Best Practices: Large Scale Multiphysics

Frank Ham & Sanjeeb Bose, Cascade Technologies

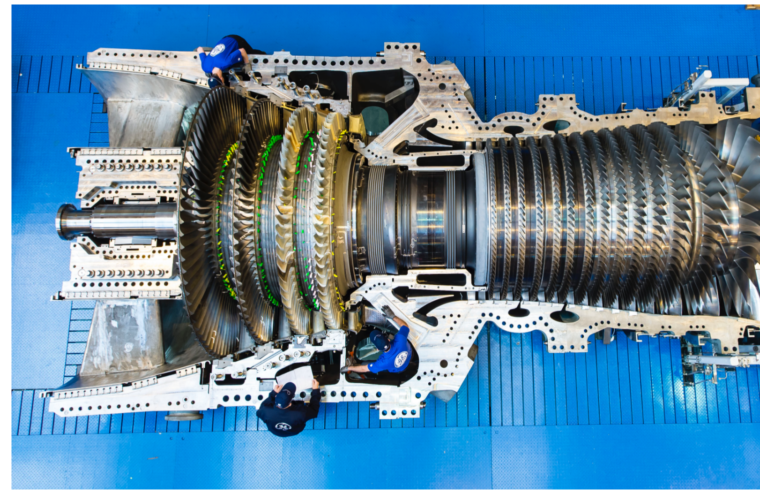
L Shunn, G Bres, D Philips, M Emory, A Saghafian, D Kim, B Hejazi, P Quang [Cascade]

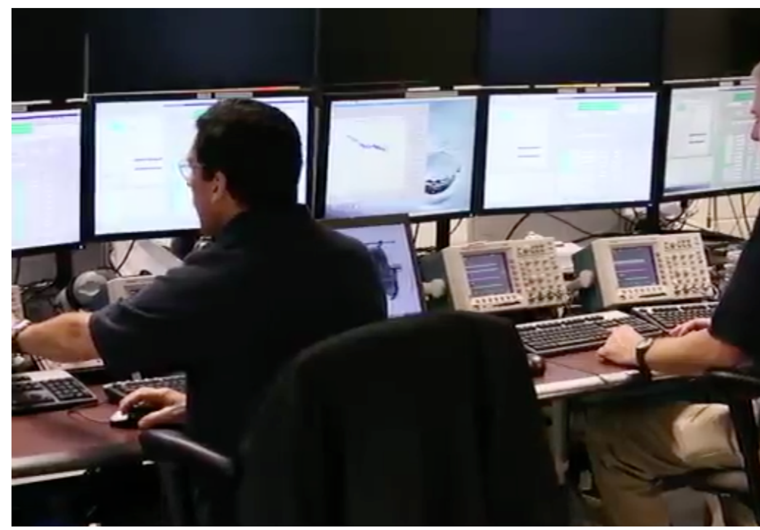
CASCADE TECHNOLOGIES

Acknowledging financial, technical, or computational support from: NASA, Navair, Dept of Energy Office of Science, Oak Ridge National Labs, Argonne National Labs, ERDC (DoD), GE

Motivation: A multiphysics problem
Gas Turbine Self-Excited Dynamics (SED)

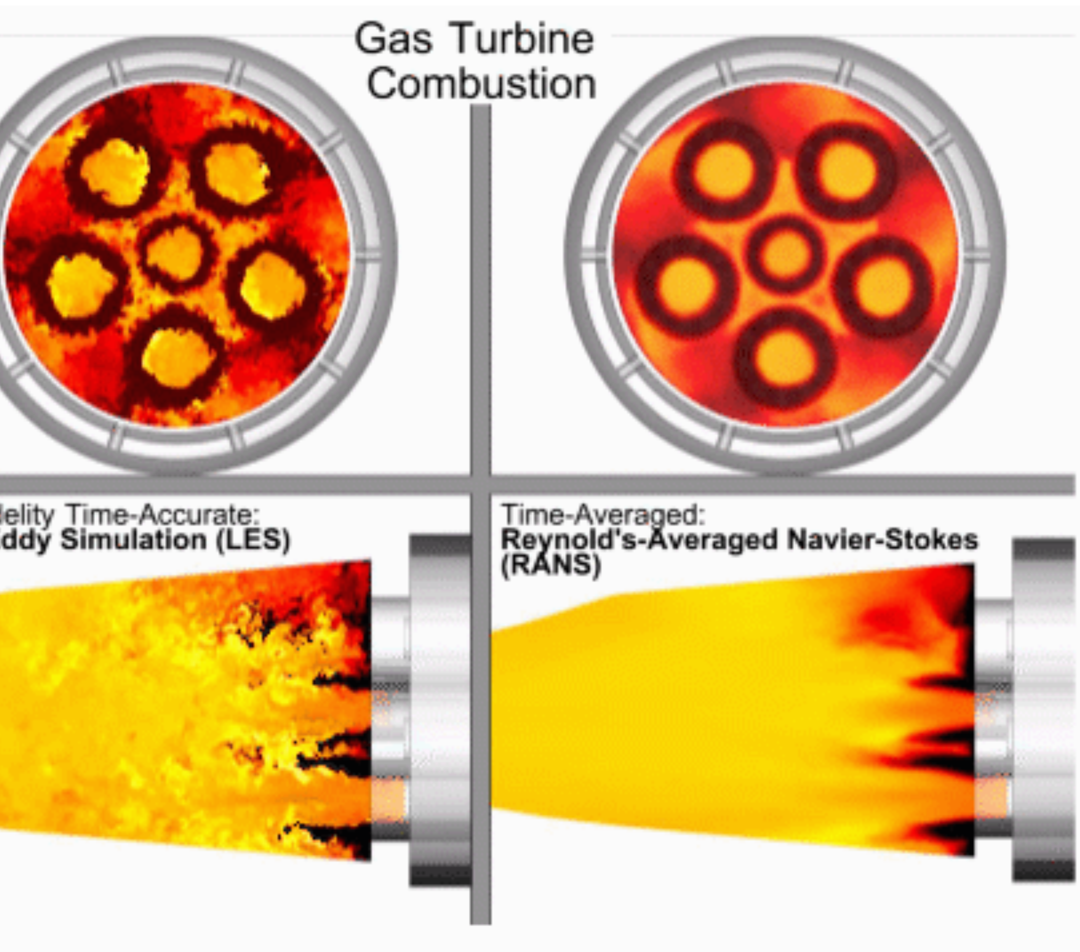
- In 2014, GE observed combustion dynamics during a full-scale test of a gas turbine
- The particular frequency had not been predicted by lower-fidelity models, and had not been observed in single can combustor testing
- Amplitudes were not a concern, and could be managed with fueling adjustments, but important questions remained:
  - What caused the observed dynamics?
  - Would it manifest in a new engine configuration with new technology scheduled for testing in late 2015?





