Penn State Aerospace Engineering

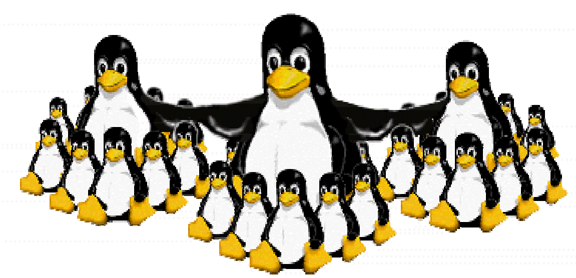
Linux Clusters: The HPC Revolution

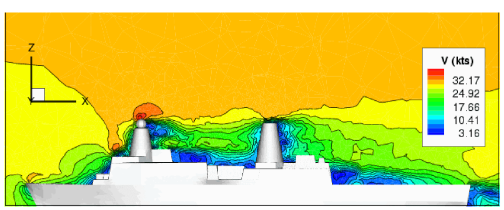
Turbulent Flow and Aeroacoustics Simulations using a Cluster of Workstations

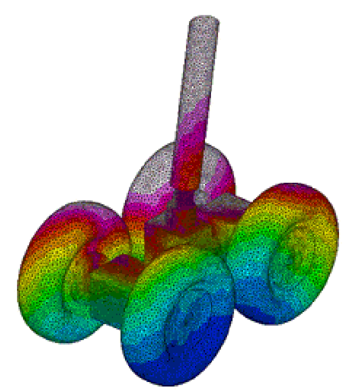
Anirudh Modi

and

Prof. Lyle N. Long



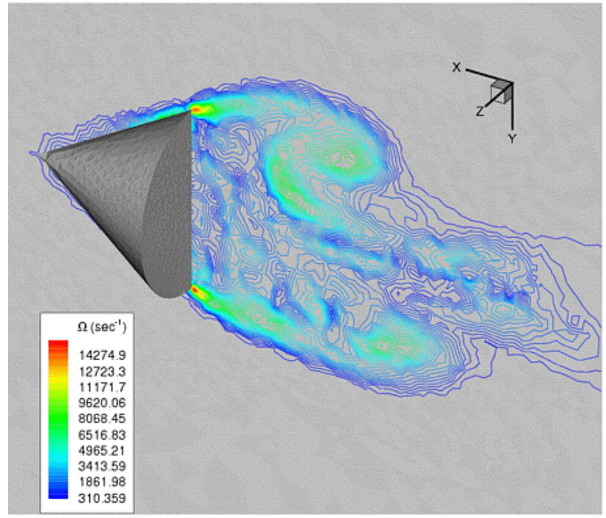


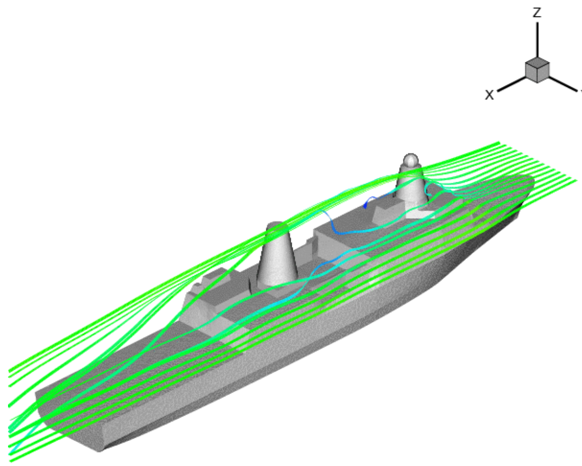


Aerospace Engineering

OUTLINE

- Introduction
- Parallel Flow Solver
- Grid Generation
- COCOA and COCOA-2
- Applications
  - •Aerodynamic Noise Prediction - Cone
  - •Aeroacoustics – Landing Gear
  - •Airwake Simulation – LPD17 ship
  - •Unsteady Flow – Helicopter Fuselage
- Concluding Remarks





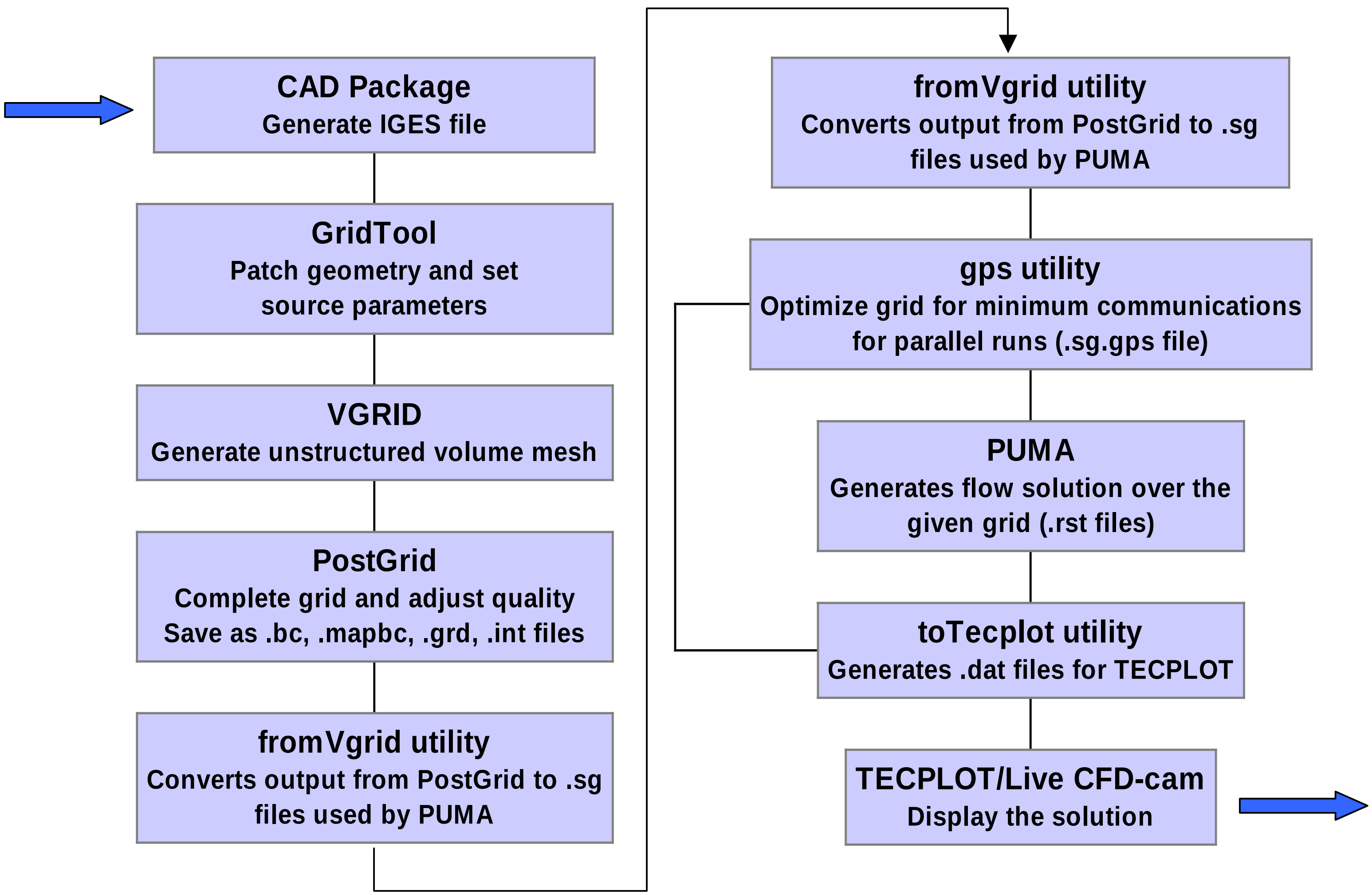
Introduction

Aerospace Engineering

- Numerical simulations are becoming increasingly important in all areas of science and engineering.
- Tremendous growth in the speed and memory of parallel computers makes it possible to run many of these simulations in reasonable time.
- One can now run very complex problems on inexpensive and indigenous cluster of workstations, which opens a whole new world of applications that were earlier seen to be beyond the realm of the Universities and small organisations.
- Aeroacoustic and turbulent flow simulations over complex geometries are examples of some complex problems that greatly benefit from this progress.

