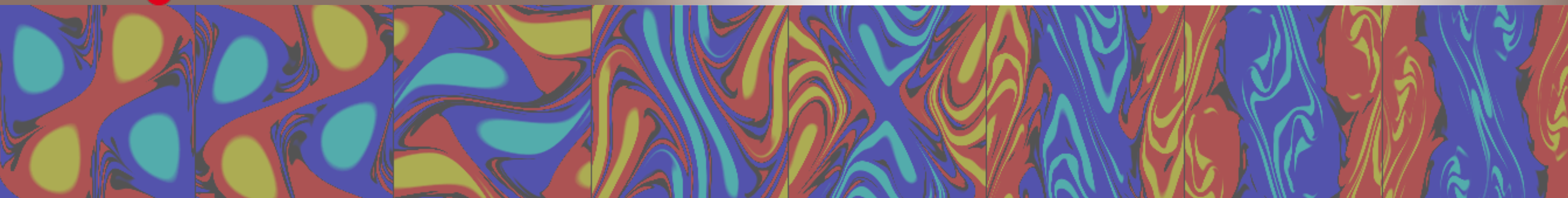
ISC High Performance
The HPC Event.

ISC 2017 Workshop
In Situ Visualization:
Introduction & Applications



WAVELETS AS AN IN SITU COMPRESSION OPERATOR FOR POST HOC EXPLORATION

SHAOMENG (SAMUEL) LI, NCAR/UO
JOHN CLYNE, NCAR
HANK CHILDS, UO

June 22, 2017

University of Oregon & NCAR

NCAR
NATIONAL CENTER FOR ATMOSPHERIC RESEARCH

Note

- Proposed talk was to be on in situ configurations for fusion simulations, but results did not come together in time…
- Instead: recent results by Shaomeng Li on viability of wavelets

Outline

- Context
- Study #1: tradeoffs
- Study #2: many-core
- Study #3: burst buffer

Outline

- Context
- Study #1: tradeoffs
- Study #2: many-core
- Study #3: burst buffer

Where this work fits…

- Traditional paradigm: saving full time slices at regular or irregular intervals.
- The traditional paradigm is jeopardized by the I/O gap.
- In response, there are three main strategies.

Strategy #1: Save less often

- Saving fewer time slices to compensate for the decrease in I/O power → temporal sparsity

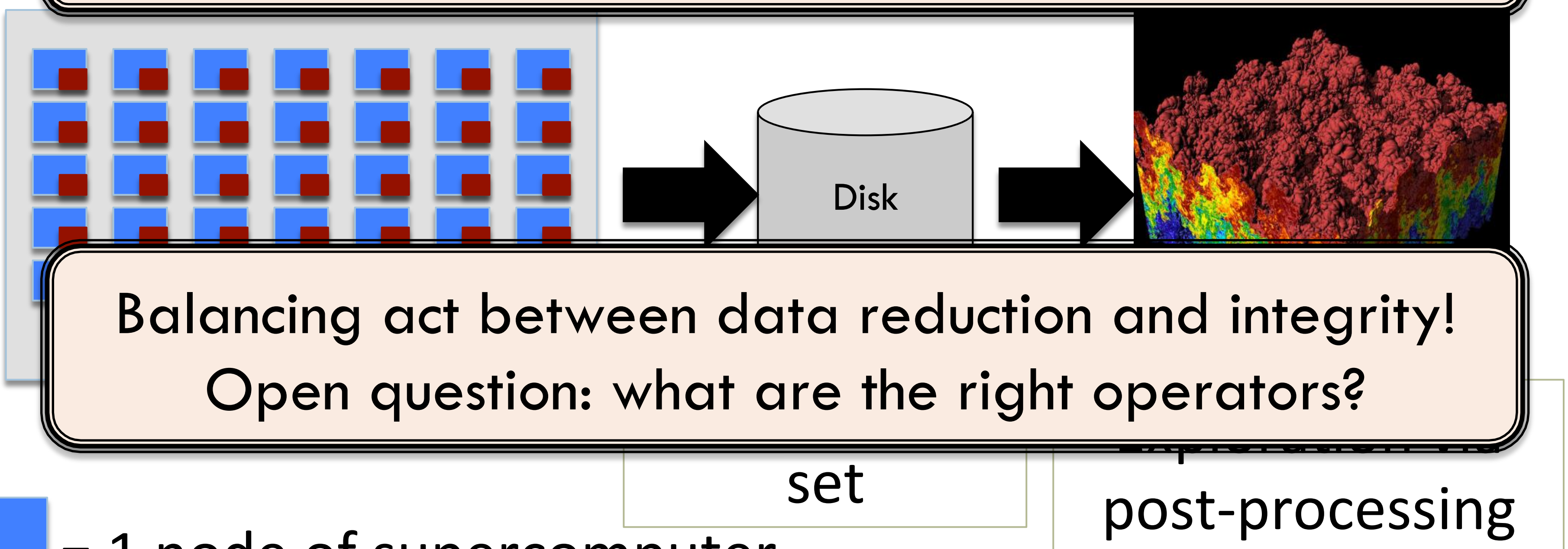
- Problems with temporal sparsity:
  - limits the discovery of phenomena
  - lack the proper temporal resolution to be observed