2015 Press Release: Super Collaboration

GE and Cascade Apply New Technologies and Super-Computer Capabilities to Improve Gas Turbine Combustion



GE Press Release June 22, 2015

"The enhanced simulation and visualization capabilities enable by our collaboration with Cascade can help us deliver eve higher efficiency and lower emissions in the next generatio of gas turbines,"

John Lammas, vice president, power generation engineering GE Power and Water

The timeline:

Simulating Gas Turbine Self-Excited Dynamics (SED)

2015

2016

4 full speed
load test
covers
expected
ability

Joint Development Agreement announced between GE and Cascade to develop simulation to improve Gas Turbine Combustion

2015 test planned for new engine configuration including new technology

The timeline:

Simulating Gas Turbine Self-Excited Dynamics (SED)

Lots of hard work

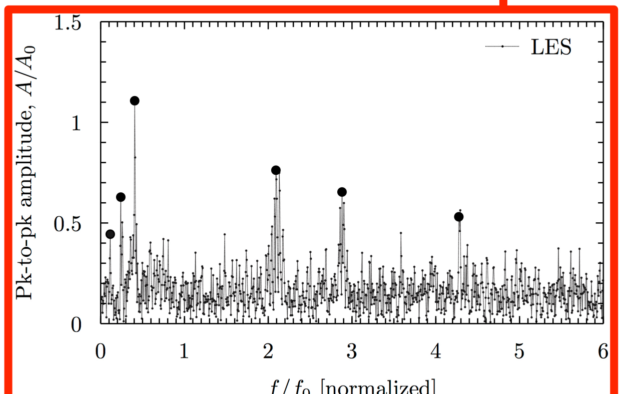
2015

2016

4 full speed load test covers expected tability

Joint Development Agreement announced between GE and Cascade to develop simulation to improve Gas Turbine Combustion

Pre-test prediction one month prior to scheduled full-speed full load testing



2015 test planned for new engine configuration including new technology

HPC Partnerships: critical for success stories
(and the likely model for early industrial use of exascale)

- Owner of the problem/challenge
- Business urgency (read $)
- Deep domain-specific knowledge
- Compelling energy and US competitive arguments

- Owner of the hardware (Titan)
- Aligned with energy and US competitive argument

- Owner of the software
- HPC Software/Multiphysics/ Numerics development team able to move fast and touch all parts of the simulation workflow

The full story

ttps://www.olcf.ornl.gov/2016/05/31/better-combustion-for-power-generation/

SCIENCE - Written by Jonathan Hines on May 31, 2016

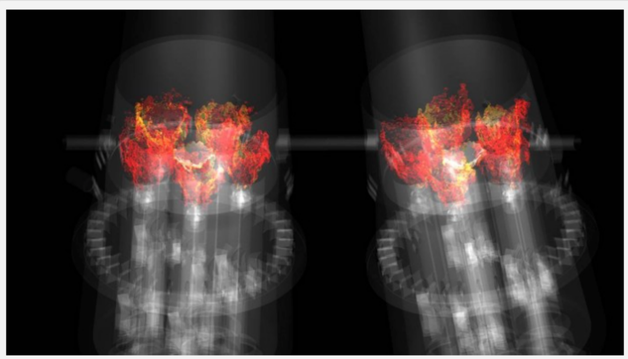
Better Combustion for Power Generation

Tags: ACCEL, Combustion, GE Global Research, Industrial Partnerships, Industry, Supercomputing

SHARE

Titan propels GE beyond the limits of gas turbine testing

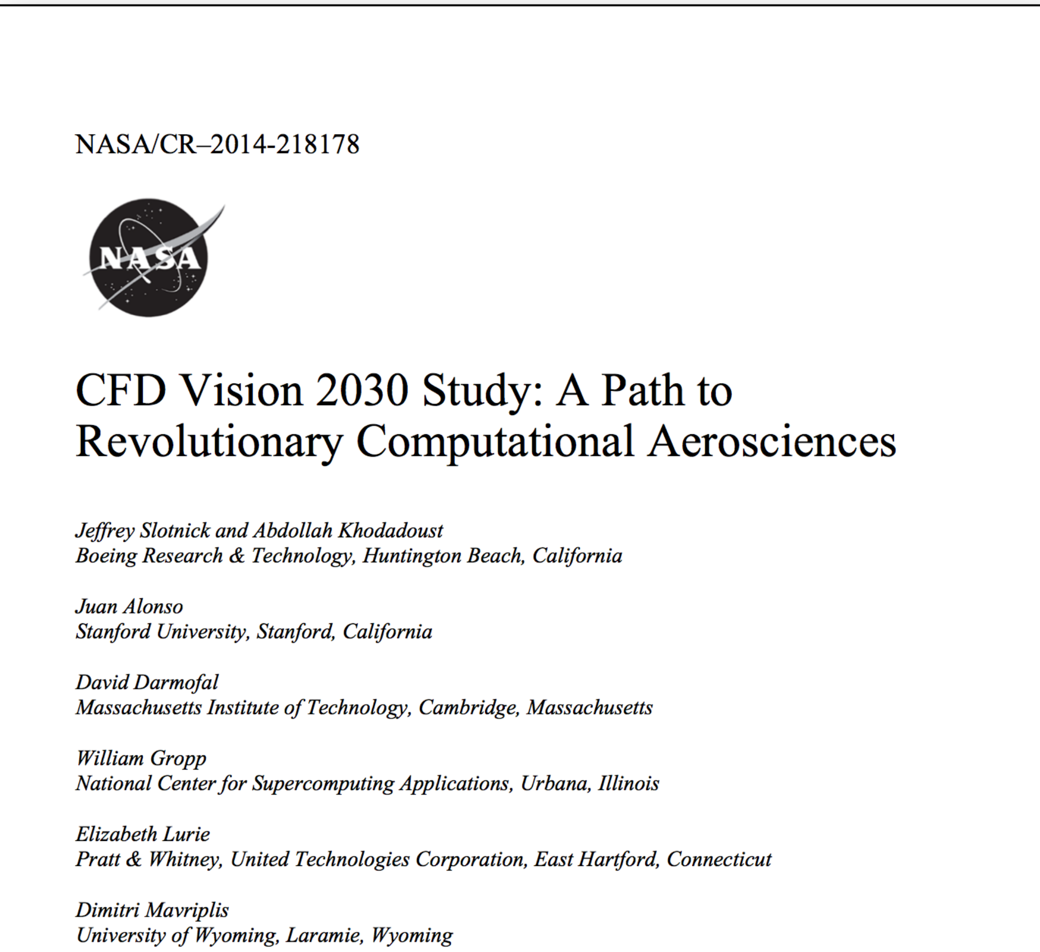
In the United States, the use of natural gas for electricity generation continues to grow. The driving forces behind this development? A boom in domestic natural gas production, historically low prices, and increased scrutiny over fossil fuels' carbon emissions. Though coal still accounts for about a third of US electricity generation, utility companies are pivoting to cleaner natural gas to replace decommissioned coal plants.



A simulation of combustion within two adjacent gas turbine combustors. GE researchers are incorporating advanced combustion modeling and simulation into product testing after developing a breakthrough methodology on the OLCF's Titan supercomputer. Image courtesy of Cascade Technologies

Revolutionary Computational Aerosciences

5 revolutions required…


NASA/CR–2014-218178

CFD Vision 2030 Study: A Path to Revolutionary Computational Aerosciences

Jeffrey Slotnick and Abdollah Khodadoust
Boeing Research & Technology, Huntington Beach, California

Juan Alonso
Stanford University, Stanford, California

David Darmofal
Massachusetts Institute of Technology, Cambridge, Massachusetts

William Gropp
National Center for Supercomputing Applications, Urbana, Illinois

Elizabeth Lurie
Pratt & Whitney, United Technologies Corporation, East Hartford, Connecticut

Dimitri Mavriplis
University of Wyoming, Laramie, Wyoming

- Efficient deployment on next-generati computer hardware and algorithmic innovation
- Complex geometries, mesh generatior and adaptation remain a bottleneck
- Wall modeling, and current inability predict separated turbulent flows
- The need for more than a single prediction: sensitivity analysis, design optimization, UQ, and the consequent data management issues
- Multi-physics modeling – push toward constitutive models that reflect the underlying physics more closely than ever before

Revolutionary Computational Aerosciences

5 revolutions required…

NASA/CR–2014-218178

CFD Vision 2030 Study: A Path to Revolutionary Computational Aerosciences

Jeffrey Slotnick and Abdollah Khodadoust
Boeing Research & Technology, Huntington Beach, California