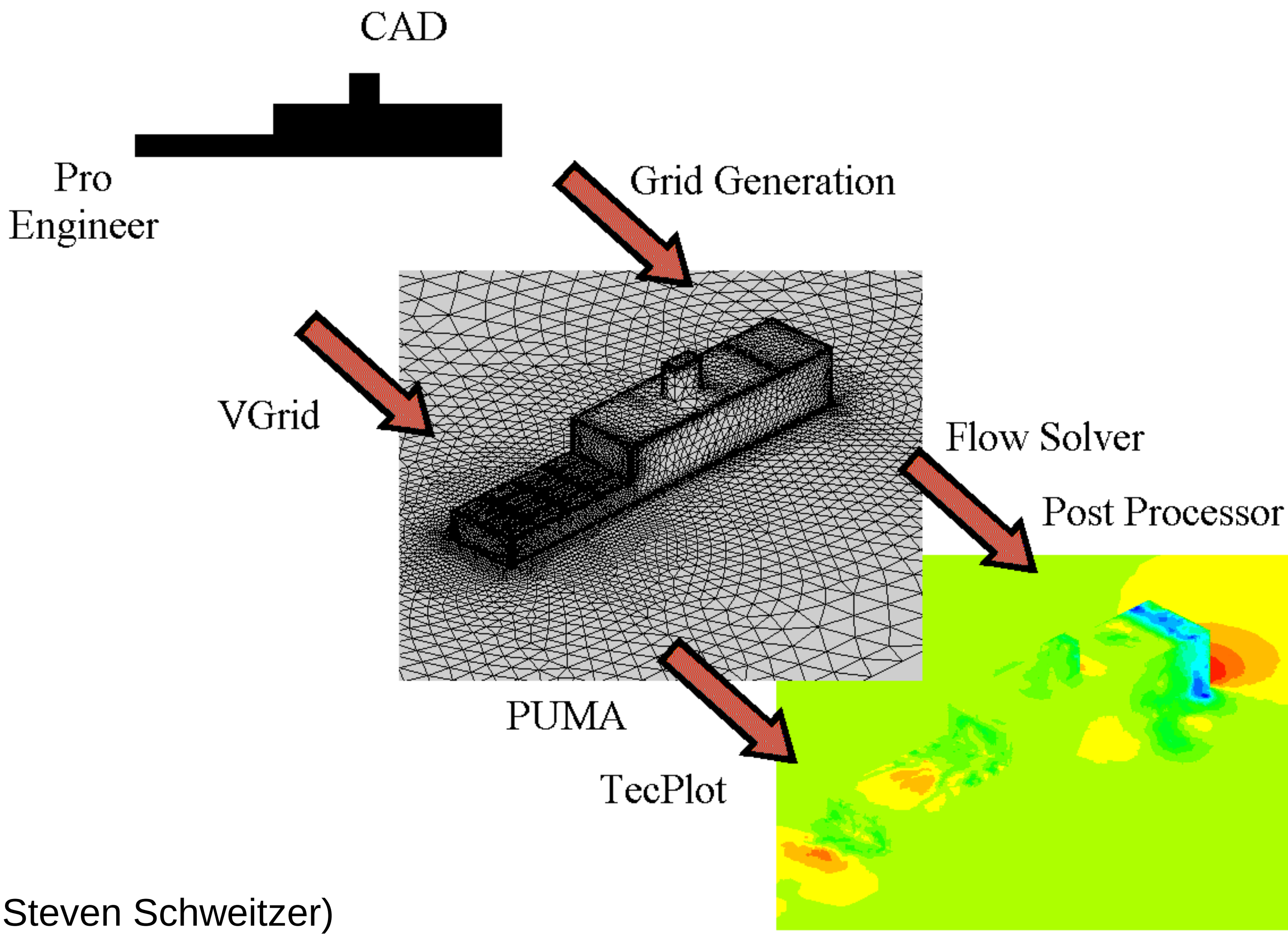
CAD to Solution

Aerospace Engineering



Example

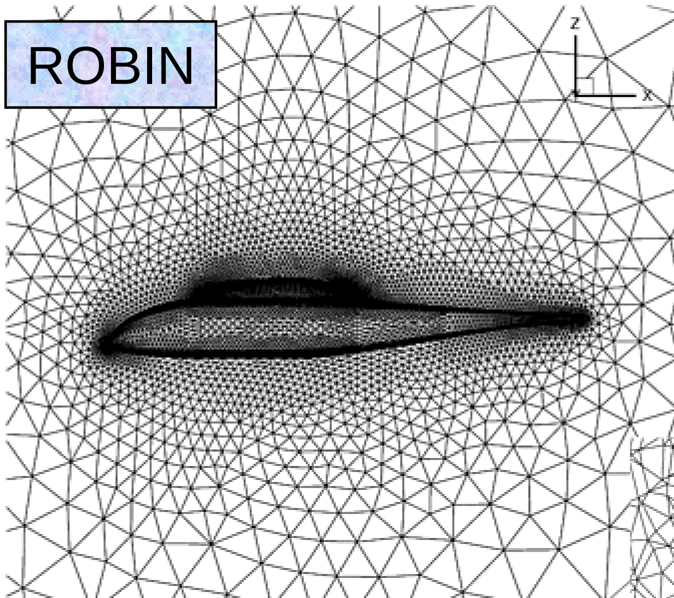
Aerospace Engineering



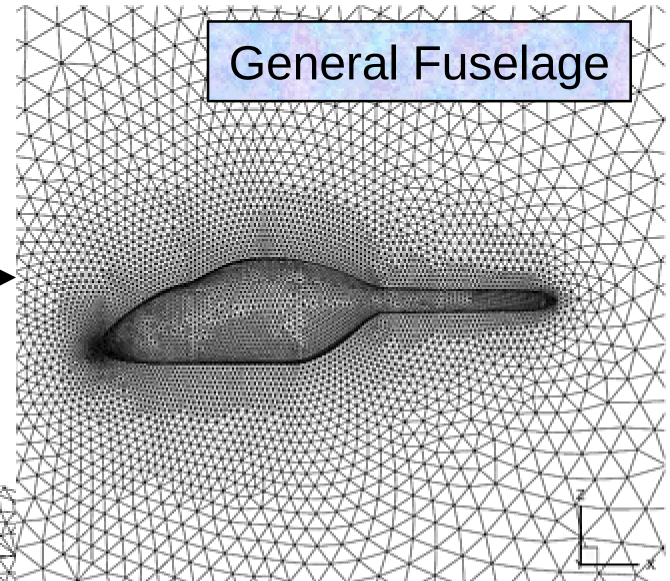
(Courtesy Steven Schweitzer)

Unstructured Grid Samples

Aerospace Engineering

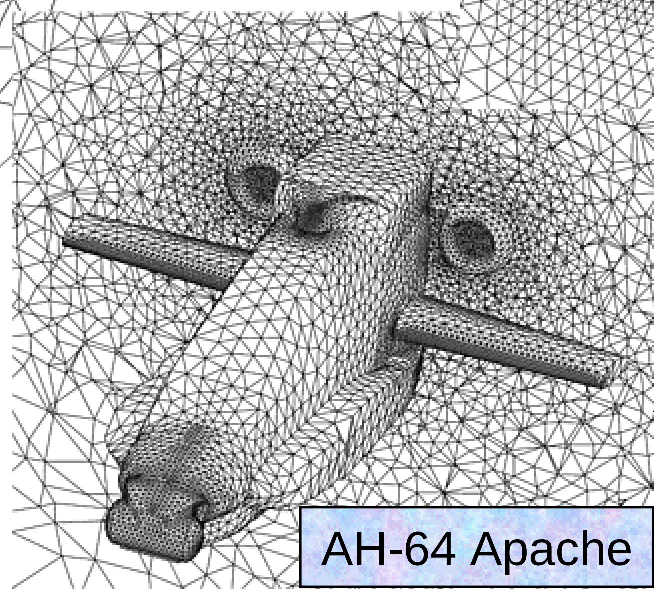


260,858 cells
532,492 faces



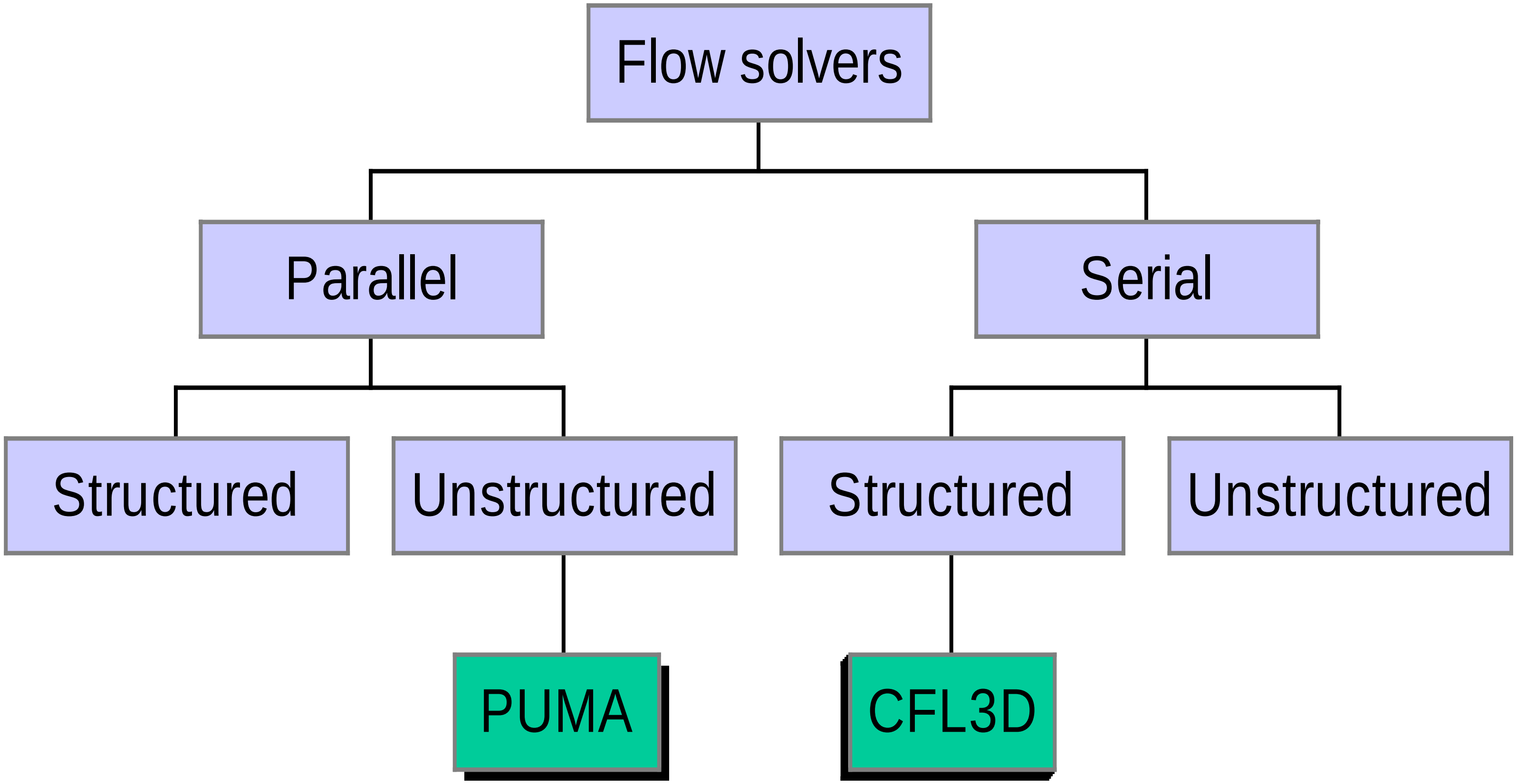
380,089 cells
769,240 faces

Helicopter Configurations



555,772 cells
1,125,596 faces

Flow Solvers



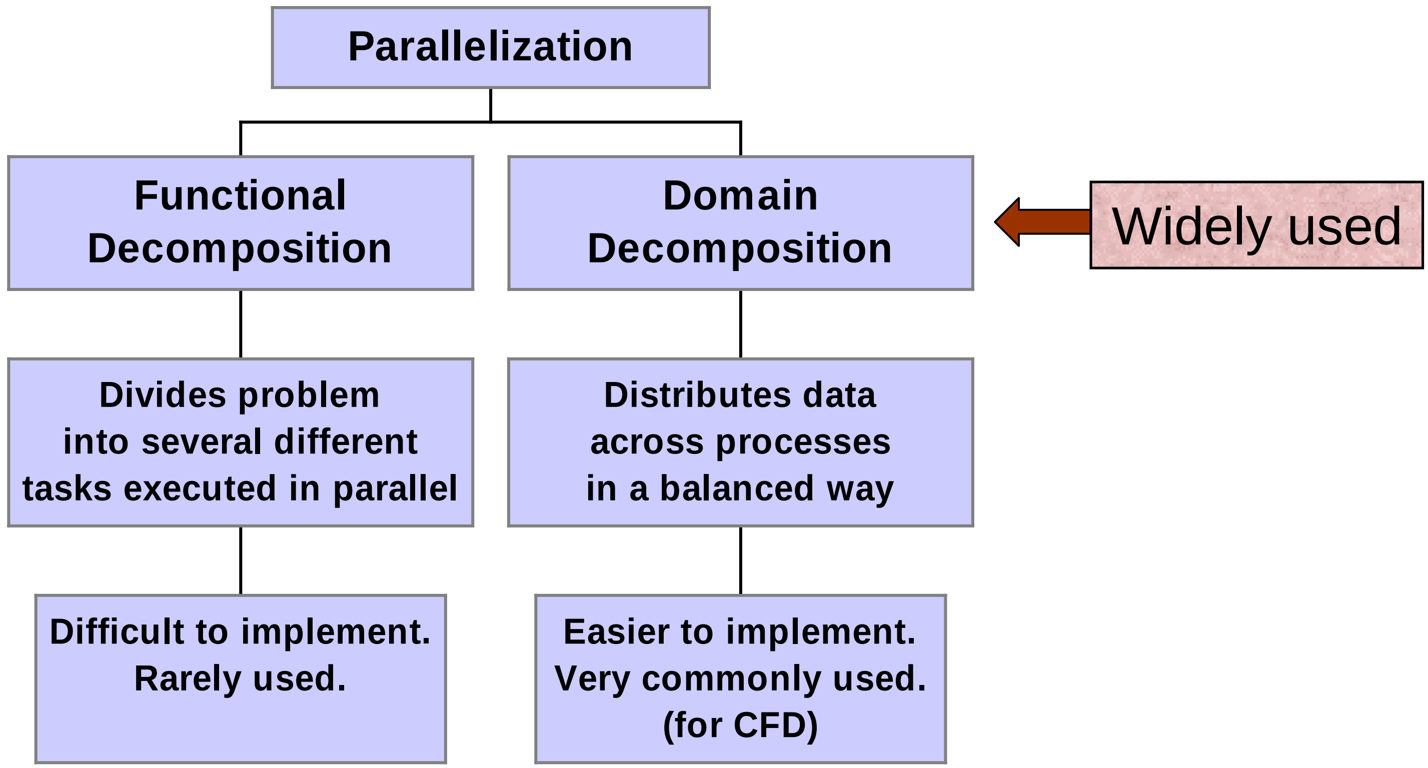
PUMA: Introduction

Penn State Aerospace Engineering

- Parallel Unstructured Maritime Aerodynamics. Written by Dr. Christopher W.S. Bruner (U.S. Navy, PAX River)
- Computer program for analysis of internal and external non-reacting compressible flows over arbitrarily complex 3D geometries (Navier-Stokes solver).
- Written entirely in ANSI C using MPI library for message passing and hence highly portable giving good performance.
- Based on Finite Volume method and supports mixed topology unstructured grids composed of tetrahedra, wedges, pyramids and hexahedra (bricks).