Strategy #2: Apply viz techniques in situ

- When it works:
  - scientists know what they want to see prior to the simulation running
- When it does not work:
  - explorative visualization requires saving data for post-hoc processing
    - (no a priori info)

Strategy #3: transform + reduce in situ, explore post hoc

This strategy is the motivation for our study.



Balancing act between data reduction and integrity!

Open question: what are the right operators?

= 1 node of supercomputer

= in situ data reduction routine

Wavelets 101

- $x[n]$: data array of size K
- $u_k[n]$: a set of K basis functions (wavelet kernel)
- A *linear combination* of K basis functions to represent $x[n]$ (<u>wavelet transform</u>)

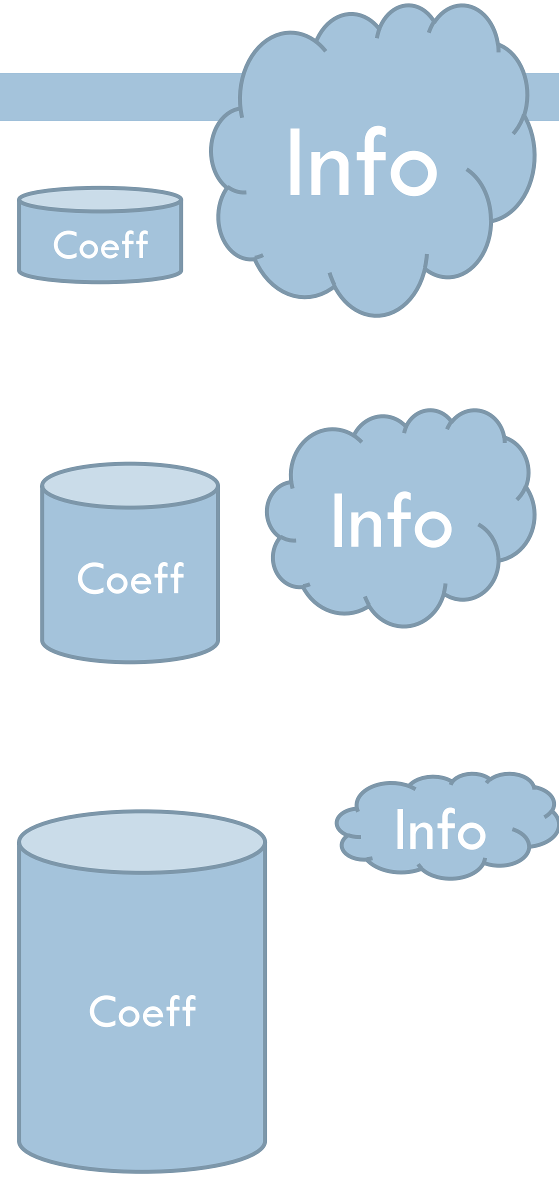
coefficients

$$x[n] = \sum_{k=0}^{K-1} a_k \cdot u_k[n]$$

- <u>Input data $x[n]$ and output coefficients a_k have same size</u>: K
- Reconstructing $x[n]$:
  - Use all K coefficients: *lossless & no compression*
  - Use a subset of coefficients: *lossy & compression*
  - <u>Coefficient prioritization</u> to select the most information-rich coefficients

Why use wavelets for compression?