Juan Alonso
Stanford University, Stanford, California

David Darmofal
Massachusetts Institute of Technology, Cambridge, Massachusetts

William Gropp
National Center for Supercomputing Applications, Urbana, Illinois

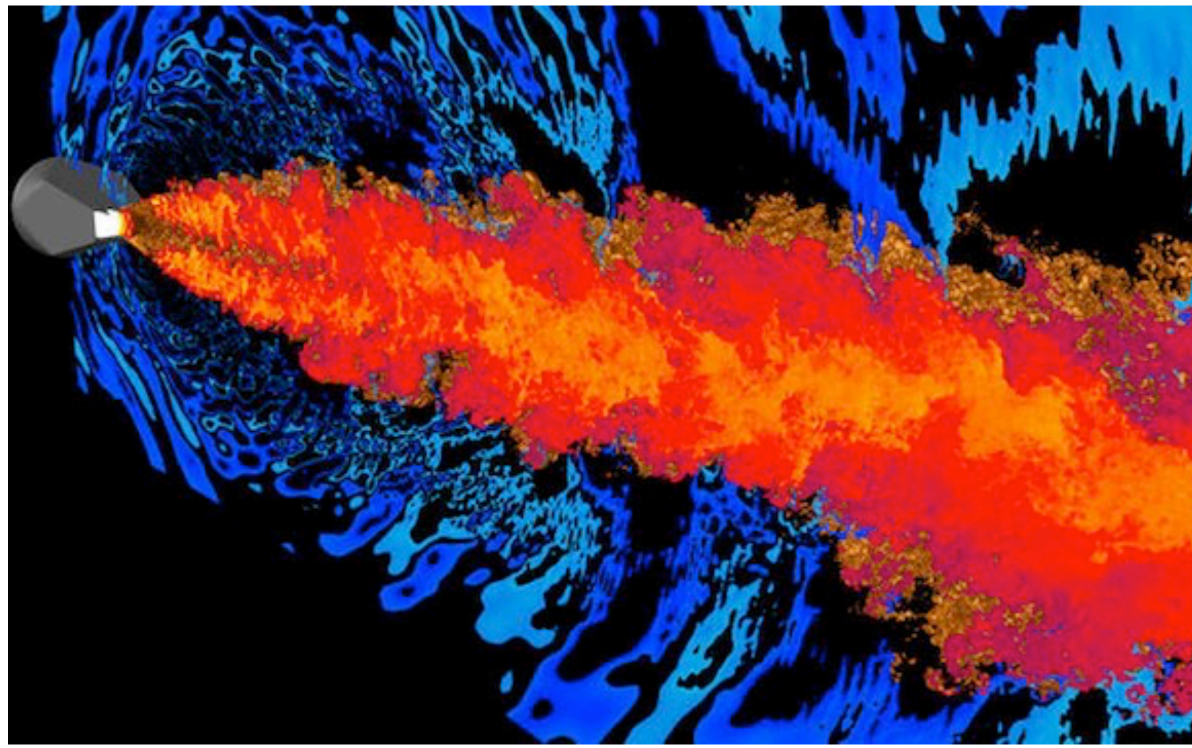
Elizabeth Lurie
Pratt & Whitney, United Technologies Corporation, East Hartford, Connecticut

Dimitri Mavriplis
University of Wyoming, Laramie, Wyoming

- Efficient deployment on next-generati computer hardware and algorithmic innovation
- Complex geometries, mesh generatio and adaptation remain a bottleneck
- Wall modeling, and current inability predict separated turbulent flows
- The need for more than a single prediction: sensitivity analysis, design optimization, UQ, and the consequent data management issues
- Multi-physics modeling – push toward constitutive models that reflect the underlying physics more closely than ever before

Starting point: Cascade's CharLES solver 2015

- Finite volume method
- Explicit RK3 time advancement
- Compressible reacting flow
- Flamelet-based tabulation of EOS
- Arbitrary unstructured grids with parMetis-based domain-decomposition
- Significant verification and validation



Pressure and temperature field of a screeching supersonic je
Image from "Stanford researchers break million-core supercomputing barrier".

Can we do grid generation on the HPC resource?

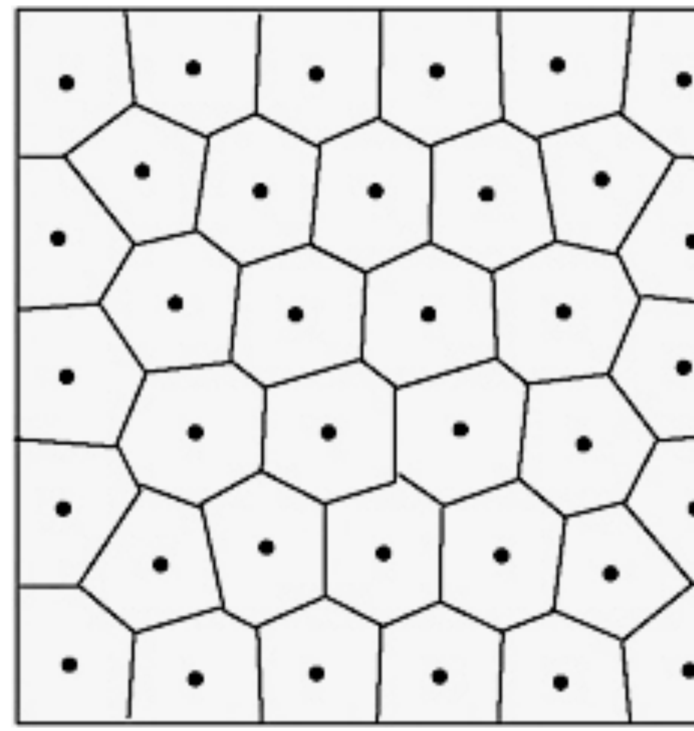
Generating grids is still a bottleneck for most solvers working with geometries of engineering relevance

Grids not going away for turbulent flows: problems are non-smooth and require energy estimates to ensure stability

Grid generation/optimization should ultimately be part of the simulation process (to minimize grid-related only known during the simulation)

What about Voronoi diagrams?

Clipped Voronoi Diagrams



Key technology: Robust direct generation of the clipped Voronoi diagram inside an arbitrary hull

Fully and uniquely defines the the mesh from a set of points (connectivity, volumes, normals)

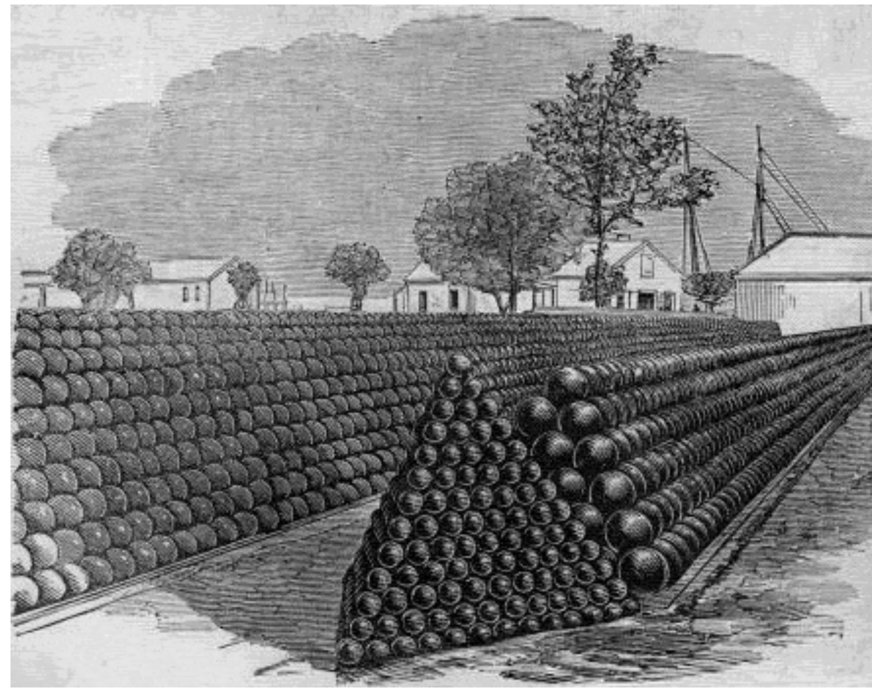
To modify the mesh, move, add, or remove points

mesh changes are continuous and differentiable with point motion

Voronoi diagram has compact support, and can thus be essentially “rendered” from the given point set