Parallelization strategies

Aerospace Engineering



Most massively parallel codes use *Single Program Multiple Data (SPMD)* parallelism, i.e., same code is replicated to each process.

Parallelization in PUMA

PennState
Aerospace Engineering

communication time = latency + (message size)/(bandwidth)

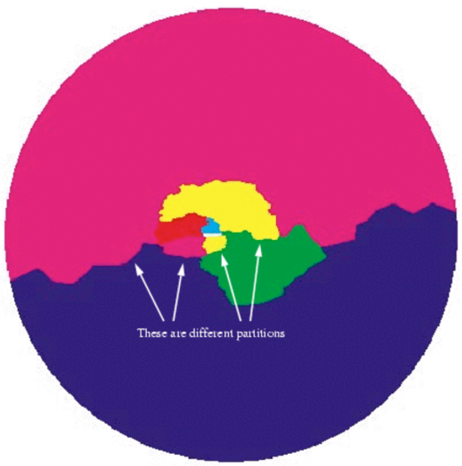
First term — Second term

Domain Decomposition

Approach #1 Minimization of inter-domain boundaries	Approach #2 Minimization of communications
Length of all shared data is minimized	Number of messages exchanged is minimized
Neglects latency. Ideal for slow networks.	Neglects second term. Ideal for fast networks.
Popular techniques require eigenvalues of connectivity matrix. Expensive.	Large knowledge base exists. Implementation is easy. Expensive.
Requires a-priori knowledge of number of domains/processors	Independent of number of domains/processors

PUMA

Grid around RAE 2822 a/f



8-way partitioning.
Using METIS s/w

8-way partitioning.
Using GPS reordering.

What is Beowulf?

Aerospace Engineering

- It is a system built using commodity hardware components, like any PC capable of running Linux, standard Ethernet adapters, and switches.
- It does not contain any custom hardware components and is trivially reproducible.
- Beowulf also uses commodity software like the Linux operating system, Parallel Virtual Machine (PVM) and Message Passing Interface (MPI), and other widely available open-source software.
- A Beowulf system behaves more like a single machine rather than many workstations as the server node controls the client nodes transparently.

COCOA

Aerospace Engineering

COst effective COmputing Array (COCOA)

25 Dual PII 400 MHz
512 MB RAM each (12+ GB!!)
100 GB Ultra2W-SCSI Disk on server
100 Mb/s Fast Ethernet cards
Baynetworks 450T 27-way switch
(backplane bandwidth of 2.5 Gbps)
Monitor/keyboard switches
RedHat Linux with MPI

http://cocoa.ihpca.psu.edu

Cost just $100,000!! (1998 dollars)

COCOA-2

Penn State
Aerospace Engineering

COst effective COmputing Array-2 **(COCOA-2)**
21 Dual PIII 800 MHz (1U each)
1 GB PC133 ECC each (21 GB!!)
100 GB Ultra160-SCSI Disk on server
Dual 100 Mb/s Fast Ethernet cards
Two 1U HP-Procurve 2324 24-way switches (b/p bandwidth of 9.6 Gbps)
Monitor/keyboard switches
RedHat Linux 7.0 with MPI
Two 4U 3000 VA APC SmartUPS

http://cocoa2.ihpca.psu.edu

Cost just $58,000!! (2001 dollars)

COCOA: Motivation

Aerospace Engineering

- To get even 50,000 hrs of CPU time in a supercomputing center is difficult. COCOA can offer more that 400,000 CPU hrs annually!
- One often has to wait for days in queues before the job can run.
- Commodity PCs are getting extremely cheap. Today, it just costs $2K to get a dual PIII-800 computer with 512 MB RAM from a reliable vendor like Dell!
- Advent of Fast Ethernet (100 Mbps) networking has made a reasonably large PC cluster feasible (at a very low cost; 100 Mbps ethernet adaptor ~ $70). Myrinet and Gigabit networking are popular.
- Price/performace (or $/Mflop) for these cheap clusters is way better than for a IBM SP/SGI/Cray supercomputer (usually factor of 10 better!)
- Maintenance for such a PC cluster is less cumbersome than the big computers. A new node can be added to COCOA in just 10 minutes!

COCOA: Software

Aerospace Engineering

- COCOA runs on commodity PCs using commodity software (RedHat Linux).
- Cost of software: negligible. The only commercial software installed are Absoft and Portland Group Fortran 90 compiler and TECPLOT.
- Free version of MPI from ANL (MPICH) and Pentium GNU C compiler (generates highly optimized code for Pentium class chips) are installed.
- Distributed Queueing System (DQS) has been setup to submit the parallel/serial jobs. Several minor enhancements have been incorporated to make it extremely easy to use. Live status of the jobs and the nodes is available on the web:

http://cocoa.ihpca.psu.edu

- Details on how COCOA was built can be found in the COCOA HOWTO:

http://bart.ihpca.psu.edu/cocoa/HOWTO/

COCOA: Advantages

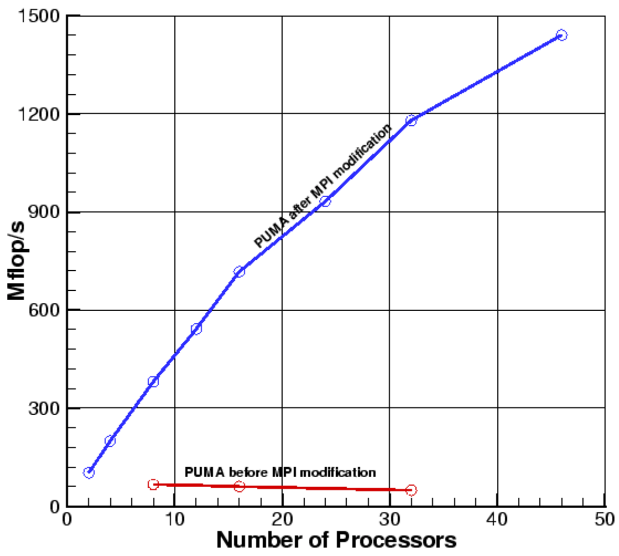
Aerospace Engineering

- Easy to administer since the whole system behaves like one single machine.
  - All modifications to the server/clients are done remotely.
- Due to the kickstart installation process, adding a new node is a very simple process which barely takes 10 minutes!!
- Due to the uniformity in the software installed on all the clients, upgrading them is very easy.
- Submitting and deleting jobs is very easy.
  - A regular queue is present for regular jobs which take hours or days to complete.
  - An 8-proc debug queue is present for small jobs.

COCOA: Modifications to PUMA

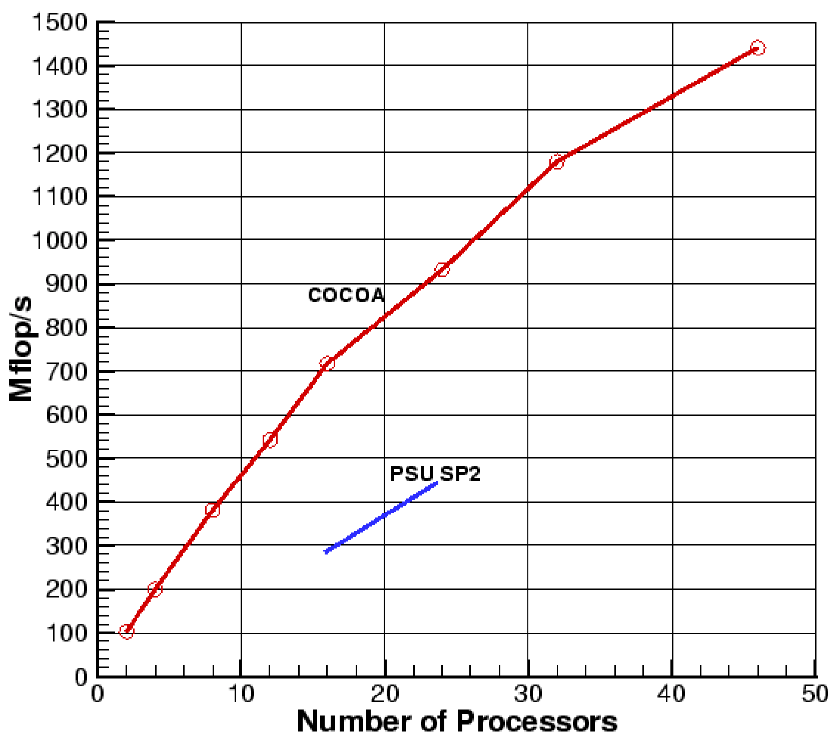
Aerospace Engineering

- ➤ Although PUMA is portable, it was aimed at very low-latency supercomputers. Running it on a high-latency cluster like COCOA posed several problems.
- ➤ PUMA often used several thousand very small messages (< 100 bytes) for communication which degraded its performance considerably (latency!!). These messages were non-trivially packed into larger messages (typically > 10 Kbyes) before they were exchanged.
- ➤ After modification, the initialization time was reduced by a factor of 5-10, and the overall performace was improved by a factor of 10-50!!

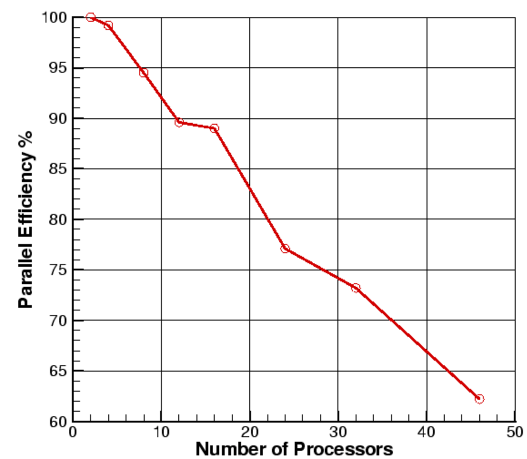


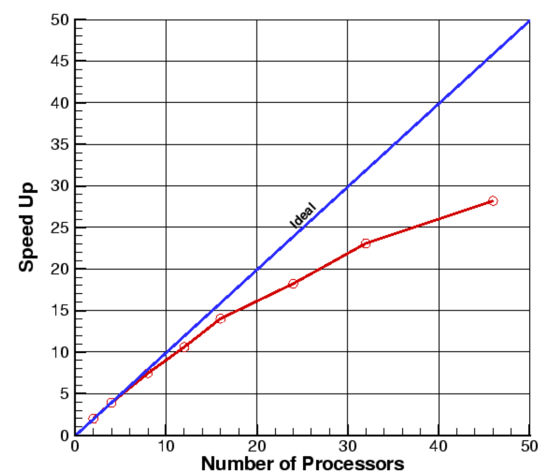
COCOA: Benchmarks

PENNSTATE
Aerospace Engineering



Performance of “*Modified PUMA*”