- Excellent at energy (information) compaction
  - Data tends to have coherence (unless complete randomness)
  - Wavelet transforms decorrelate the coherence
- Easy to extend to higher dimensions
  - Wavelet transforms are essentially 1D
  - Simply apply the same 1D transform independently along each dimension
- Low complexity: linear
  - Core calculation: discrete convolution



Outline

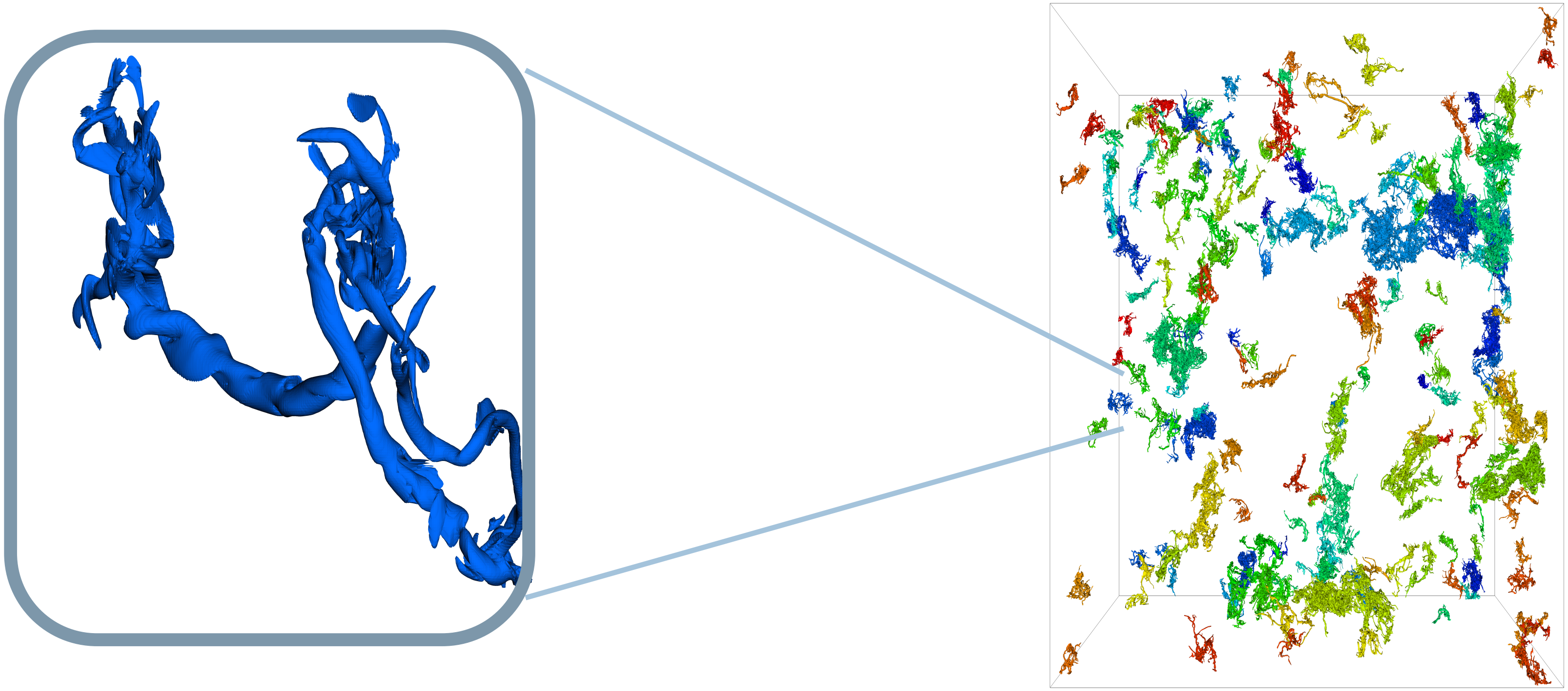
- Context
- Study #1: tradeoffs
- Study #2: many-core
- Study #3: burst buffer