Voronoi Generating Points

- HCP = hexagonal close-packed lattice
- Leads to centroidal Voronoi meshes of self-similar truncated octahedra
- User specifies regions/windows and desired point density
- HCP mesh inserted using fast scanning approach
- Similar to pixel “rendering”
- Parallel, scalable

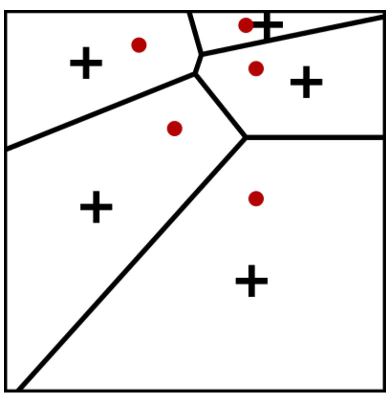


Optimal sphere packing is the motivation for HCP-lattice based Voronoi diagrams

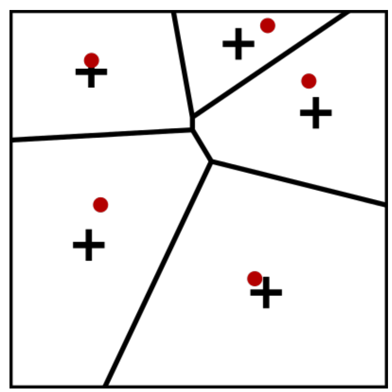
Boundary Recovery using Lloyd Iteration

Lloyd iteration is a robust and powerful mesh smoothing capability for Voronoi diagrams that can be used to:

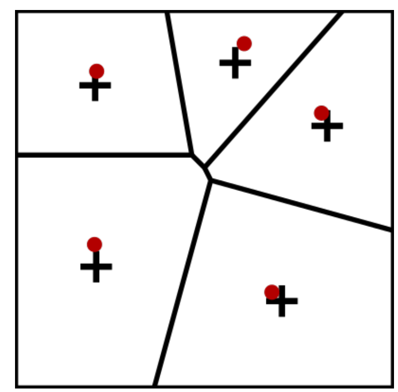
- Improve the mesh quality either locally or globally
- Introduce boundary alignment



Starting mesh: Voronoi sites and mesh centroids very different

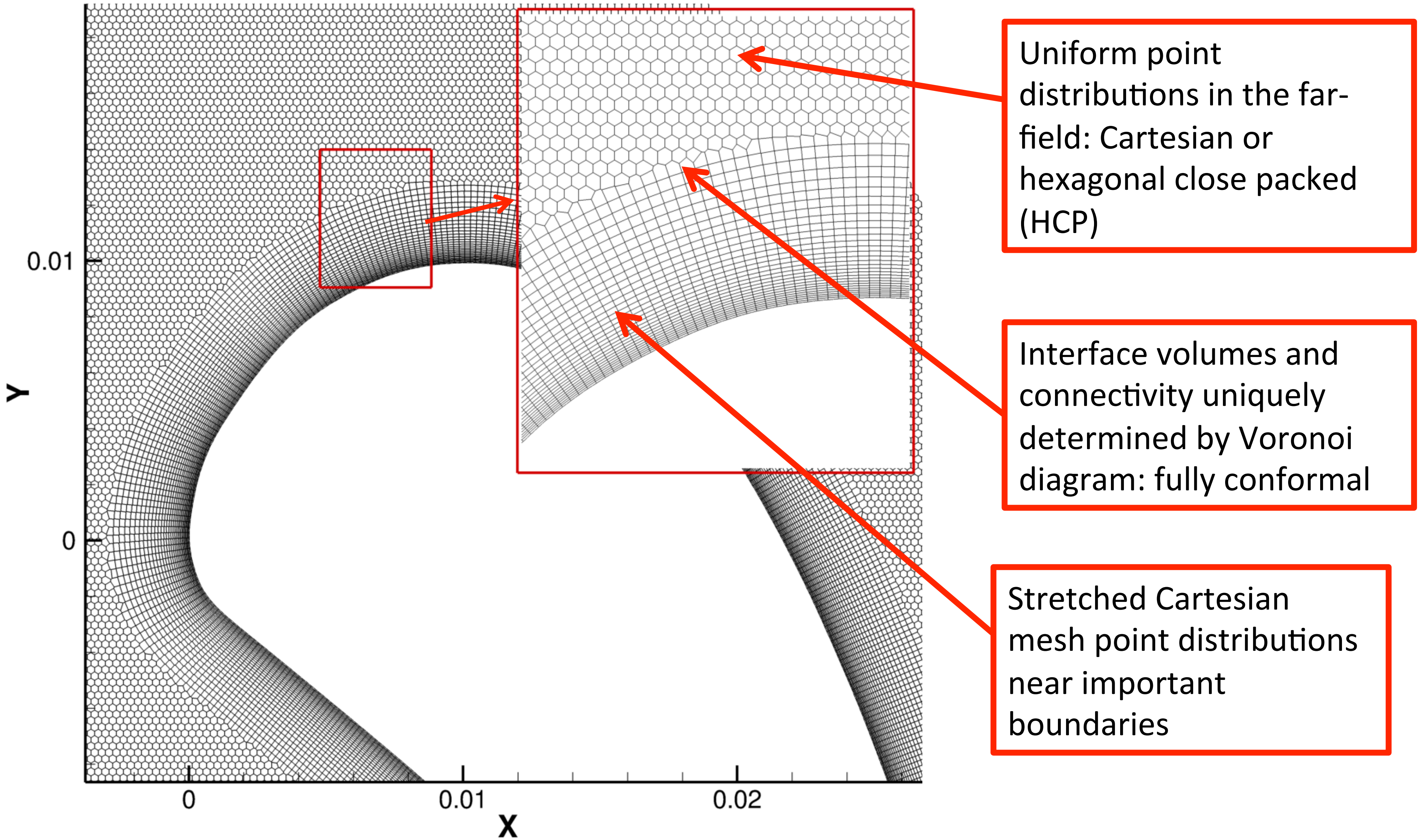


Iteration 1: regenerate Voronoi from centroids of previous mesh



Iteration 2: regenerate Voronoi from centroids of previous mesh

Example of a Voronoi Mesh around an airfoil



CPU-side solver optimizations: 1/2

- Restructuring data layout for increased cache efficiency

```
void calcInternalFlux(flux,fa[ifa],rho[icv0], rho[icv1],
                      u[icv0], u[icv1], p[icv0], p[icv1],
                      h[icv0], h[icv1], …);
```

original

```
class State {
public:
   double vol;
   double rho;
   double u[3];
   double rhoE;
   double h;
   double p;
};

void calcInternalFlux(flux,fa[ifa],cv[icv0],cv[icv1]);
```

refactored

- Aggressive overlapping computation/communication (latency hiding)
- Vectorizing instructions, aggressive inlining
- Improved partitioning algorithm (native, no longer ParMetis)

