, NM 87545–1663 USA
Email: {brislawn,woodring,smm,ahrens}@lanl.gov
†Kitware, Inc., 21 Corporate Drive, Clifton Park, NY 12065–8662 USA
Email: dave.demarle@kitware.com

***Abstract*—The ISO/IEC JPEG 2000 image coding standard is a family of source coding algorithms targeting high-resolution image communications. JPEG 2000 features highly scalable embedded coding features that allow one to interactively zoom out to reduced resolution thumbnails of enormous data sets or to zoom in on highly localized regions of interest with very economical communications and rendering requirements. While intended for fixed-precision input data, the implementation of the irreversible version of the standard is often done internally in floating point arithmetic. Moreover, the standard is designed to support high-bit-depth data. Part 2 of the standard also provides support for three-dimensional data sets such as multicomponent or volumetric imagery. These features make JPEG 2000 an appealing candidate for highly scalable communications coding and visualization of two- and three-dimensional data produced by scientific simulation software. We present results of initial experiments applying JPEG 2000 to scientific simulation data produced by the Parallel Ocean Program (POP) global ocean circulation model, highlighting both the promise and the many challenges this approach holds for scientific visualization applications.**

***Index Terms*—scientific visualization; JPEG 2000; data compression; image coding; subband coding; floating point data; high-performance computing**

I. Introduction

While scientific computing continues to tackle bigger and bigger modeling and simulation problems (e.g., Figure 1), improvements in serial hardware performance have slowed as clock rates have stagnated at a few GHz. To maintain a steady increase in supercomputing performance, the recent trend has been an exponential growth in processor parallelism. This is changing the fundamental bottlenecks in high-performance computing (HPC). When supercomputers reach exascale (10^{18} operations/second) some time in the next decade, the limiting factor in system performance is expected to be power consumption for data movement and memory, not floating point computations. It will be impossible to move all of the data computed by an exascale simulation out of core and into nonvolatile storage fast enough to maintain a reasonable pace for the simulation. For a good overview of the prospects for exascale computing, see the reports at http://science.energy.gov/ascr/news-and-resources/program-documents

In such a bandwidth-limited HPC environment, it is imperative to make the best possible use of the available communications bandwidth, memory, and nonvolatile storage. This means that source coding and compression tailored for simulation data will likely play an important role in successful exascale HPC data management and analysis. Source coding represents a new direction for scientific computing, however, since communications issues in large-scale scientific computing have traditionally been addressed with hardware solutions, an approach that is not expected to scale up to exascale.

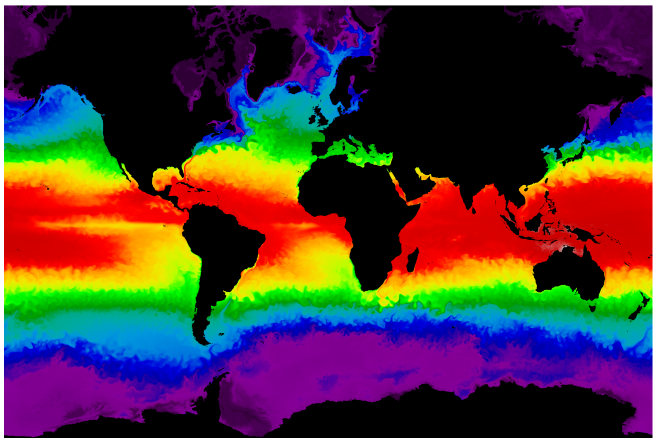


Fig. 1. Surface temperature field from a POP ocean simulation.

Adding a new core capability to the HPC technology mix raises many new questions and challenges. *How* should data be encoded for optimal HPC bandwidth utilization? *Where* in the simulation flow should encoding be performed? *Which part* of the data is most amenable to significant bandwidth reduction? *What effects* will bandwidth reduction have on data analysis and visualization? *How much entropy* needs to be retained in the various components of an HPC data stream to preserve the scientific answers the simulation was designed to deliver?