Study #1: tradeoffs between accuracy and data redution

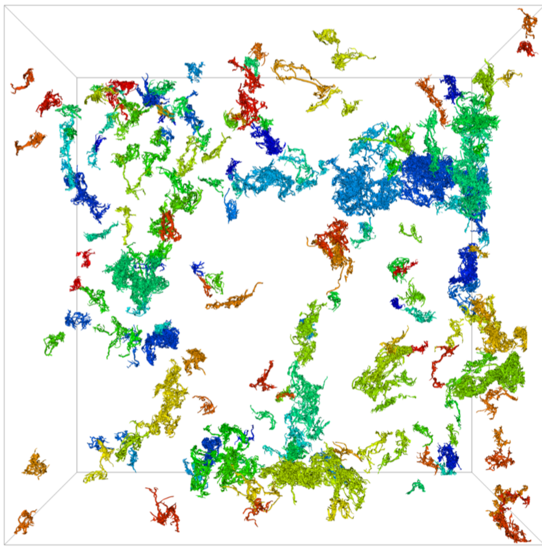
- Comparisons:
  - Traditional usage of wavelets in vis (multi-res + Haar kernel)
  - Best practices from image processing (prioritized coefficients + CDF9/7)
- Evaluation:
  - Traditional: apply wavelets to a bunch of scientific data sets, measure resulting error
  - Additional: consider two scivis use cases

Application 1 -- Critical structure identification

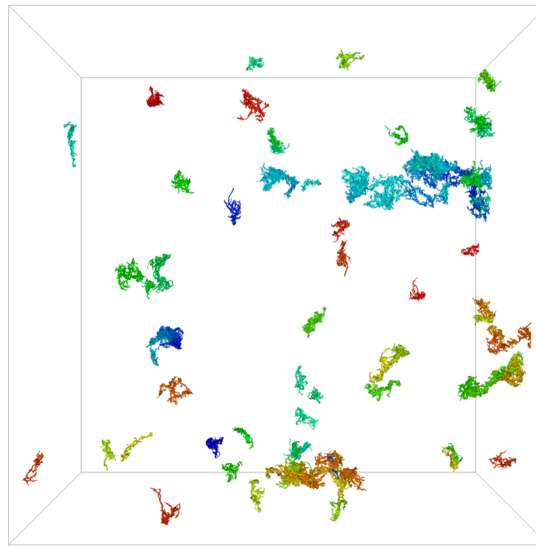
- Turbulent-flow data set at $4{,}096^3$
- Critical structure: a region with higher enstrophy values than its surroundings
- Threshold the structure volume (millions → hundreds)



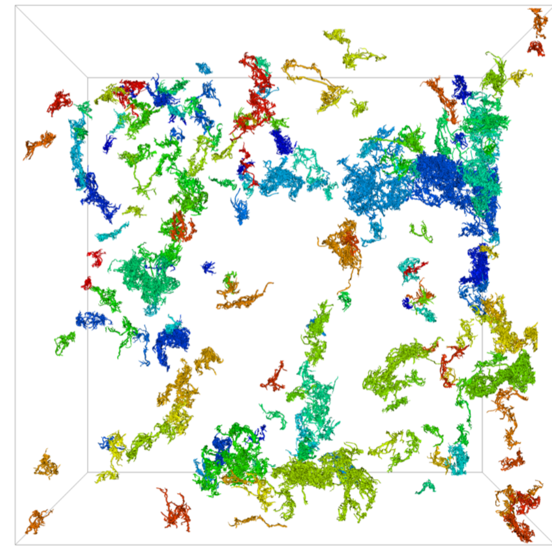
Application #1: Comparison



Original data



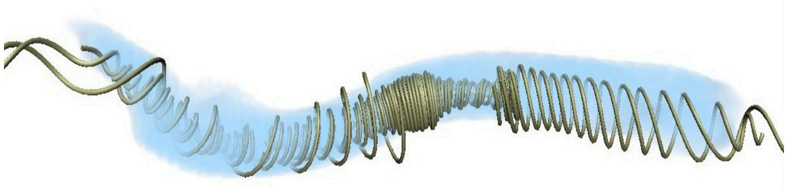
8:1 compression,
Haar + multi-res



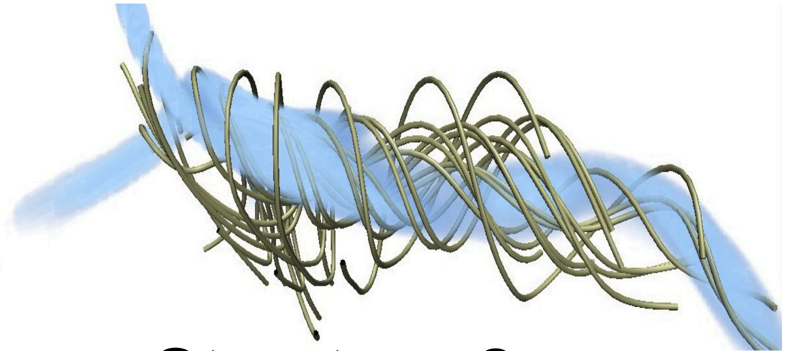
128:1 compression,
CDF 9/7 + prioritized
coefficients