CPU-side solver optimizations: 2/2

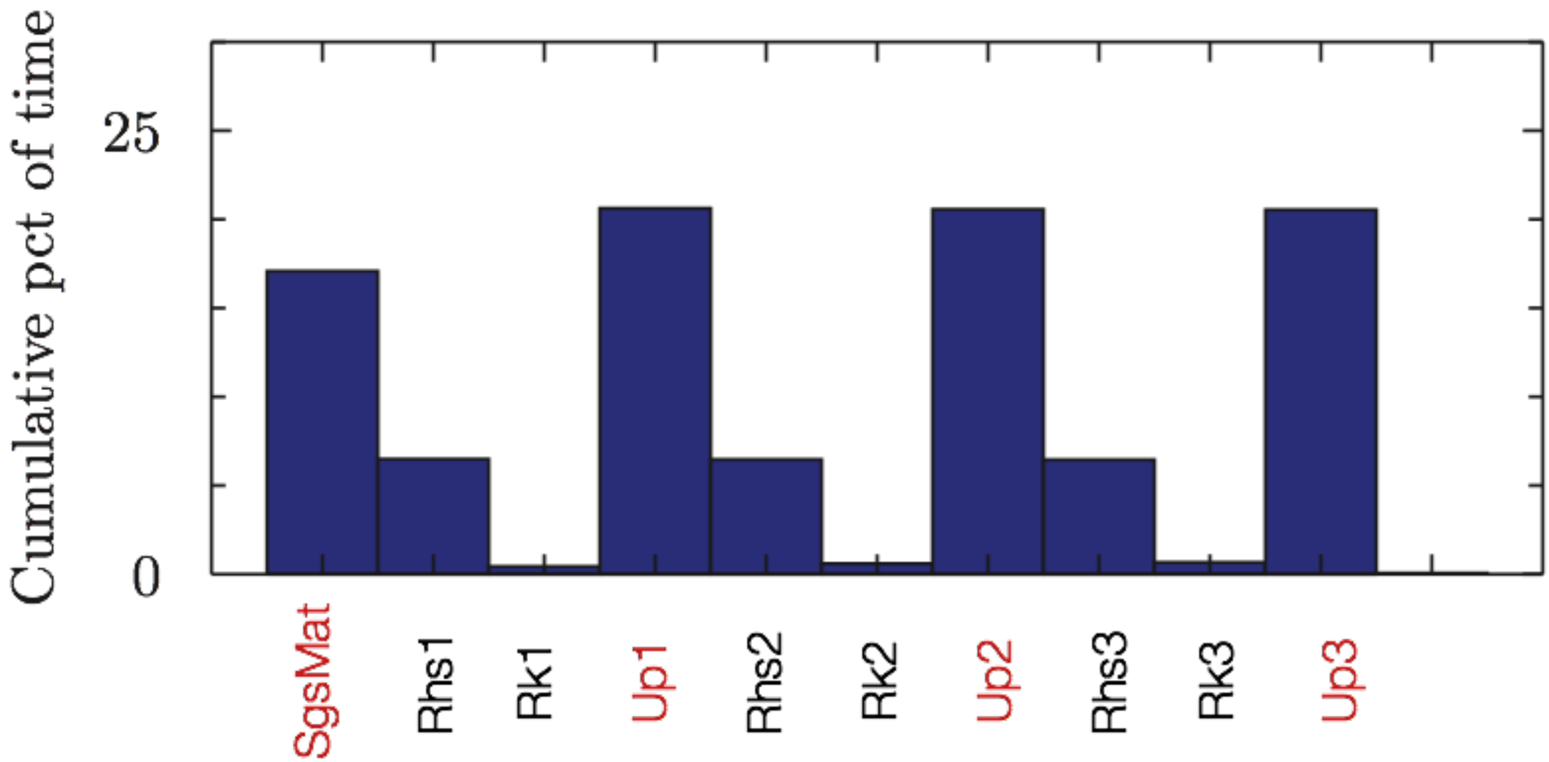
LLVM bytecode: load operations for internal flux calculation [Dec 2015

```
%2 = load double* %1, align 8, !dbg !542, !tbaa !434                      cv0.idx 1   [cv0.rho]
%5 = load double* %4, align 8, !dbg !542, !tbaa !434                      cv1.idx 1   [cv1.rho
%10 = load double* %9, align 8, !dbg !546, !tbaa !425                     cv0.idx 2,0 [cv0.u0]
%12 = load double* %11, align 8, !dbg !546, !tbaa !425                    cv1.idx 2,0 [cv1.u1]
%16 = load double* %15, align 8, !dbg !546, !tbaa !425                    cv0.idx 2,1 [cv0.u2]
%18 = load double* %17, align 8, !dbg !546, !tbaa !425                    cv1.idx 2,1
%22 = load double* %21, align 8, !dbg !546, !tbaa !425                    cv0.idx 2,2
%24 = load double* %23, align 8, !dbg !546, !tbaa !425                    cv1.idx 2,2
%38 = load double* %37, align 8, !dbg !551, !tbaa !451                    cv0.idx 9
%40 = load double* %39, align 8, !dbg !551, !tbaa !451                    cv1.idx 9
%46 = load double* %45, align 8, !dbg !555, !tbaa !556                    cv0.idx 6
%49 = load double* %48, align 8, !dbg !555, !tbaa !556                    cv1.idx 6
%59 = load double* %58, align 8, !dbg !562, !tbaa !425                    fa idx1,0
%62 = load double* %61, align 8, !dbg !562, !tbaa !425                    fa idx1,1
%66 = load double* %65, align 8, !dbg !562, !tbaa !425                    fa idx1,2
%72 = load double* %58, align 8, !dbg !564, !tbaa !425                                   fa idx1,0
%77 = load double* %61, align 8, !dbg !565, !tbaa !425                                   fa idx1,1
%82 = load double* %65, align 8, !dbg !566, !tbaa !425                                   fa idx1,2
%103 = load <2 x double>* %102, align 8, !dbg !568, !tbaa !425
%106 = load <2 x double>* %105, align 8, !dbg !568, !tbaa !425
%113 = load double* %58, align 8, !dbg !570, !tbaa !425                                  fa idx1,0
%115 = load double* %61, align 8, !dbg !570, !tbaa !425                                  fa idx1,1
%118 = load double* %65, align 8, !dbg !570, !tbaa !425                                  fa idx1,2
%125 = load double* %124, align 8, !dbg !573, !tbaa !574                   cv0.idx 10
```

Reordering instructions, algebraic factorizations, storing intermediate variables in internal flux routines to reduce repeated load instructions

Execution time breakdown

CPU-optimized compressible premixed solver



The most time-consuming operations are associated with tabulated lookups (cubic interpolation on a Cartesian grid) - move execution of these kernels to the GPU (cuda for Titan’s K20X Keplers)