This paper presents recent efforts by the authors to start addressing some of these questions, primarily in the context of scientific visualization, using the JPEG 2000 image coding standards. JPEG 2000 is certainly not the only source coding approach applicable to HPC simulation data. Some tools, notably VAPOR [1], simply store uncompressed floating point data in a multiresolution framework produced by a wavelet transform decomposition of the data. Older approaches [2]–[9] compressed wavelet transform coefficients using combinations of coefficient thresholding or quantization, run-length coding, and/or Huffman coding. These older approaches are gradually being superseded, however, by modern embedded coding techniques that offer greater scalability, better bandwidth efficiency, fine-grained rate control, and decoder-driven progres-

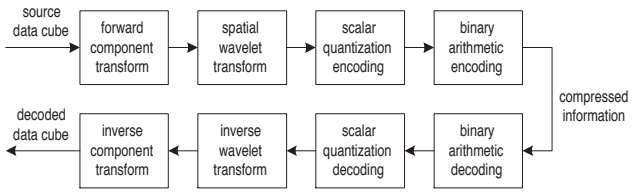


Fig. 2. Overview of JPEG 2000 irreversible encoding and decoding.

sive transmission capabilities. The ability to take advantage of these state-of-the-art source coding features using a standards-based toolkit makes JPEG 2000 an attractive candidate for HPC data management.

II. JPEG 2000 Coding

JPEG 2000 [10]–[13] is a family of international standards for digital image coding developed by the Joint Photographic Experts Group (JPEG) of the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC) and published jointly with the International Telecommunications Union (ITU) [14], [15]. As shown in Figure 2, the "irreversible" version of JPEG 2000 reads in logically rectangular source data and applies decorrelating transforms, usually in floating point arithmetic. In the HPC context, the "components" in the input could correspond either to 2-D slices in volumetric data or to multiple 2-D physical fields. The decorrelated subbands produced by cross-component and spatial wavelet transformation are then quantized to fixed-point values and entropy-encoded using block-based binary arithmetic bit-plane coding. (An alternative "reversible" path transforms fixed-point input using nonlinear integer-to-integer wavelet transforms, bypassing the quantization step and enabling lossless subband coding.)

This encoding process makes the JPEG 2000 representation highly local and highly scalable with respect to a number of important parameters. Wavelet transform coefficients produced by short FIR filters are spatially localized samplings of the input data, and the hierarchical nature of wavelet transform decompositions makes JPEG 2000 intrinsically scalable with respect to spatial resolution. The block-based structure of the arithmetic encoding ensures that the compressed output preserves the spatial localization inherent in the wavelet transform coefficients. The state-based binary arithmetic bit-plane encoding also makes the JPEG 2000 representation highly scalable with respect to sample precision: approximate values for the coefficients can be reconstructed in each coding block from an entropy-encoded bitstream truncated after any arbitrary number of coding passes.

For the HPC data experiments conducted so far, the scalar quantization step has been designed using subband quantization characteristics based on a nominal input entropy of around 25–27 bits per input sample (assuming 32-bit floating point data). Entropy encoding then encodes *all* quantized bit-planes, with "quality layers" set in the compressed codestream. Quality layering optimizes the reconstructed data decoded at a set of prespecified bit rates, typically spaced logarithmically; e.g., 8, 4, 2, 1, 0.5, 0.25 bits/sample. This approach results in

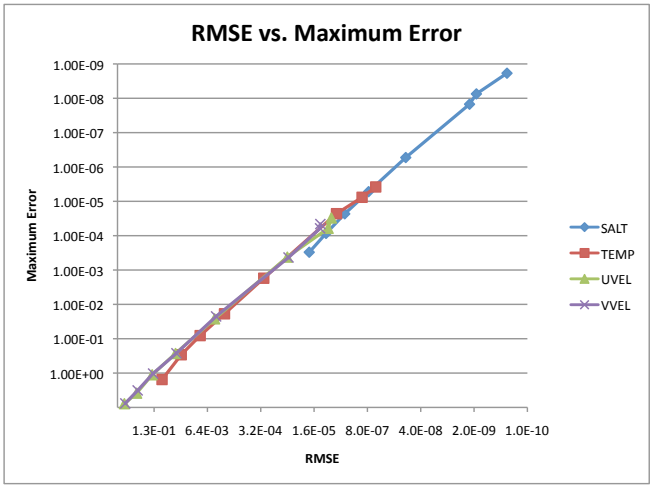


Fig. 3. Plots of max error vs. root-mean-square error for temperature, salinity, and velocity components at various bit rates.