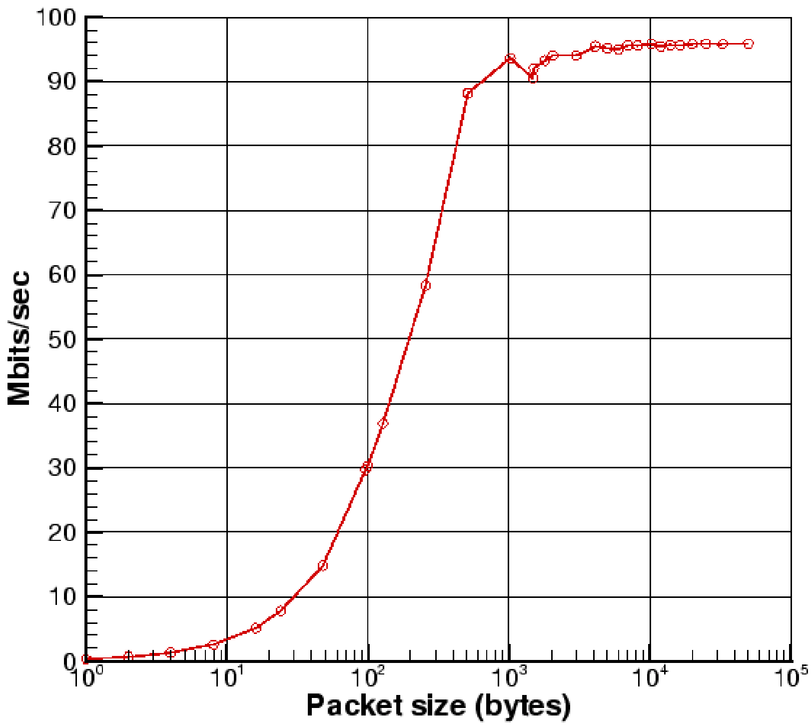




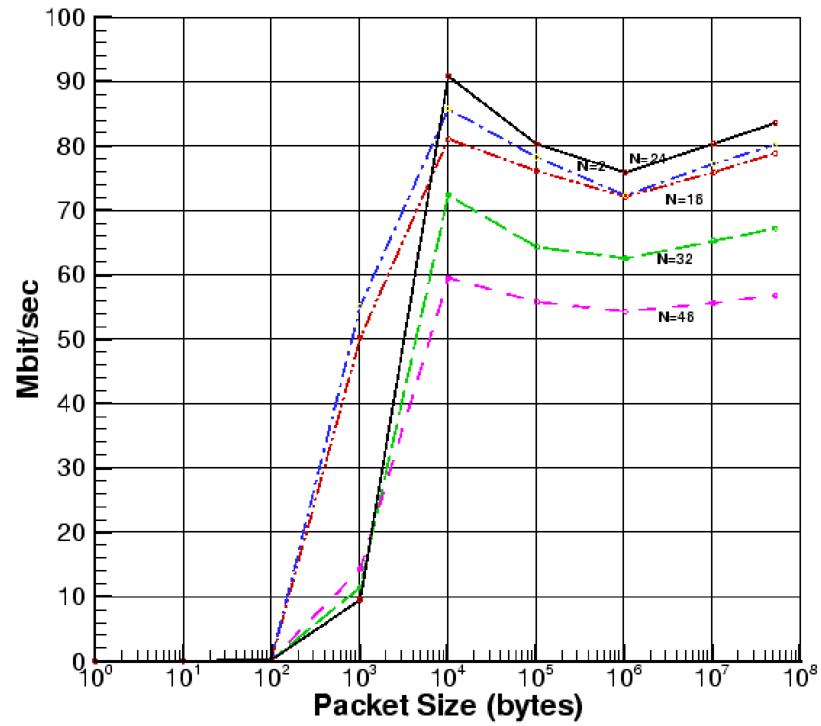
COCOA: Benchmarks

Aerospace Engineering

Network Performance



netperf test between any two nodes



MPI_Send/Recv test

➡ Ideal message size >= 10 Kbytes

COCOA: Benchmarks

Aerospace Engineering

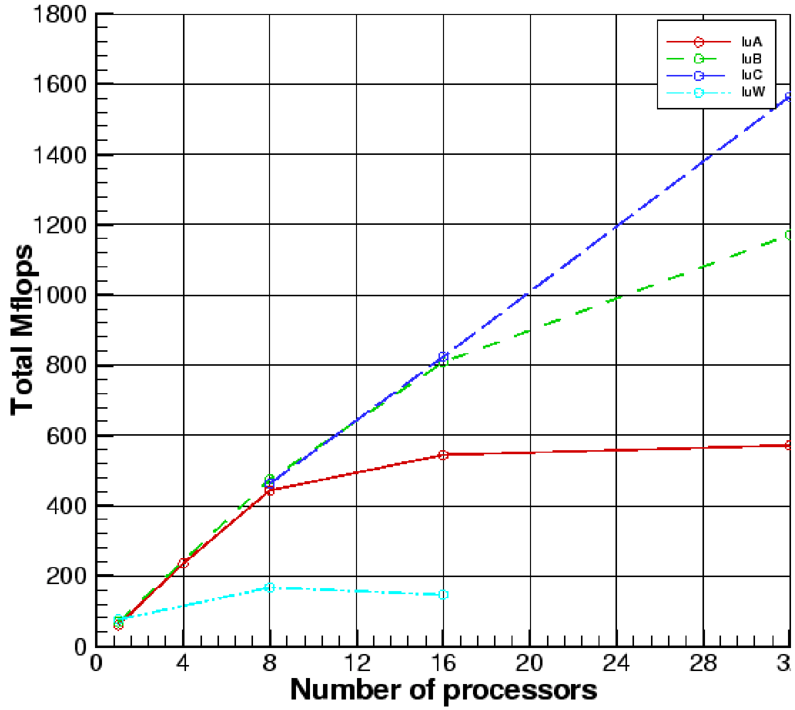
NAS Parallel Benchmarks (NPB v2.3):

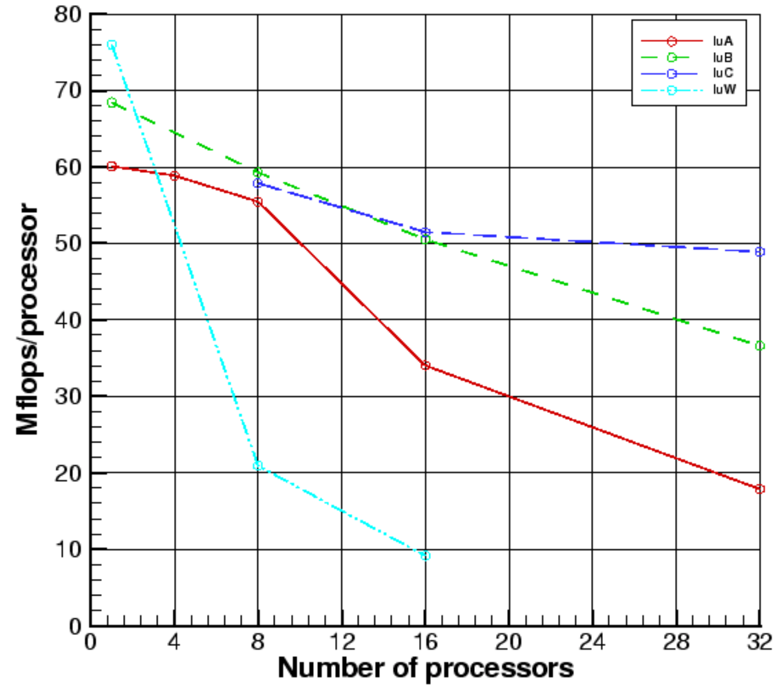
- Standard benchmark for CFD applications on large computers.
- 4 sizes for each benchmark: Classes W, A, B and C.
  - » Class W: Workstation class (small in size)
  - » Class A, B, C: Supercomputer class (C being largest)

Benchmark Code	Class A	Class B	Class C
Embarrassingly parallel (EP)	2^{28}	2^{30}	2^{32}
Multigrid (MG)	256^3	256^3	512^3
Conjugate gradient (CG)	1.4×10^4	7.5×10^4	1.5×10^5
3D FFT PDE (FT)	$256^2 \times 128$	512×256^2	512^3
Integer sort (IS)	2^{23}	2^{25}	2^{27}
LU solver (LU)	64^3	102^3	162^3
Pentadiagonal solver (SP)	64^3	102^3	162^3
Block tridiagonal solver (BT)	64^3	102^3	162^3