Speedup from CPU-side optimizations

including flux simplifications associated with Voronoi diagrams

	≈ 25k cvs/core	≈ 3k cvs/core
Dec 2015	74	87
Feb 2016 (CPU only)	14.8	16

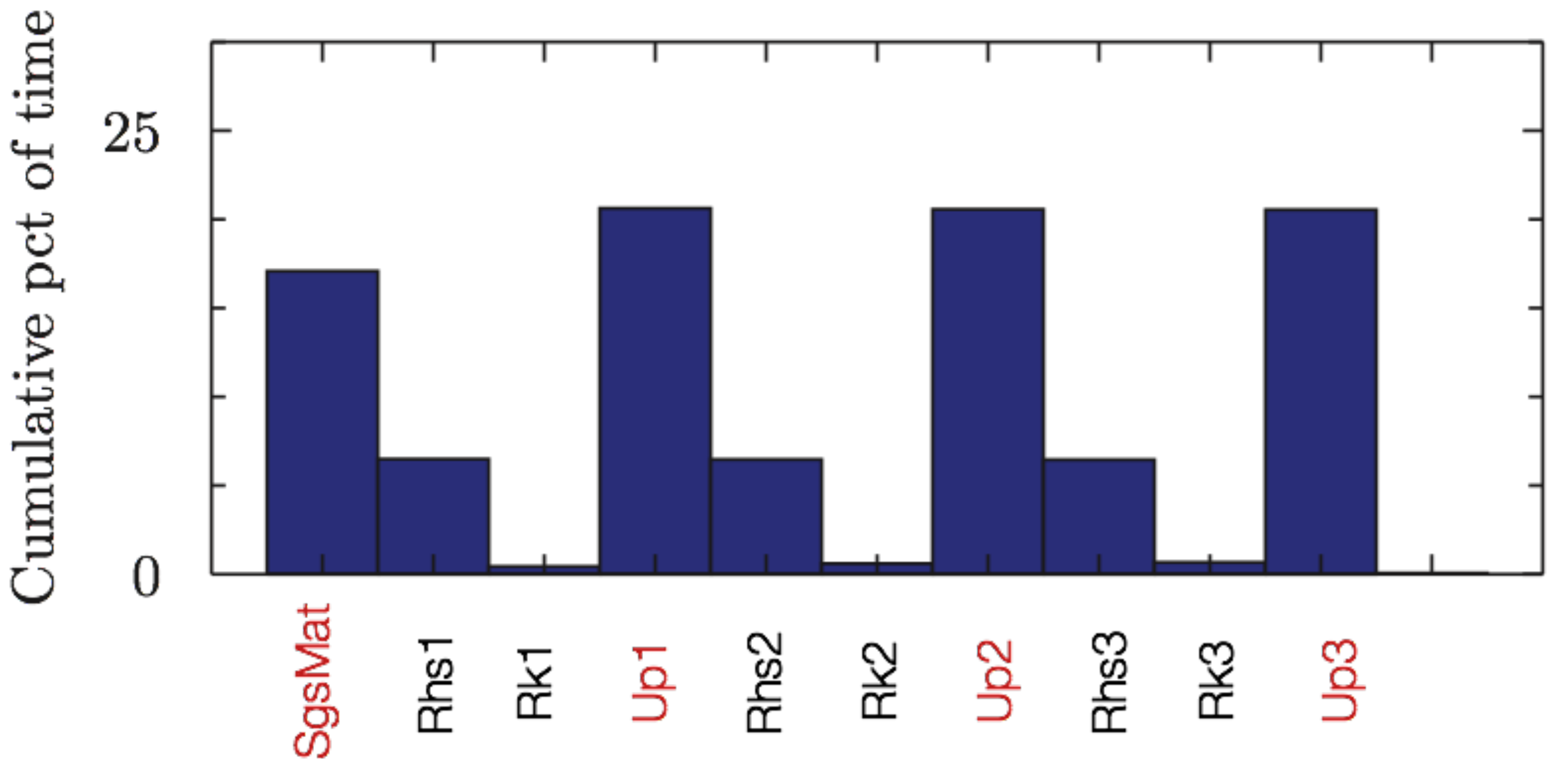
normalized timing: core-sec/M cells/step at 2 levels of strong scaling

Approximately speedup of 5X for the CPU only execution

slightly better strong scaling due to latency hiding

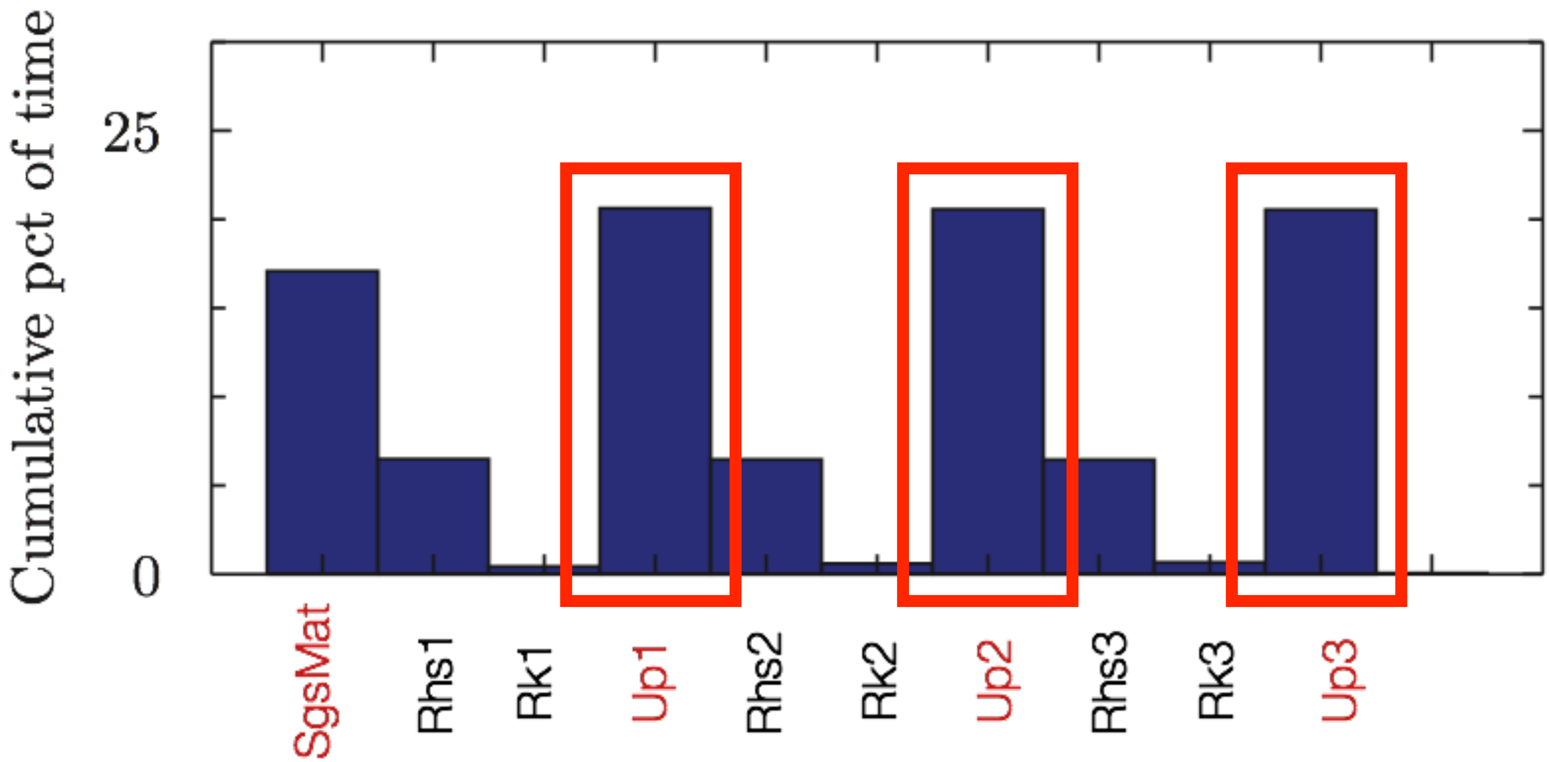
Execution time breakdown

CPU-optimized + GPU port of tabulation



The most time-consuming operations are associated with tabulated lookups (cubic interpolation on a Cartesian grid) - move execution of these kernels to the GPU (cuda for Titan's K20X Keplers)

Execution time breakdown
CPU-optimized + GPU port of tabulation



The most time-consuming operations are associated with tabulated lookups (cubic interpolation on a Cartesian grid) - move execution of these kernels to the GPU (cuda for Titan's K20X Keplers)