relatively little (around 2:1) reduction in the total size in bytes of the input data but restructures the data in a way that enables *huge* reductions in the bandwidth needed to reconstruct a region of interest at a resolution and precision appropriate for a particular task, such as visualization. The ISO standard enables users to perform demand-driven, packet-level retrieval of compressed data from a JPEG 2000 archive using the JPEG 2000 Part 9 interactive client-server protocol (JPIP) [16]. These features—localization with respect to region of interest, spatial resolution, sample precision and data field, plus standardized client-server message queueing—totally decouple the cost of accessing small portions of a huge dataset from the size of the dataset. The client pays transmission and decoding/rendering costs only for the fraction of data that is explicitly requested.

III. Analysis and Visualization of Compressed HPC Data

The simulation shown in Figure 1 comes from a global ocean circulation model with nominal $\frac{1}{10}^{\circ}$ resolution generated by the Parallel Ocean Program (POP) [17]. The $3600 \times 2400 \times 42$ arrays of floating point data (temperature, salinity, E-W velocity, and N-S velocity) are truncated to single precision for scientific visualization and analysis, but this still amounts to 5.4 GB per time step. POP data is compressed using the Kakadu JPEG 2000 software implementation per the above outline, saving all arithmetic coding passes in a compressed codestream with multiple quality layers. Kakadu [18], designed and written by one of JPEG 2000's principal architects [19], is the most complete and most rigorously tested software implementation of JPEG 2000 available at present. We note, though, that there is at least one open-source JPEG 2000 software project (OpenJPEG [20]) in active development. A JPEG 2000 reader has been implemented in ParaView [21] using the Kakadu library to enable visualization and various quantitative error analysis tasks for JPEG 2000-compressed data within ParaView/VTK.

While Kakadu maximizes the signal-to-noise ratio (SNR) in images reconstructed at each quality layer bit rate, it is

also desirable in HPC data management scenarios to quantify the maximum pointwise (or L^∞) error in a reconstructed data set to provide scientific end-users with a worst-case pointwise error bound. Optimizing source coding schemes to minimize an L^∞ rate-distortion metric is probably intractably hard, so it is of interest to see the highly linear relationship between RMSE and L^∞ error reported in Figure 3. The L^∞ error in Figure 3 is proportional to RMSE for each physical field across reconstructed bit rates ranging from 8 down to 0.25 bits/sample. Particularly striking is the observation that the constant of proportionality is *about the same* for all four fields, L^∞ error $\approx 10 *$ RMSE, despite markedly different distributional properties for the different physical variables. This finding is very preliminary, but if it holds empirically in more general contexts then it may be possible to model empirical L^∞ error as a function of bit rate in terms of RMSE.

In a similar preliminary vein, Figure 4 presents empirical L^∞ rate-distortion behavior for numerical partial derivatives of the physical fields. Directional derivatives in the N-S direction behave similarly. Partial derivatives of velocity components are key features in the analysis of ocean eddies using the Okubo-Weiss approach [22]. We expect SNR to have a certain range of linear rate-distortion behavior for well-designed source coders, but it is striking to see such linear L^∞ rate-distortion behavior for numerical derivatives. Additional rate-distortion analysis on JPEG 2000-encoded POP data is presented in [23].