COCOA: Benchmarks

Aerospace Engineering

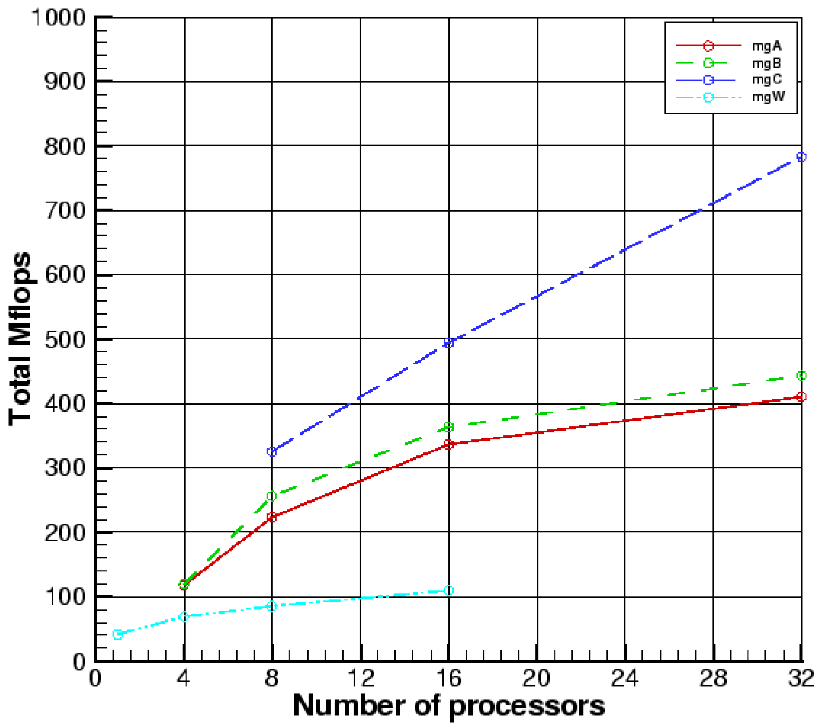


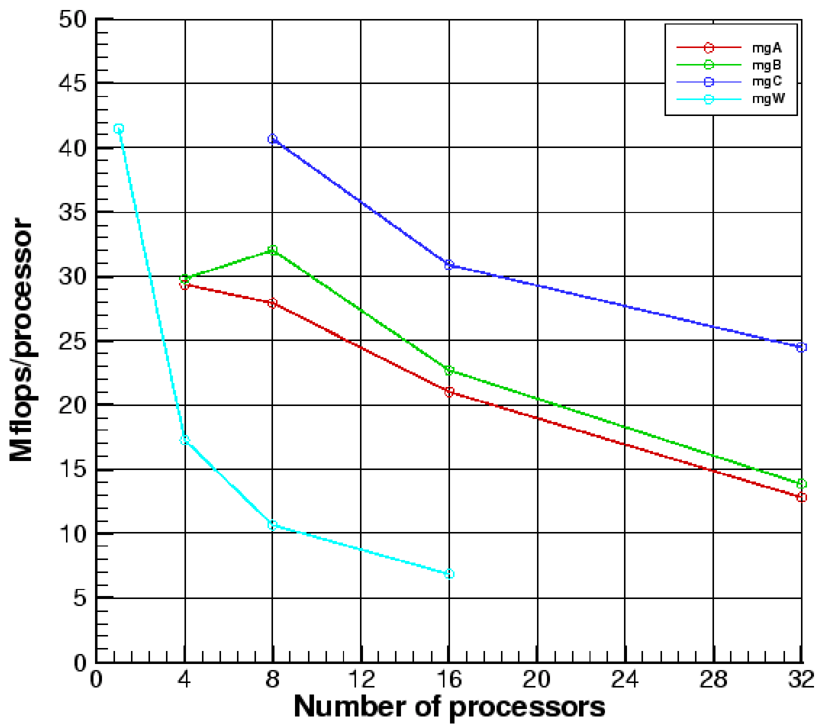


NAS Parallel Benchmark on COCOA*: LU solver* (LU) test

COCOA: Benchmarks

Aerospace Engineering

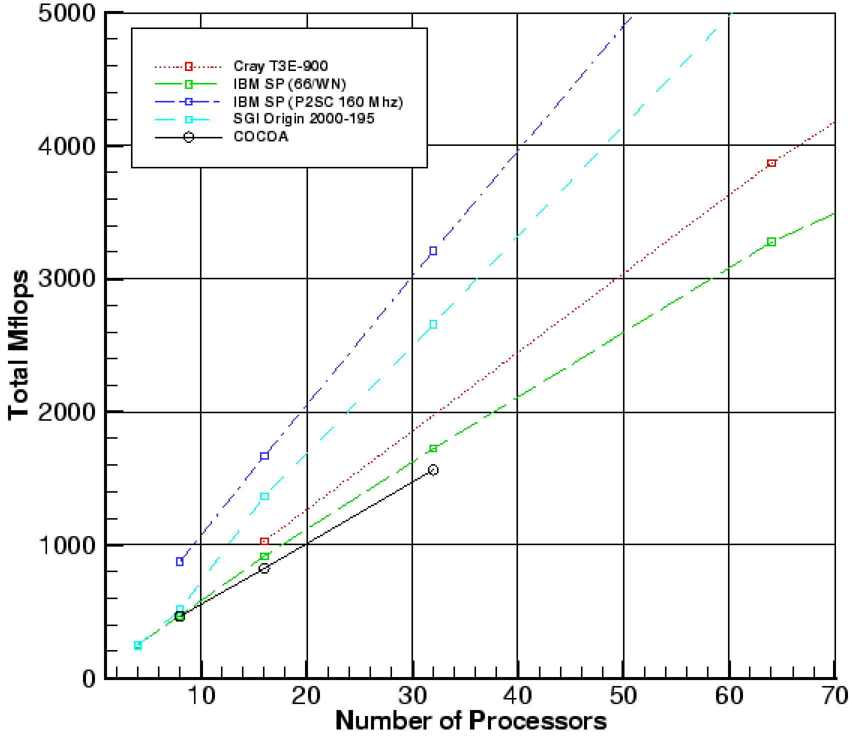


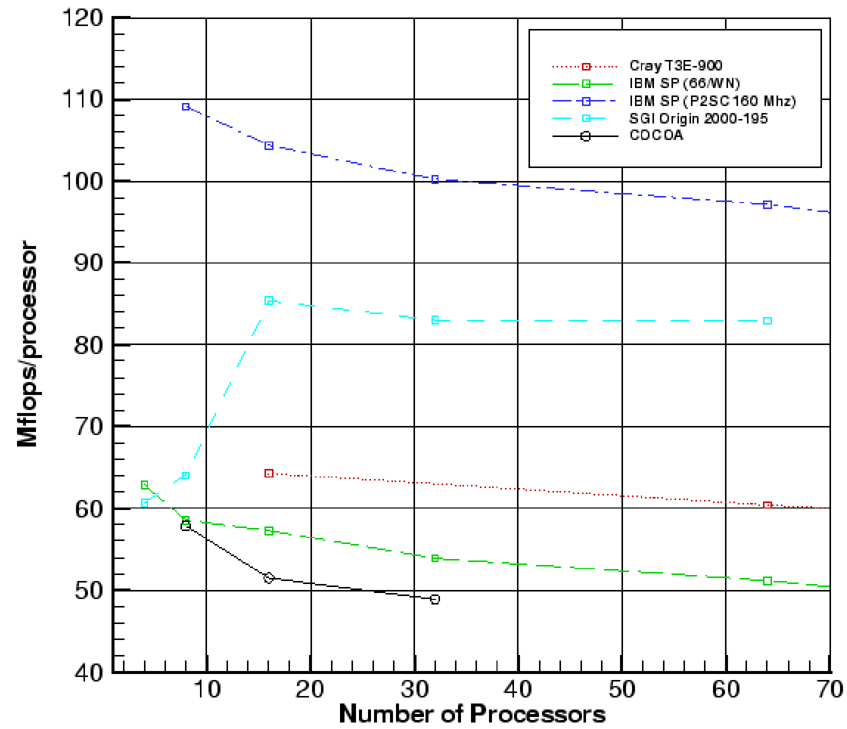


NAS Parallel Benchmark on COCOA*: Multigrid* (MG) test

COCOA: Benchmarks

Aerospace Engineering

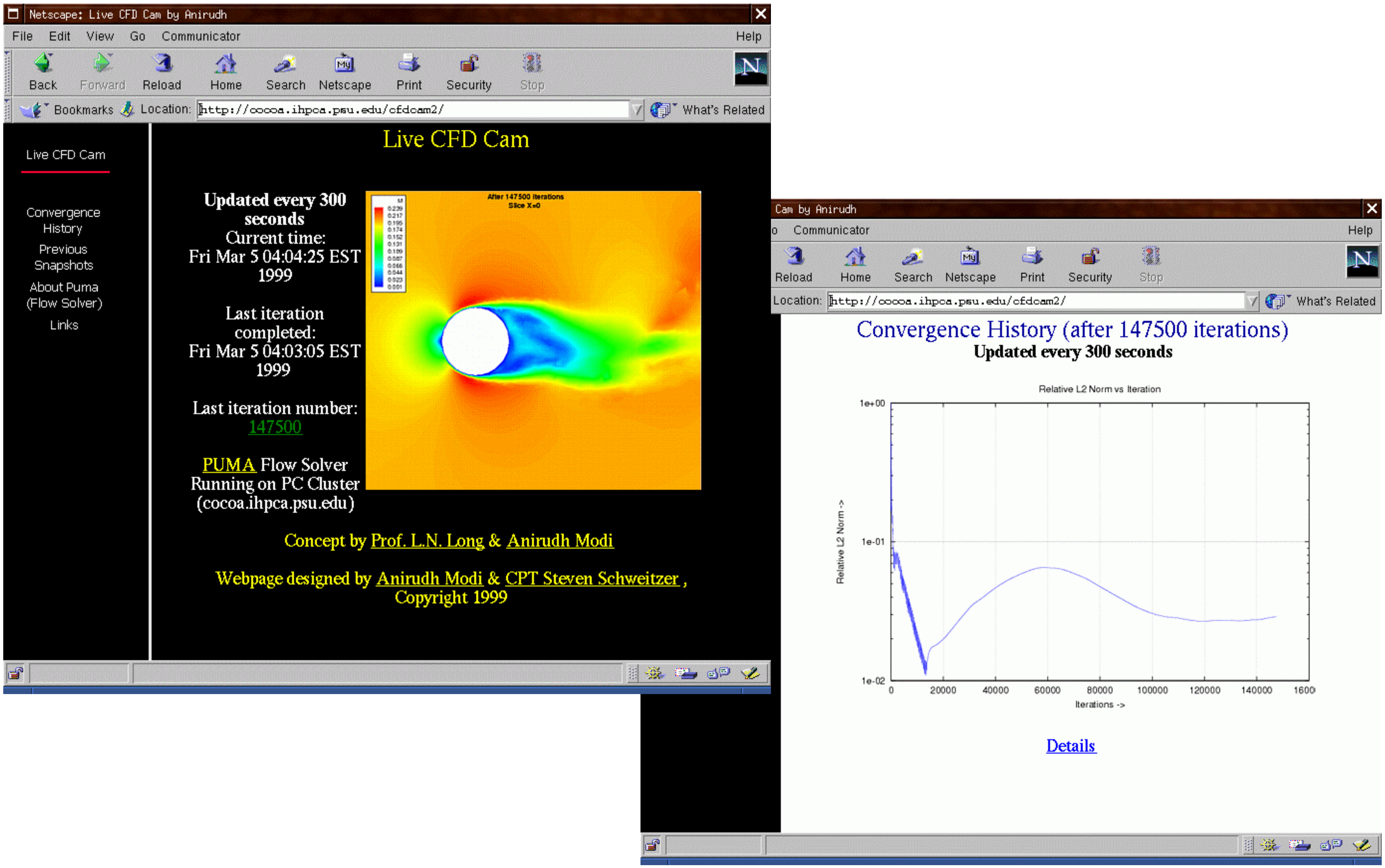




LU solver (LU) test: Comparison with other machines

Live CFD-Cam

Aerospace Engineering



Applications

Aerodynamic Noise

Penn State Aerospace Engineering

Comprises of :

» **Airframe Noise**

¤ landing gear, trailing edge, fuselage, high-lift devices etc.

» **Propulsive Noise**

¤ jet noise (from the engine)

¤ fan noise

Becoming increasingly important because of strict FAAR regulations.

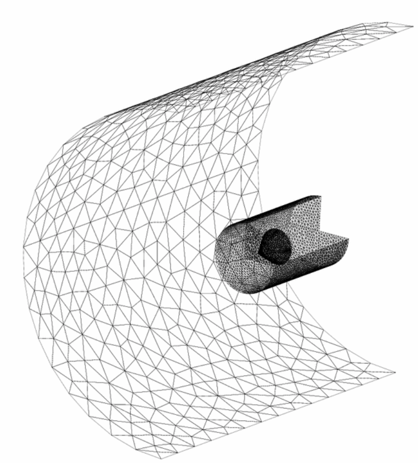
(Courtesy: Anupam Sharma, Fred Souliez, Prof. Lyle N. Long)

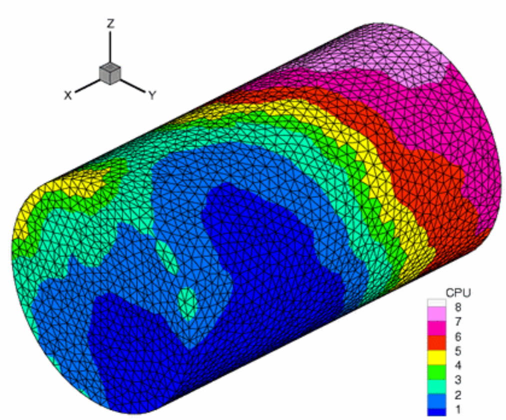
Test Case: Cone Flow

Aerospace Engineering

- Unstructured mesh around cone generated using Gridgen (280,000 tetrahedra).
- 8,500 face data on the FW-H surface.
- Computation of one shedding cycle based on Strouhal number of 0.171.