S. Li, K. Gruchalla, K. Potter, J. Clyne, and H. Childs. Evaluating the Efficacy of Wavelet Configurations on Turbulent-Flow Data. In Proceedings of IEEE Symposium on Large Data Analysis and Visualization, Oct. 2015.

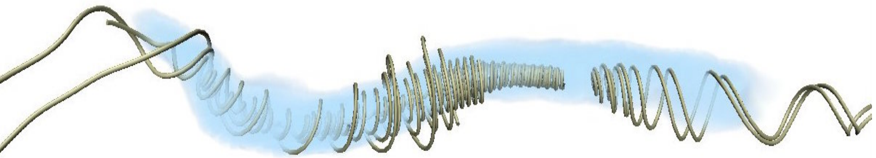
Application 2 -- Results



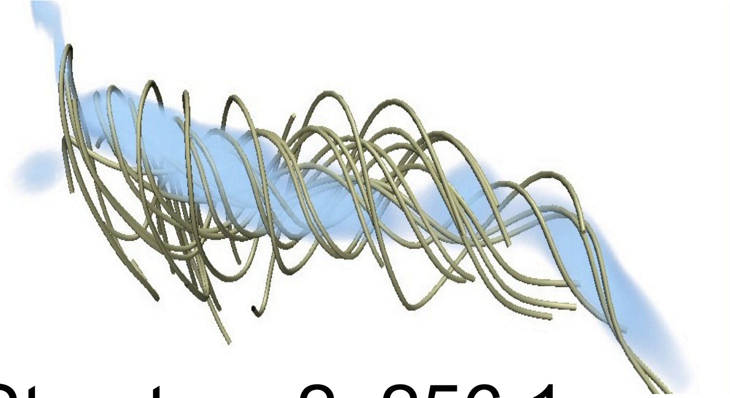
Structure 1,
baseline



Structure 2,
baseline



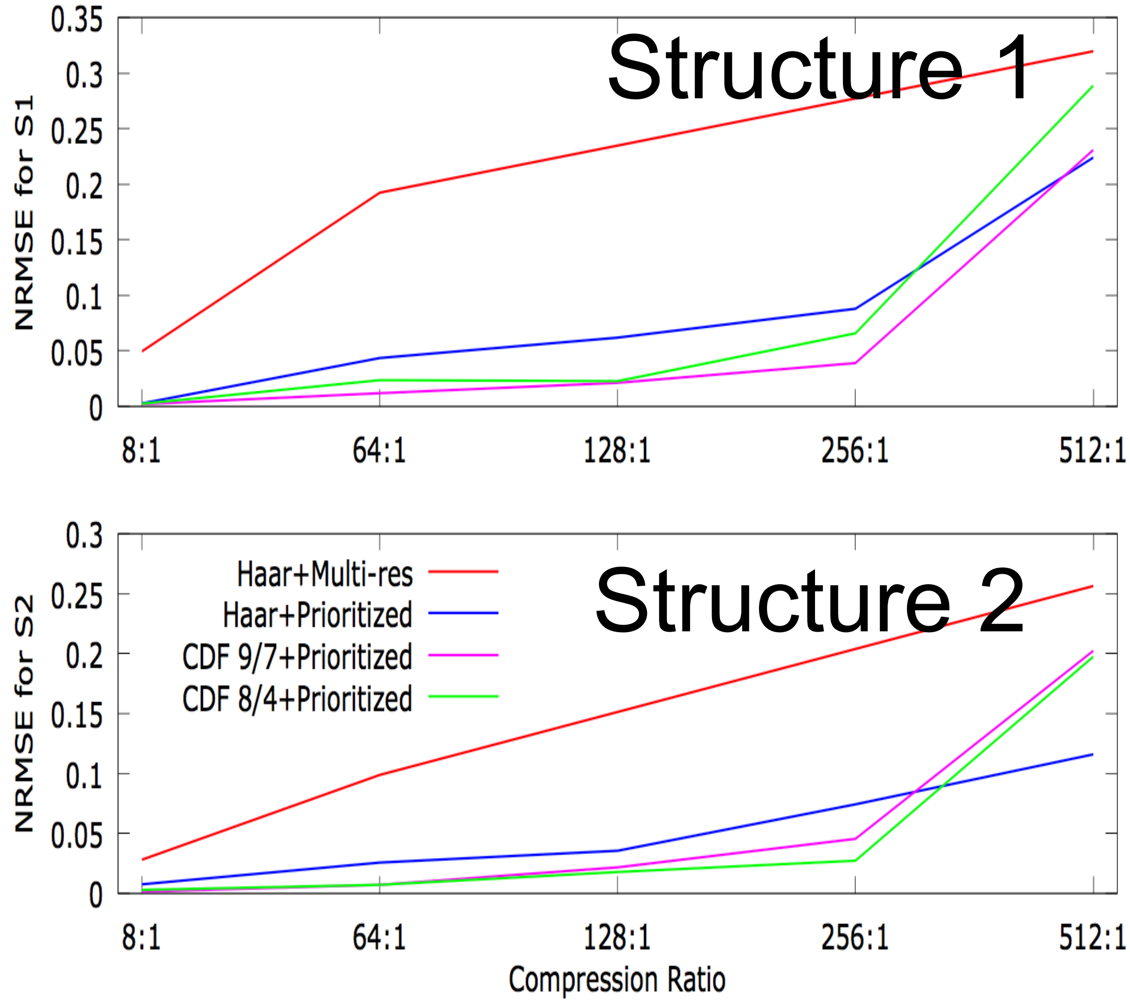
Structure 1, 128:1



Structure 2, 256:1

Application 2 – Evaluation Results

- Visualizations show some differences
- Errors stay within 5% for both structures at 256:1