To see the perceptual quality of scientific visualization on compressed and reconstructed data, Figure 5 shows a region of size 475×358 in a salinity field progressively decoded and rendered using the ParaView JPEG 2000 reader at entropies of 0.25, 0.5, 1.0, and 8.0 bits/sample [23]. Isocontours and amplitude colormaps were generated separately for each reconstructed array. The global maximum relative error (relative to the original 32-bit input) in the data reconstructed at 8.0 bits/sample is less than 10^{-7}, or better than 140 dB SNR, which is roughly machine precision. The 8-bit reconstruction thus provides analyses and visualizations that are

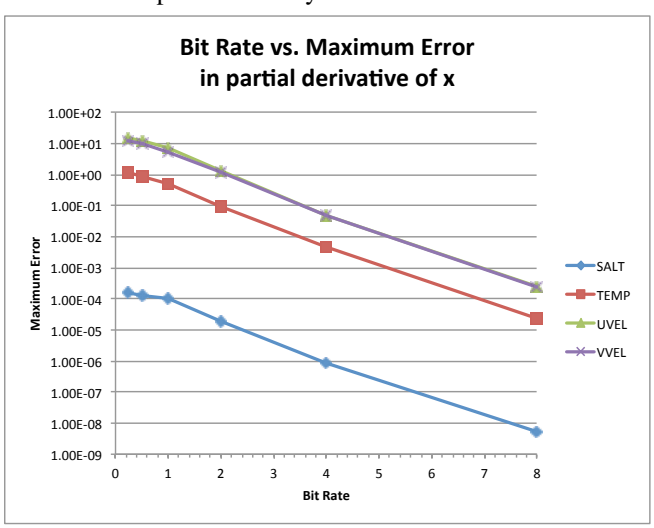


Fig. 4. Max error vs. rate for E-W directional derivatives of salinity, temperature, and velocity components at various bit rates.

indistinguishable from those obtained using the original 32-bit floating point data, in spite of being compressed 4:1. Small discrepancies, smoothing artifacts, and isocontour degradation are visible at 0.5 and 0.25 bits/sample, but the degradation is graceful as rates drop below 1 bit/sample. It is thus entirely plausible that many basic scientific visualization tasks like these can be performed reliably at entropies as low as 1 bit/sample, a 32:1 reduction in bandwidth (1.5 orders of magnitude) relative to single-precision floating point. The potential savings from region-of-interest and reduced-resolution retrieval from exascale data sets are even greater.

IV. Conclusions and Future Challenges

The surface has barely been scratched when it comes to incorporating modern transform-based communications source coding standards in HPC data management. Part of the problem is that HPC end-users traditionally have not had to think much about bandwidth-efficient data management, and HPC suppliers have until now relied on big iron to overcome HPC bottlenecks. Consequently, there are no established best-practices in this area, so many HPC research groups have been experimenting with in-house floating point data compression schemes. The LANL-Kitware group is pursuing a path of standards-based source coding technology both to avoid reinventing as many wheels as possible and in recognition of the fact that HPC data encoding is really a *communications* issue, and communications require the interested parties to agree on standardized protocols. In particular, it seems clear that *someone* ought to explore just how far the JPEG 2000 standard can be pushed in the HPC arena.

There are plenty of open questions raised by this particular use of JPEG 2000. Following are a few of the problems the LANL-Kitware group is working on (or worrying about).

1) Efficiently estimate the entropy of floating point input to enable appropriate quantization of wavelet transform data in irreversible JPEG 2000 encoding processes.
2) Enable three-dimensional, multicomponent data encoding using JPEG 2000 Part 10 [24].
3) Implement interactive multiscale client-server semantics between ParaView clients and JPIP servers.
4) Develop low-entropy schemes for interpolating uninitialized grid regions in HPC data (e.g., continents and islands in POP ocean simulations).
5) In-situ JPEG 2000 encoding for exascale simulations.

Acknowledgments

This work was supported by the Advanced Scientific Computing Research Program (ASCR) operated by the U. S. Department of Energy's Office of Science.

References

[1] M. Rast and J. Clyne, "Coupled analysis and visualization of high resolution astrophysical simulations," in *Numerical Modeling of Space Plasma Flows (ASTRONUM-2007)*, ser. ASP Conference Series, N. V. Pogorelov, E. Audit, and G. P. Zank, Eds., vol. 385. Astronomical Society of the Pacific, 2008, pp. 299–308.