Location of FW-H surface between cone and domain limits





CPU partition on the FW-H surface mesh

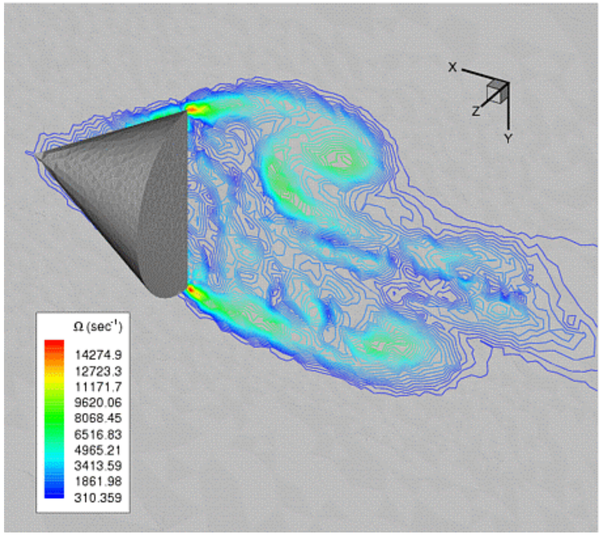
(Courtesy: Fred Souliez, Prof. Lyle N. Long)

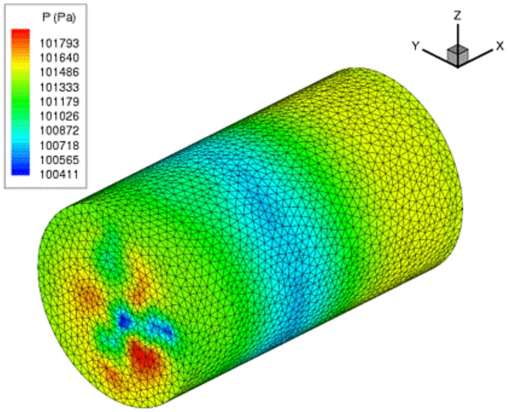
Results: Cone Flow

Aerospace Engineering

- Two shedding cycles covered / week
- Re = 50,000, M=0.2
- Pressure, density and velocity collected on FW-H surface

Instantaneous vorticity contours



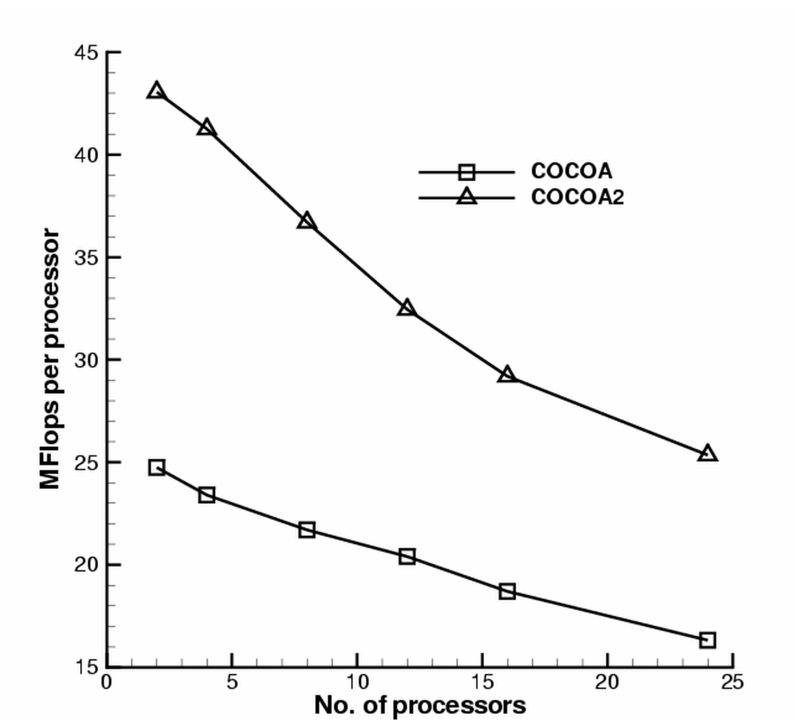


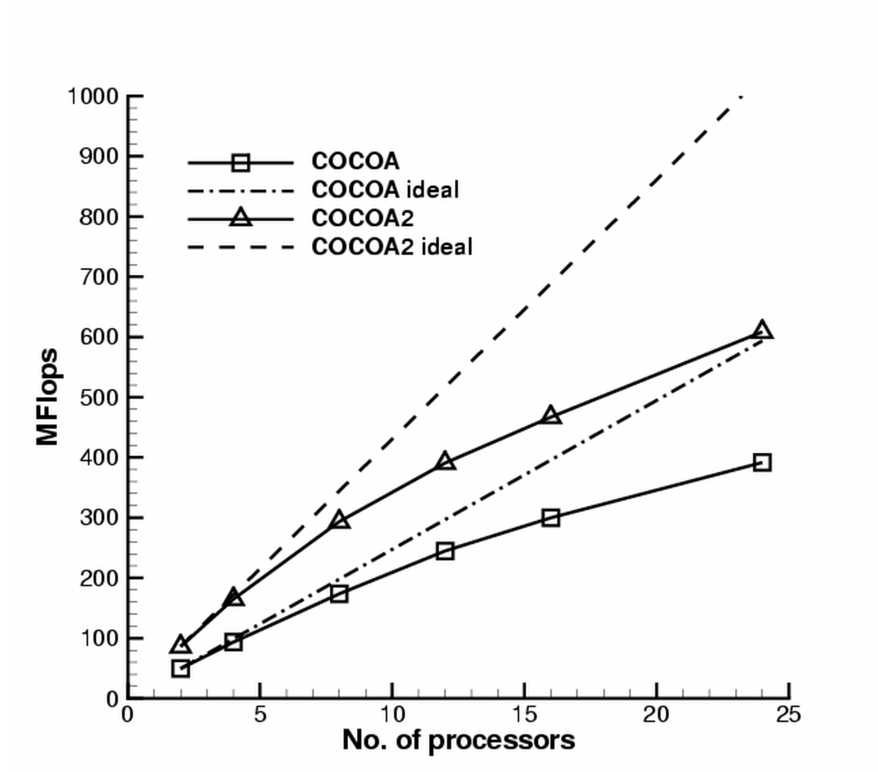
Instantaneous pressure distribution on FW-H surface

(Courtesy: Fred Souliez, Prof. Lyle N. Long)

Performance for Cone Case

Aerospace Engineering





Performance Comparison of COCOA and COCOA-2 for the aeroacoustics flow simulation over the cone (280,000 cells).

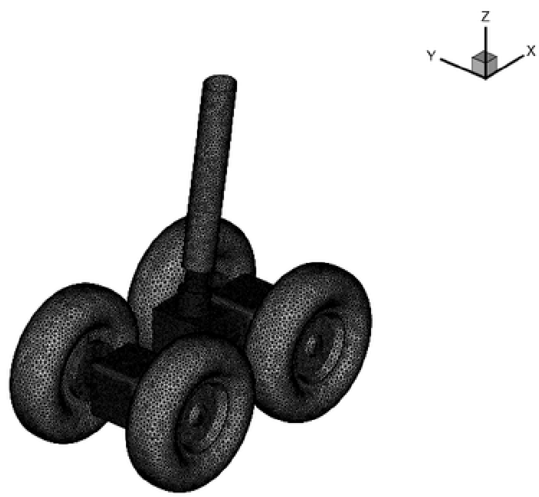
(Courtesy: Fred Souliez, Prof. Lyle N. Long)

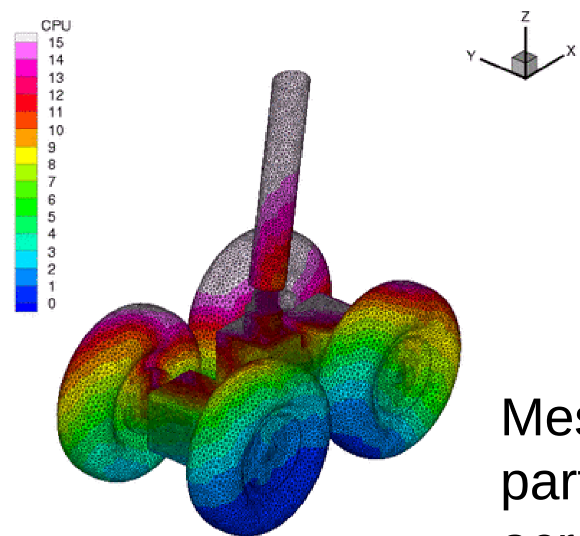
Landing Gear Aeroacoustics

Aerospace Engineering

- Landing gear mesh generated using Gridgen
  - 100,000 faces on landing gear
  - 2,000,000 tetrahedra in volume mesh (approx 5 GB memory for R-K, 10 GB for SSOR)

Surface mesh on landing gear



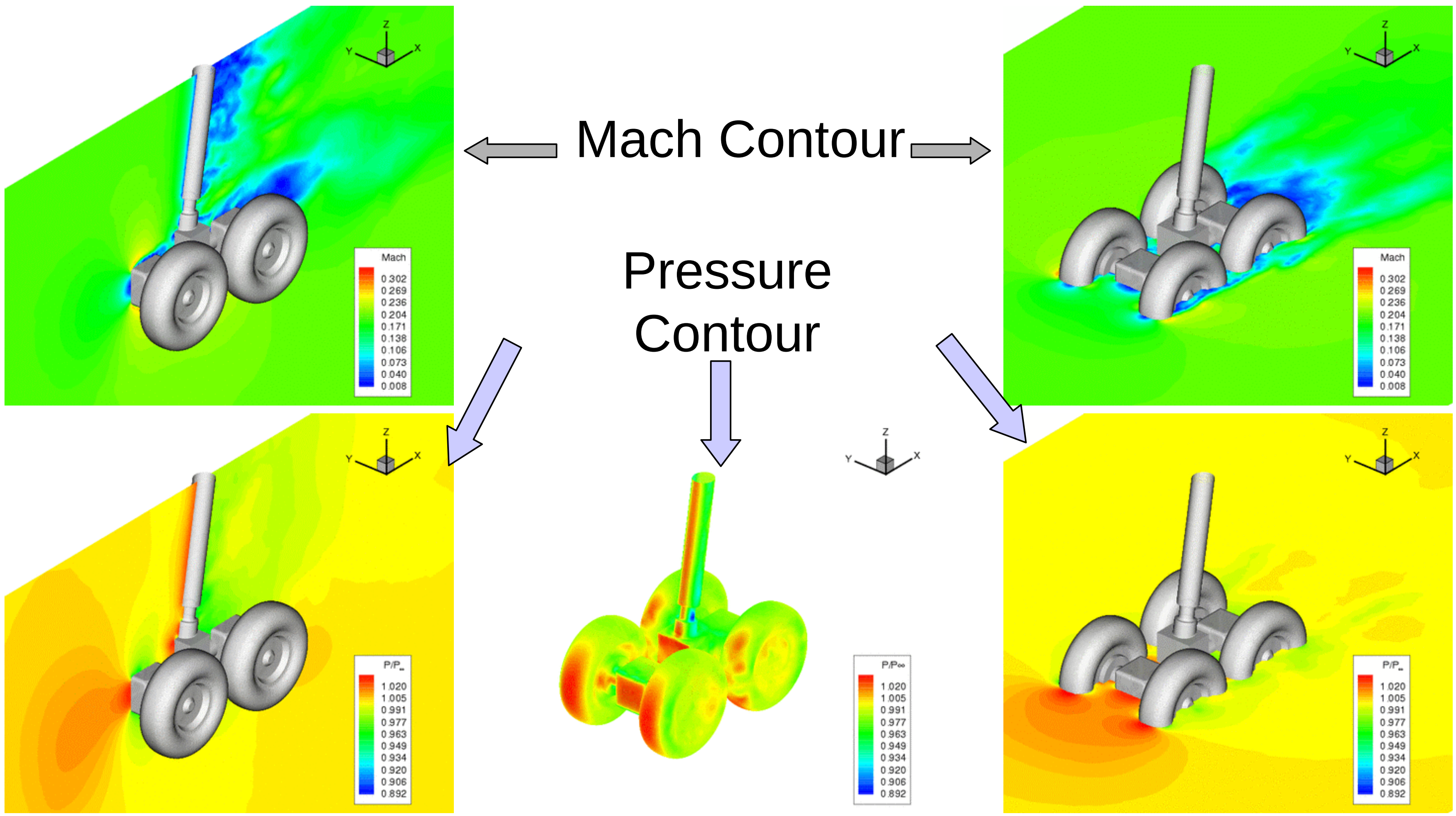


Mesh partition across 16 CPUs

(Courtesy: Fred Souliez, Prof. Lyle N. Long)

Landing Gear Aeroacoustics

- Preliminary steady-state solution completed

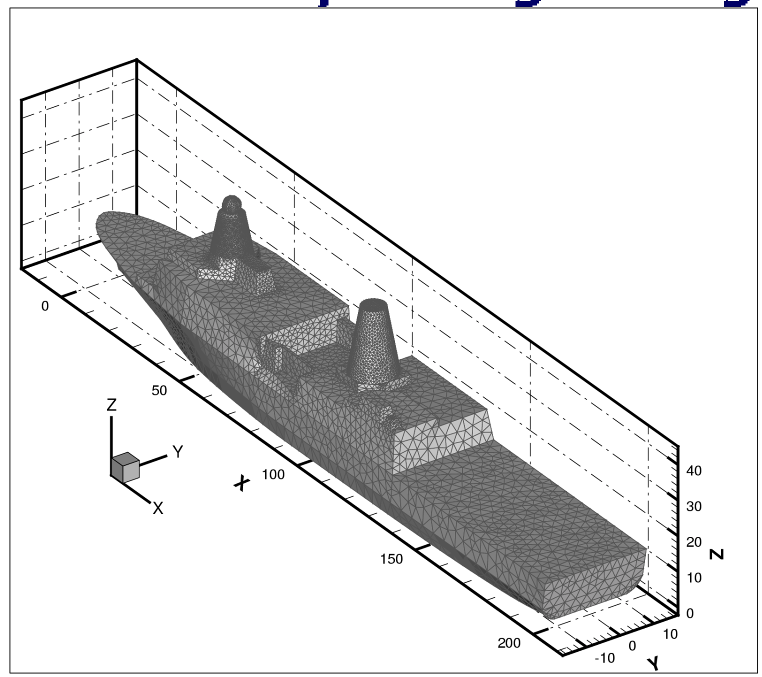


Airwake: LPD17 Grid

Aerospace Engineering

» Complex geometry.