Outline

- Context
- Study #1: tradeoffs
- Study #2: many-core
- Study #3: burst buffer

How to Achieve Portable Performance

- ~~See next talk!~~
- Idea:
  - Declare fixed set of primitives
    - *Not*: while, for, etc.
    - *Instead*: map, reduce, gather, scatter, etc.
  - Then design vis algorithms using primitives
    - I.e., use "map" instead of "for"
  - Experts worry about fast "map" (or whatever) for GPU, CPU, etc
  - This is the idea behind VTK-m…
- On to the highlights…

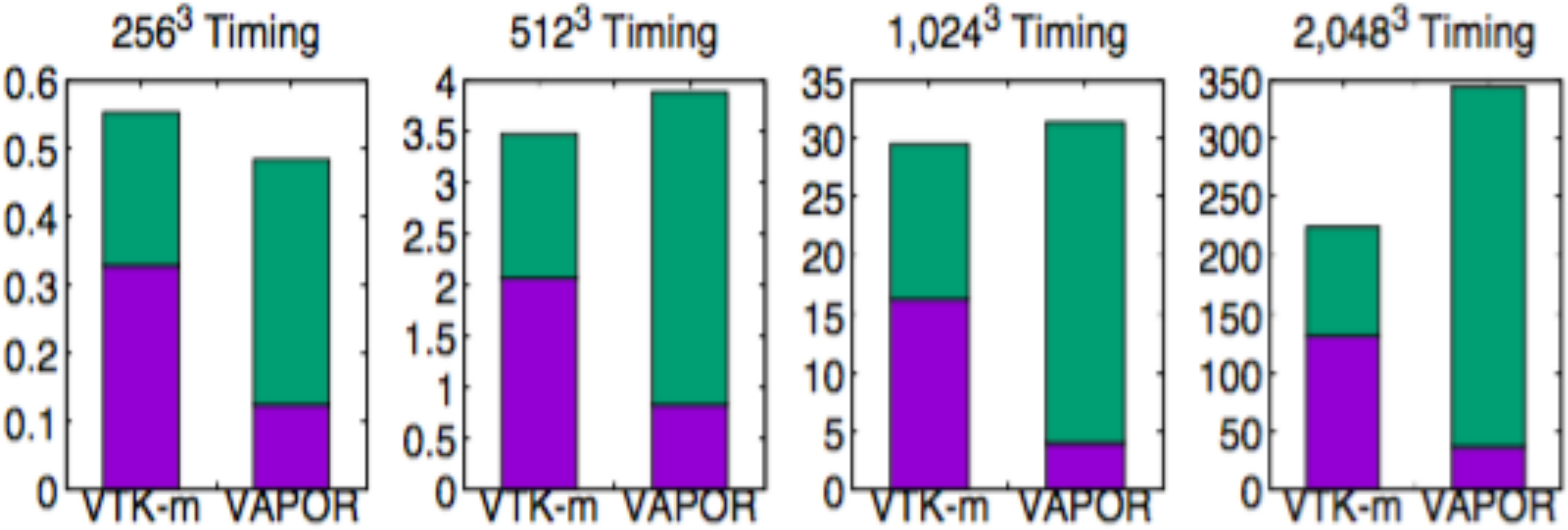
Algorithm 1 Worklet for 3D Wavelet Transform in the X Axis