[2] J. N. Bradley and C. M. Brislawn, "Wavelet transform–vector quantization compression of supercomputer ocean models," in *Proc. Data Compress. Conf.* Snowbird, UT: IEEE Computer Soc., Mar. 1993, pp. 224–233. [Online]. Available: http://dx.doi.org/10.1109/DCC.1993.253127

[3] J. N. Bradley, C. M. Brislawn, D. J. Quinlan, H. D. Zhang, and V. Nuri, "Wavelet subband coding of computer simulation output using the A++ array class library," in *Proc. Space Earth Science Data Compress. Workshop*, ser. JPL Conf. Pub., no. 95-8. Snowbird, UT: NASA, Mar. 1995, pp. 57–68. [Online]. Available: http://dx.doi.org/10.1109/DCC.1995.515564

[4] A. Trott, R. Moorhead, and J. McGinley, "Wavelets applied to lossless compression and progressive transmission of floating point data in 3-D curvilinear grids," in *Visualization '96. Proceedings.* IEEE, Nov. 1996, pp. 385–388.

[5] I. Ihm and S. Park, "Wavelet-based 3d compression scheme for very large volume data," in *Proceedings of Graphics Interface '98*, 1998, pp. 107–116.

[6] T. Kim and Y. Shin, "An efficient wavelet-based compression method for volume rendering," in *Computer Graphics and Applications, 1999. Proceedings. Seventh Pacific Conference on*, 1999, pp. 147–156.

[7] F. Rodler, "Wavelet based 3D compression with fast random access for very large volume data," in *Computer Graphics and Applications, 1999. Proceedings. Seventh Pacific Conference on*, 1999, pp. 108–117.

[8] S. Guthe, M. Wand, J. Gonser, and W. Strasser, "Interactive rendering of large volume data sets," in *Visualization Conference, IEEE*. Los Alamitos, CA, USA: IEEE Computer Society, 2002, pp. 50–60.

[9] C. Wang, J. Gao, L. Li, and H. Shen, "A multiresolution volume rendering framework for large-scale time-varying data visualization," in *Fourth International Workshop on Volume Graphics*, 2005, pp. 11–223.

[10] B. E. Usevitch, "A tutorial on modern lossy wavelet image compression: Foundations of JPEG 2000," *IEEE Signal Process. Mag.*, vol. 18, no. 5, pp. 22–35, Sep. 2001.

[11] D. S. Taubman and M. W. Marcellin, *JPEG2000: Image Compression Fundamentals, Standards, and Practice*. Boston, MA: Kluwer, 2002.

[12] C. M. Brislawn and M. D. Quirk, "Image compression with the JPEG-2000 standard," in *Encyclopedia of Optical Engineering*, R. G. Driggers, Ed. New York: Dekker, 2003, pp. 780–785.

[13] T. Acharya and P.-S. Tsai, *JPEG2000 Standard for Image Compression: Concepts, Algorithms and VLSI Architectures*. Wiley, 2004.

[14] *Information technology—JPEG 2000 image coding system, Part 1*, ser. ISO/IEC Int'l. Standard 15444-1, ITU-T Rec. T.800. Int'l. Org. Standardization, Dec. 2000.

[15] *Information technology—JPEG 2000 image coding system, Part 2: Extensions*, ser. ISO/IEC Int'l. Standard 15444-2, ITU-T Rec. T.801. Int'l. Org. Standardization, May 2004.

[16] *Information technology—JPEG 2000 Image Coding System, Part 9: Interactivity tools, APIs, and protocols*, ser. ISO/IEC Int'l. Standard 15444-9. Int'l. Org. Standardization, 2005.

[17] R. D. Smith and P. Gent, "Reference manual of the parallel ocean program (POP)," Los Alamos National Laboratory, Los Alamos, NM, Technical Report LA-UR-02-2484, 2002.

[18] "Kakadu." [Online]. Available: http://www.kakadusoftware.com

[19] D. S. Taubman, "Directionality and scalability in image and video compression," Ph.D. dissertation, Univ. of California, Berkeley, 1994.