» Unstructured grid using Gridgen.
- ¤ 0.2 million cells (tetrahedra).
- ¤ full-scale geometry.
- ¤ 200,000 iterations
- ¤ 20 days on 8-proc COCOA
- ¤ 50 Mflops/proc
- ¤ 500 MB for R-K
- ¤ A case with 1.2M cells also attempted (3.2 GB memory)



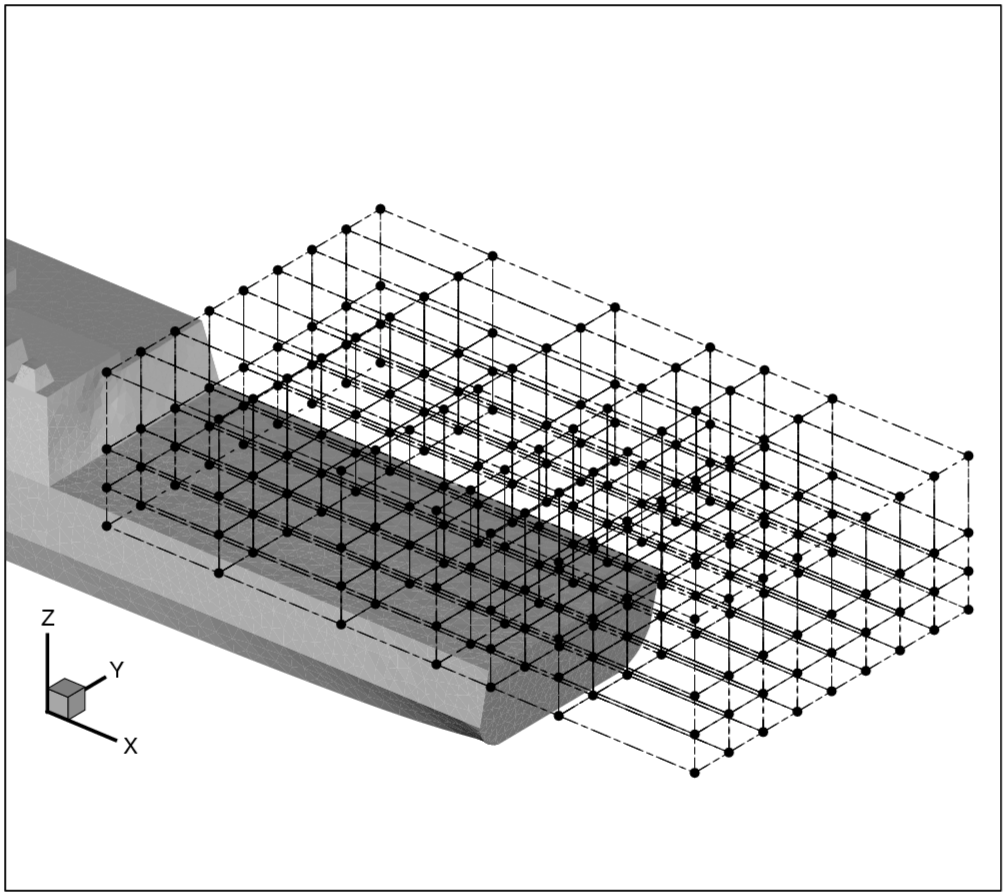
» Characteristics :
- ¤ Superstructure and sharp edges cause flow separation.
- ¤ Flow is inherently unsteady.
- ¤ Ship is asymmetric.

(Courtesy: Anupam Sharma, Prof. Lyle N. Long)

Airwake: LPD17 Results

Aerospace Engineering

Points where experimental data was collected for the 0º yaw case.



(Courtesy: Anupam Sharma, Prof. Lyle N. Long)

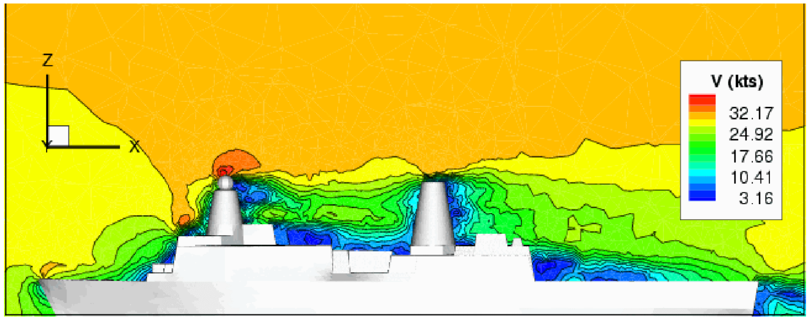
Results: Pseudo SS

Aerospace Engineering

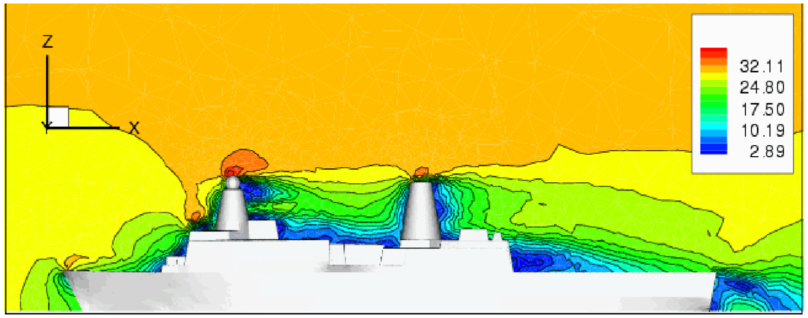
Simulated time averaged solution

» Time accurate data sampled for a unit non-dimensional time.
» 50 samples are collected for time averaging.
» Compared with the pseudo SS solution.

Velocity contours for 0º yaw case



Pseudo-SS



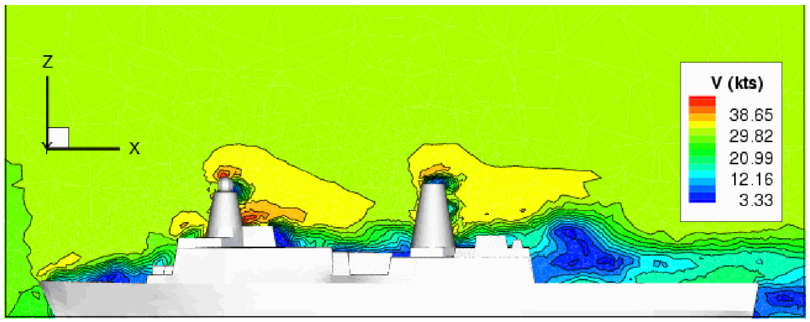
Time-averaged

(Courtesy: Anupam Sharma, Prof. Lyle N. Long)

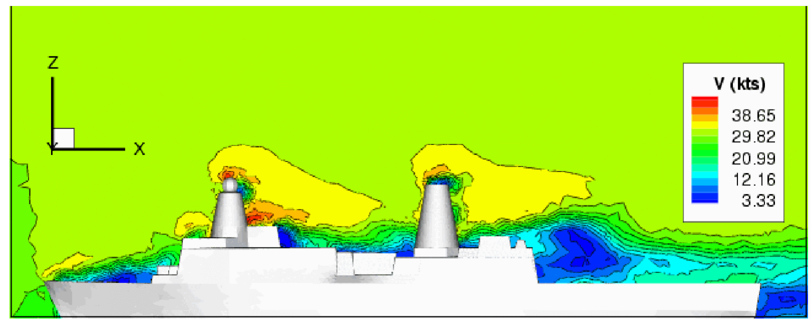
Airwake: LPD17 Results

Aerospace Engineering

Velocity contours for 30º yaw case



Pseudo-SS



Time-averaged

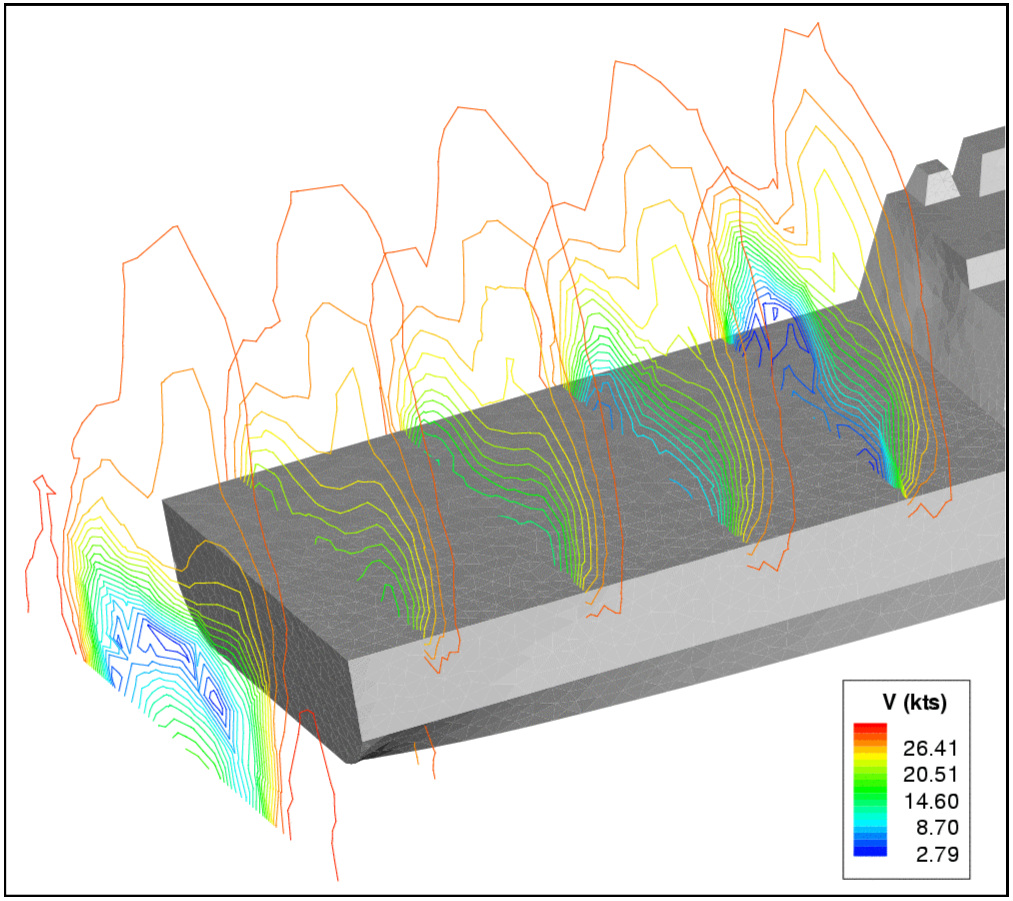
» Pseudo-SS solution compares very well with time-averaged solution.