Speedup from CPU+GPU optimizations

GPU chemistry tabulation

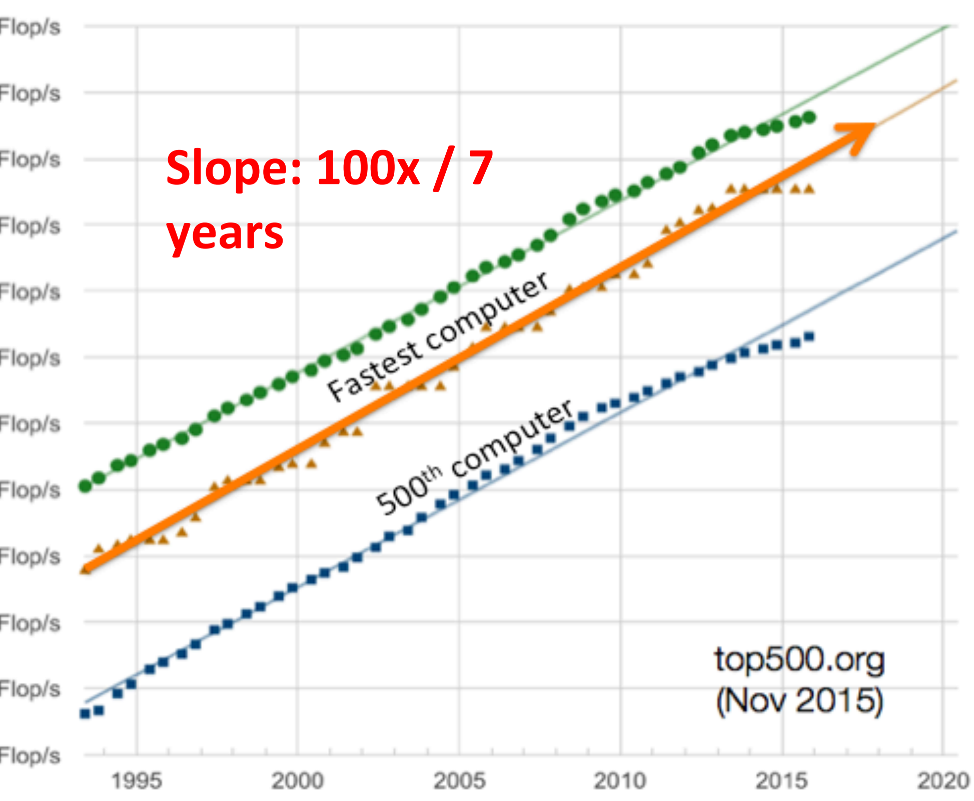
	≈ 25k cvs/core	≈ 3k cvs/core
Dec 2015	74	87
Feb 2016 (CPU only)	14.8	16
Mar 2016 (CPU/GPU)	8.3	9.7

normalized timing: core-sec/M cells/step at 2 levels of strong scaling

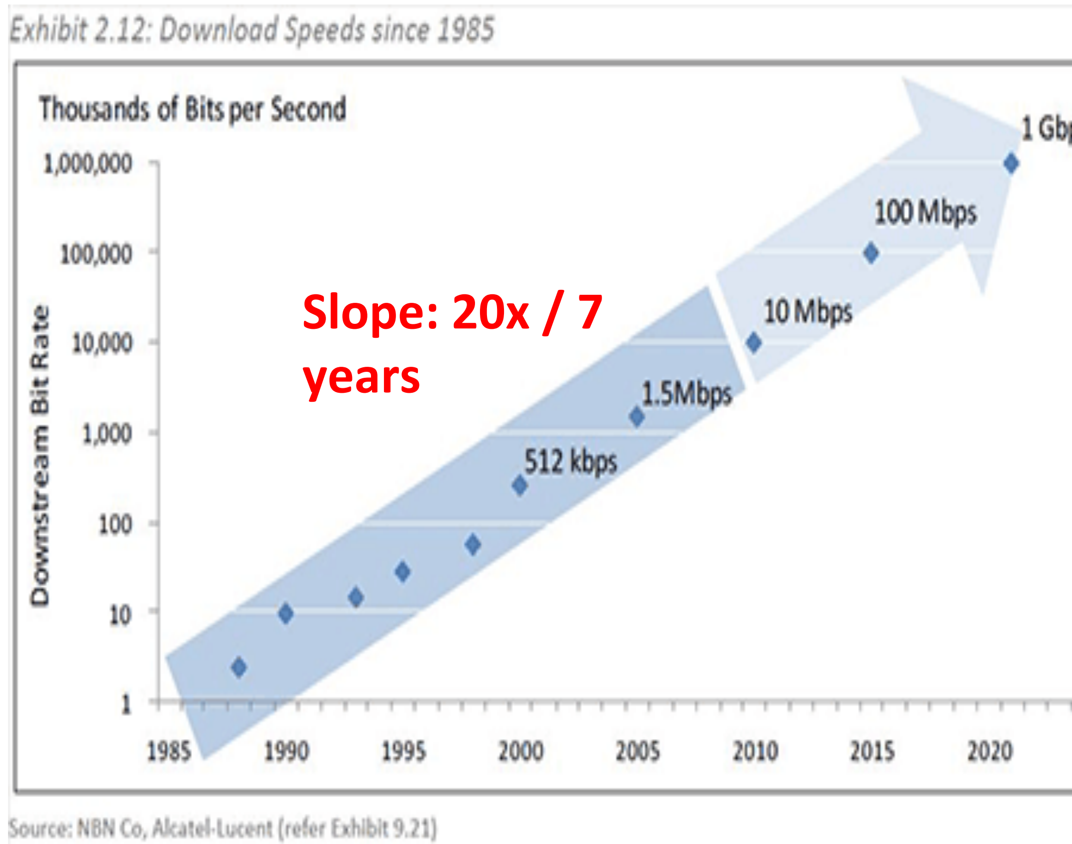
Approximately speedup of 9X for the CPU+GPU execution

Great: Simulations are running fast
Now how do we look at them/learn from them?

Ability to produce data exceeding our ability to move data by 5x/7years



History of supercomputer performance



History of network bandwidth

Solution: Images + metadata

- Instrument solvers to directly write images + metadata
- Images become Cartesian arrays of probes
- Dimensional reduction + lossless compression at source
- Intrinsically value

Simulation result file size: 20-200+ GB

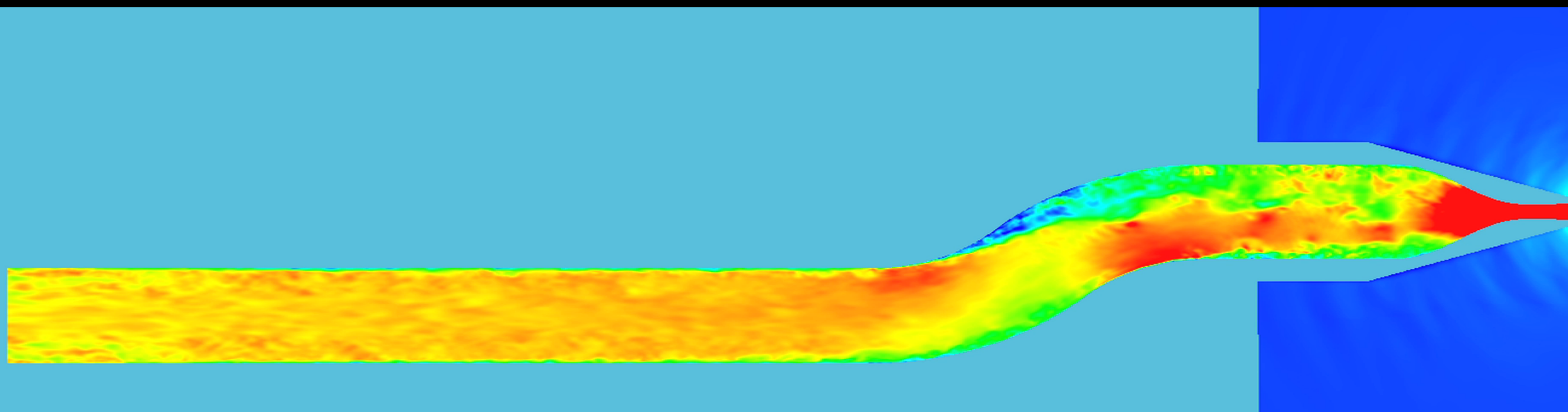
Image size: ~1 MB
(4-5 orders of magnitude smaller)