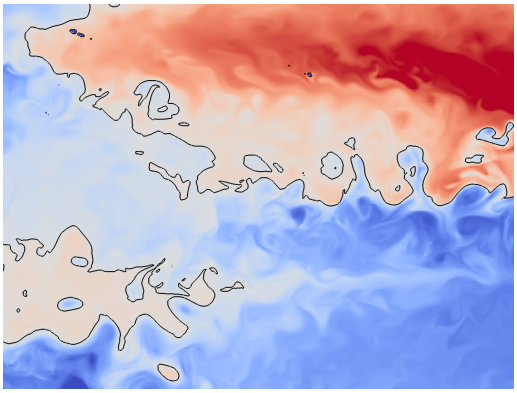
[20] "OpenJPEG." [Online]. Available: http://www.openjpeg.org

[21] "ParaView." [Online]. Available: http://www.paraview.org

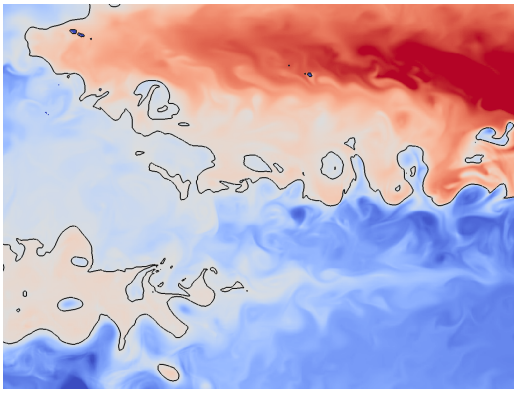
[22] S. Williams, M. Hecht, M. Petersen, R. Strelitz, M. Maltrud, J. Ahrens, M. Hlawitschka, and B. Hamann, "Visualization and analysis of eddies in a global ocean simulation," in *Proc. IEEE Symp. on Visualization (EuroVis2011)*, Eurographics Assoc. Bergen, Norway: Blackwell, Jun. 2011.

[23] J. Woodring, S. Mniszewski, C. Brislawn, D. DeMarle, and J. Ahrens, "Revisiting wavelet compression for large-scale climate data using JPEG 2000 and ensuring data precision," in *Proc. IEEE Symp. on Large Data Analysis and Visualization*. Providence, RI: IEEE Computer Society, Oct. 2011, pp. 31–38. [Online]. Available: http://dx.doi.org/10.1109/LDAV.2011.6092314

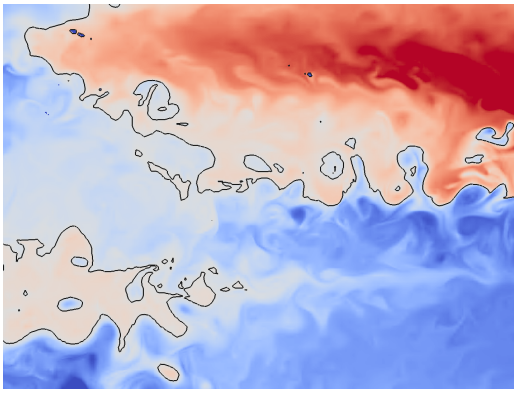
[24] *Information technology—JPEG 2000 image coding system, Part 10: Extensions for three-dimensional data*, ser. ISO/IEC Int'l. Standard 15444-10, ITU-T Rec. T.809. Int'l. Org. Standardization, 2008.



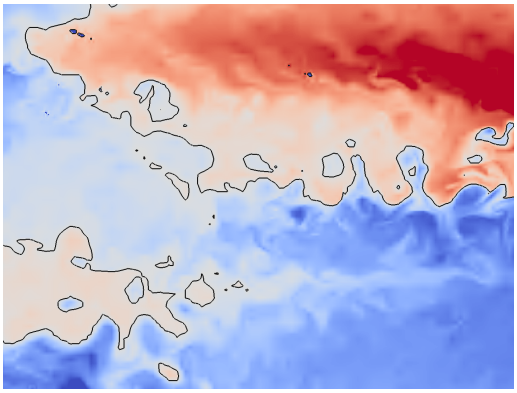
(a) 8 bits/pixel; max error = 1.49e-09



(b) 1 bit/pixel; max error = 2.31e-05



(c) 0.5 bits/pixel; max error = 8.59e-05



(d) 0.25 bits/pixel; max error = 3.03e-04

Fig. 5. Isocontours on salinity data reconstructed at various bit rates.