Input: $signal, workIndex$ {Assigned by VTK-m}
Output: $coefficients$

$(x, y, z) \leftarrow$ GetLogicalIndex($workIndex$)
if x is even **then**
  $arr \leftarrow$ ComposeX($signal, leftExt, rightExt, x, y, z$)
  $sum \leftarrow$ DiscreteConvolution($arr, lowWavele$
  $outIdx \leftarrow$ GetOutputIndexApproximationCoeff(x, y, z)
  $coefficients[\, outIdx \,] \leftarrow sum$
else
  $arr \leftarrow$ ComposeX($signal, leftExt, rightExt, x, y, z$)
  $sum \leftarrow$ DiscreteConvolution($arr, highWaveletFilter$)
  $outIdx \leftarrow$ GetOutputIndexDetailCoeff(x, y, z)
  $coefficients[\, outIdx \,] \leftarrow sum$
end if

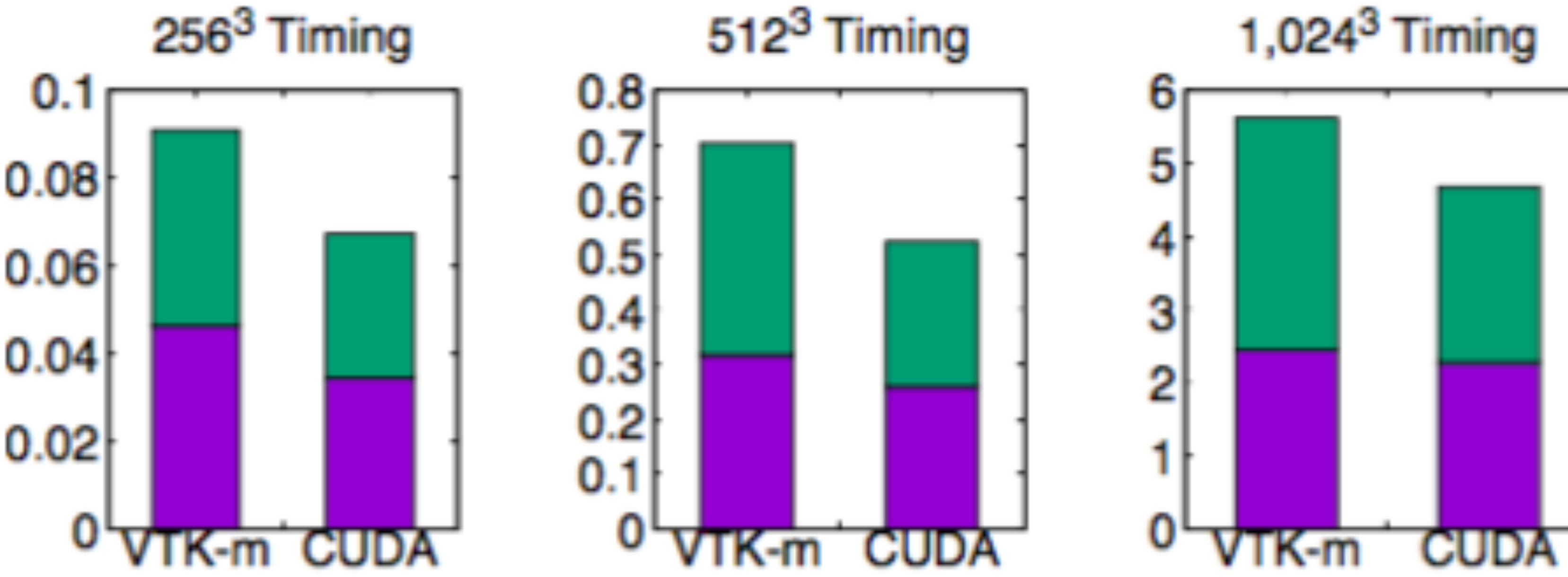
Gather
Map

CPU Results



Purple = transform time
Green = sort time

GPU Results



Purple = transform time
Green = sort time

Important

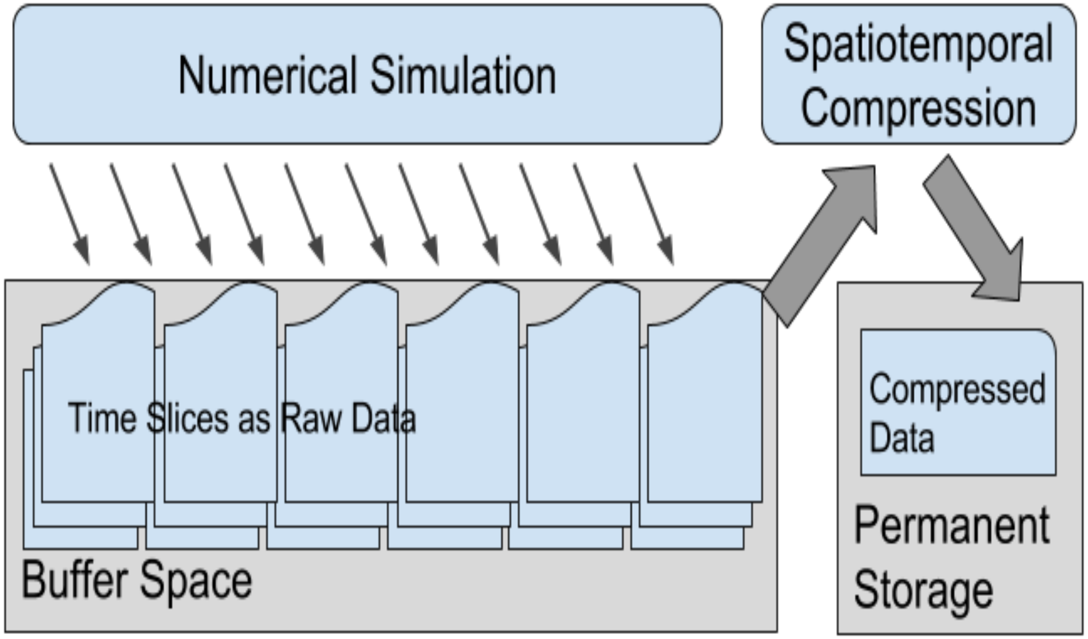
- Wavelet compression times are short -- ~0.5s
- Means that we likely fit within in situ constraints

Outline

- Context
- Study #1: tradeoffs
- Study #2: many-core
- Study #3: burst buffer

Wavelet compression with deeper memory hierarchy

- Idea: use SSDs/burst buffers to temporarily store a few times slices, and apply temporal compression on them together
- Summary: 4D wavelet compression brings 1.5X~2X improvements over 3D compression



Talk Summary

- Wavelets are a viable in situ compression operator
- Improved understanding for:
  - tradeoffs in reduction and integrity
  - how to implement for multiple architectures
  - fitting within in situ constraints
  - how to leverage burst buffer to get more information per byte

Acknowledgments

- Thanks to these funding agencies
  - US Department of Energy
    - Early Career Award, Program Manager Lucy Nowell
    - Xvis, Program Manager Lucy Nowell
    - SciDAC SDAV, Program Manager Ceren Susut-Bennett

Questions?

WAVELETS AS AN IN SITU COMPRESSION OPERATOR FOR POST HOC EXPLORATION

SHAOMENG (SAMUEL) LI, NCAR/UO
JOHN CLYNE, NCAR
HANK CHILDS, UO