(Courtesy: Anupam Sharma, Prof. Lyle N. Long)

Airwake: LPD17 Results

Aerospace Engineering

Velocity contours at different ‘x’ locations for the 0º yaw case



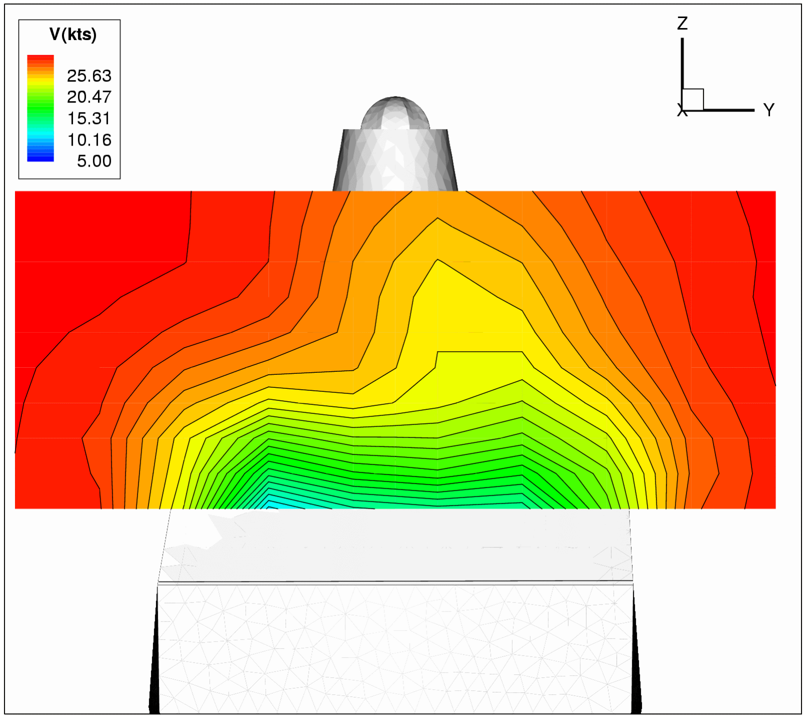
Pseudo SS

(Courtesy: Anupam Sharma, Prof. Lyle N. Long)

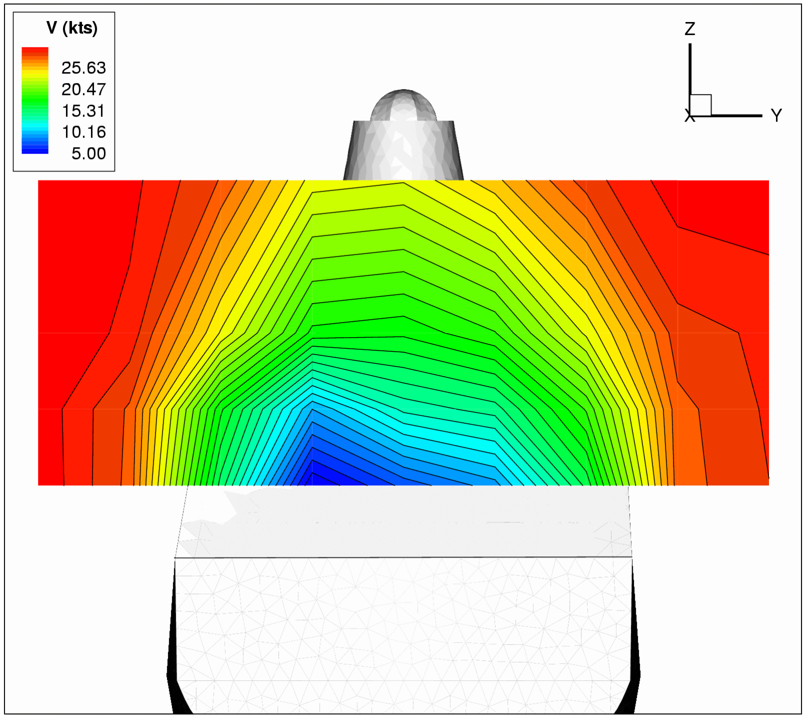
Airwake: LPD17 Results

Aerospace Engineering

Comparison of simulated time-averaged solution with experiments (0º yaw case).



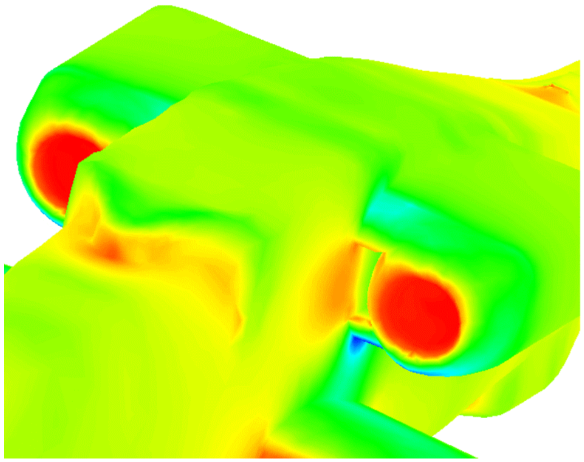
Simulated time-averaged



Experiments

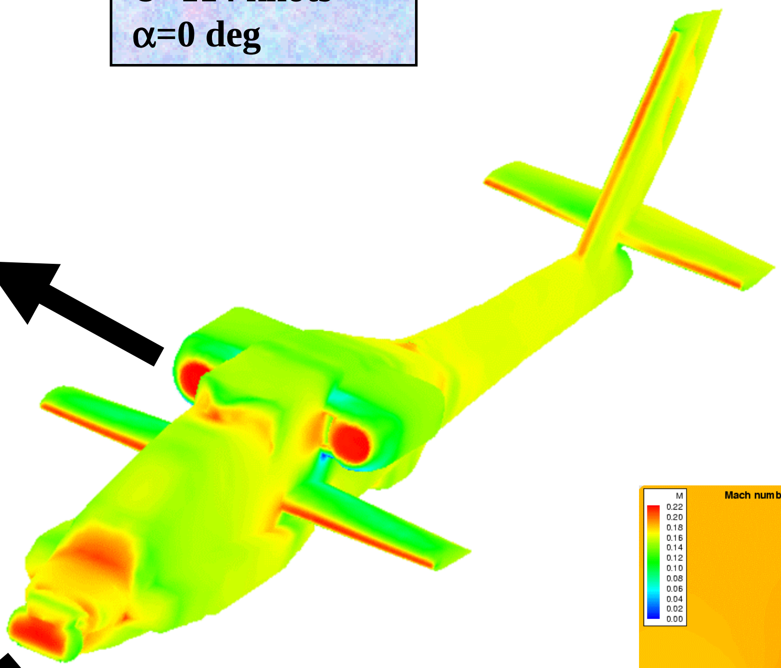
Results: Helicopter Configurations

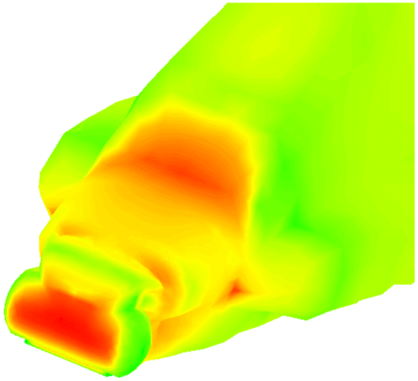
Aerospace Engineering



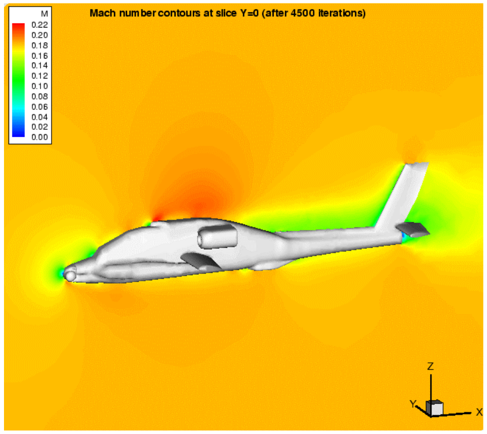
Flow conditions:
U=114 knots
α=0 deg

555,772 cells
1,125,596 faces
1.9 GB RAM



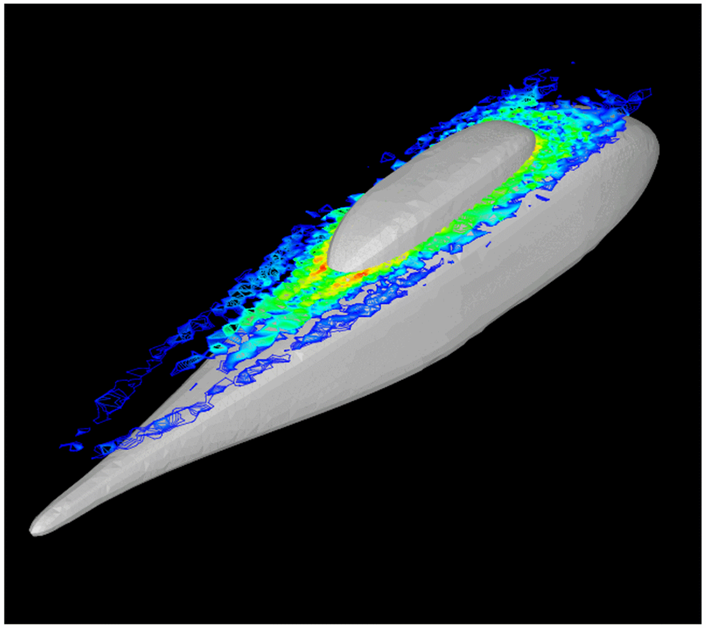


AH-64 Apache



ROBIN body

Unsteady flow over helicopter fuselage including turbulence effects using LES modeling (F90/MPI code).



Vorticity contours on the ROBIN body (343,000 cells) in the plane of the pylon/body intersection for a steady state solution where C_D = 0.028 and C_L=-0.02.

(Consumes approx 1 GB memory, 36 Mflops/proc on COCOA-2)

(Courtesy: LTC Robert Hansen, Prof. Lyle N. Long)

Other Applications

Penn State Aerospace Engineering

- We have also implemented a long-range acoustic scattering code on the cluster using F90/MPI. This code runs very well on the clusters, and because of the Cartesian grids can use extremely large numbers of grid points.
- We have also been using the cluster for supersonic jet noise simulations for many years. These are extremely difficult calculations since the far-field noise is primarily due to the turbulence in the jet. So unless the turbulence is computed properly, the far-field noise will be very inaccurate. We can now predict the far-field noise to within a few dB.
- Another application that runs very well on clusters are Monte Carlo methods.
- Long and Brentner also describe a method that is 100% parallel, a self-scheduling parallel method to predict helicopter noise using a parallel version of the Wopwop code.

Concluding Remarks

Aerospace Engineering

- Clusters of workstations running Linux can be extremely cost effective for engineering and scientific computing. (There are already about a dozen of these at Penn State, with more coming rapidly).
- While some applications will require low latency, high bandwidth networks (e.g. Myrinet), there are other applications (such as the ones shown here) which run quite well on inexpensive fast ethernet-based clusters.
- For large, time accurate (four dimensional) simulations, standard domain decomposition results in favorable ratios of communication to computations (due to the low surface to volume ratio).

Concluding Remarks

PENN STATE
Aerospace Engineering

- In order to meet the tremendous demand for scientists and engineers trained in using parallel computers, the Penn State Institute for High Performance Computing Applications (IHPCA) has established a Graduate Minor in High Performance Computing (http://www.psu.edu/dept/ihpca/).
- In the future, it will be essential to have experts trained in their chosen fields, who also understand parallel computing.
- The use and performance of commodity-based parallel computers will continue to be an important aspect of this education.

Web Addresses

Aerospace Engineering

- Prof. Lyle N. Long (publications, etc.):

  http://www.personal.psu.edu/lnl/

- COCOA (Installation, benchmarks):

  http://cocoa.ihpca.psu.edu

- IHPCA (courses, resources):

  http://www.psu.edu/dept/ihpca/

- PUMA:

  http://bart.ihpca.psu.edu/thesis/puma/