Leveraging the PNG standard

mages+metadata as Cartesian arrays of probes

Contours of streamwise velocity, u/c_0

Re_j = 418,000, St_{samp} = 2, *Data size = 50 MB*

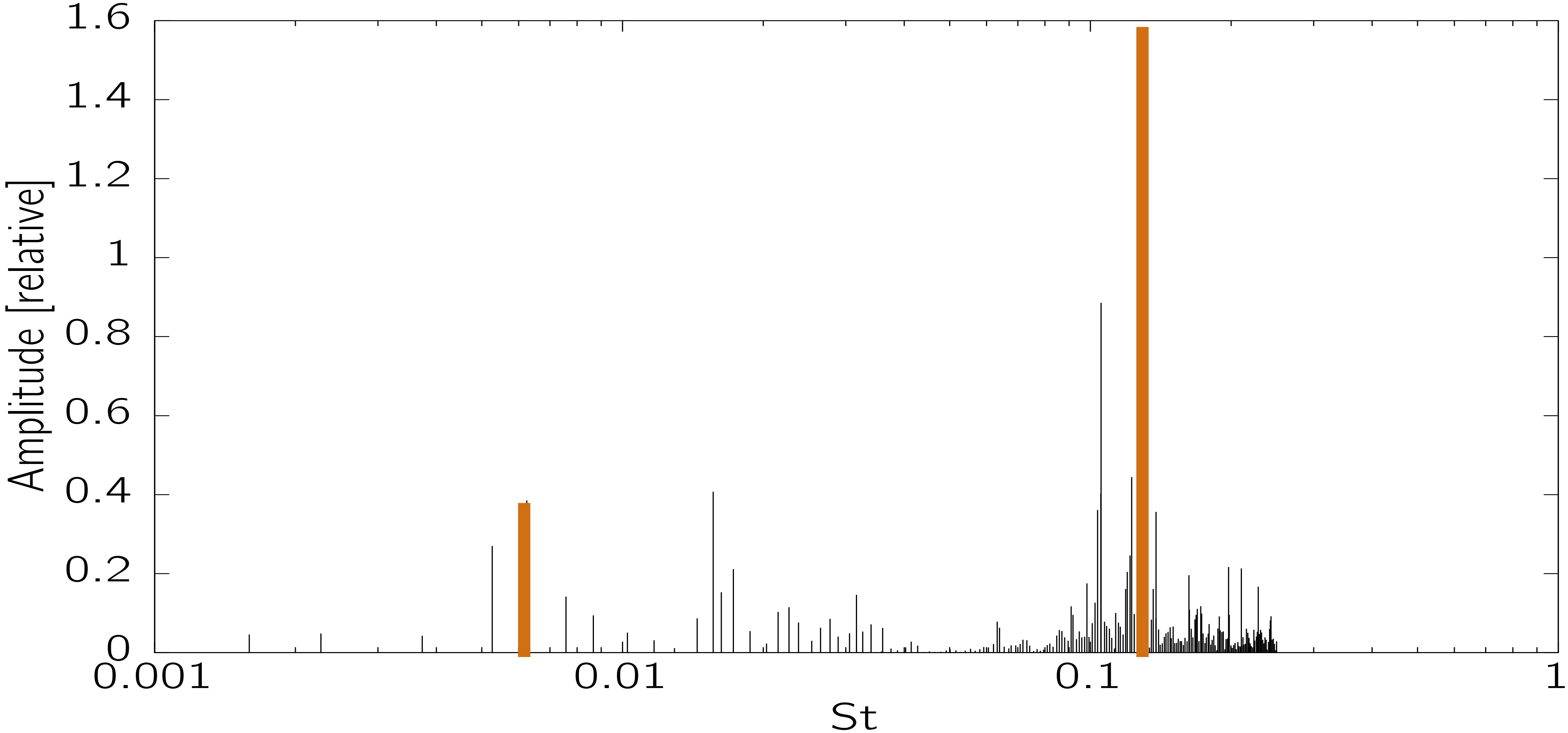


Treat each pixel in image as a data point (excluding geometry) analogous to PIV analysis

ed by NAVAIR STTR project with FSU, under the supervision of Dr.
pyropoulos. Computer time provided by DoD HPC facilities in ERDC DSRC

Quantitative data analysis from images

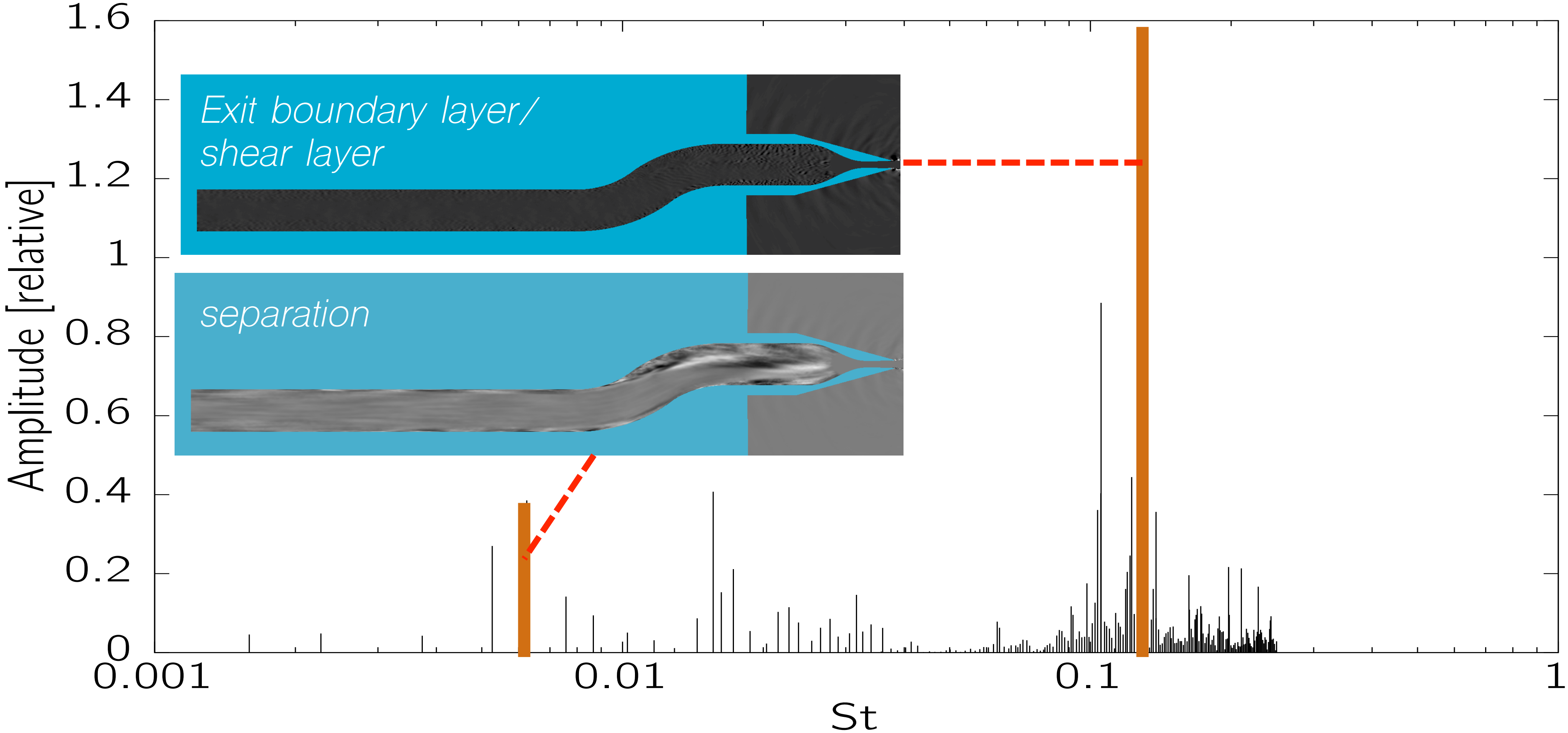
Dynamic mode decomposition



Identify salient flow structures/frequencies directly from images (similar to Schmid 2011, Schmid et al 2012 applied to expt data)

Quantitative data analysis from images

Dynamic mode decomposition



Identify salient flow structures/frequencies directly from images (similar to Schmid 2011, Schmid al 2012 applied to expt data)

Summary

- Fast development + tight coupling with internal design team AND hardware team allowed Titan-based simulations to contribute understanding and mitigate new technology risk for GE
- Solvers realized speedup of 9X relative to starting point over 2 quarter
- 3-way partnership with physics software developer in the loop critical for this success
- Speedup has allowed GE to bring many more impactful simulations in-house onto internal HPC