Perspectives for Parallel Computing in Multidisciplinary Design Optimization

Juan J. Alonso
Department of Aeronautics & Astronautics
Stanford University
jjalonso@stanford.edu

Panel on Ubiquitous and Massively Parallel Computing
10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference
Albany, NY, August 30, 2004
Some Facts…

• The good:
  – Parallel computing has reached the commodity market
  – Both performance and price/performance continue to improve
  – Parallel software is establishing its

• The bad:
  – Platform stability or lack thereof
  – We are still programming with MPI!
  – Large scale parallel software integration needed for MDO is still in its infancy
• The “easy” way
  – Run the same program 100 times on 100 different computers
  – Embarrassingly parallel
  – Poor/good algorithm poor/good implementation analogy
  – Sometimes the way to go (GAs, Monte-Carlo, etc.)
  – What about memory scalability?
• Is this all there is to it? What are CS departments across the world working on?
Current Uses of Parallel Computing (2 of 3)

• The “hard” way
  – Single discipline analysis/design where runtime is a bottleneck
  – Amdahl’s law and scalability in the iso-efficiency metric
  – Some successes...MPI domain decomp.
  – Memory scalability for fluids and OpenMP for structures
  – Main failure is that the codes become “arthritic” even when properly developed/modularized
  – Overall optimism in this category
Current Uses of Parallel Computing (3 of 3)

• The “impossible” way - multi-code parallelism and scalability
  – Multi-code, multi-discipline parallelism for design applications
  – Grid / distributed / secure computing…does it work?
  – Practically uncharted territory
  – Some encouraging efforts…still very early on
  – Hierarchical methods for design
  – Large-scale integrated simulations
  – Parallel integration environments…python?
What Do We Need and Why?

• Fundamentally, we need to worry about the problem, not the accidents of how to go about solving it.
  – Understanding problems / physical mechanisms
  – Fixing problems / making designs better
  – Development of tools for “faster, better, cheaper”

• Requirements (in our research)
  – Highly scalable discipline analysis and design modules
  – Distributed analysis and design environments
  – Scalable multi-discipline analysis and design
  – SQA, modularity, reusability
  – Faster, easier to use computers
Will This Happen? When?

- Software...maybe
- Hardware...do not hold your breath!
- We are in the business of high-performance scientific computing ~1-3% of the computing market
- In software, our needs are aligned with the business enterprise...leverage integration tools
- In hardware, our needs are pretty specific...we'd better start thinking about running codes on graphic cards!
- You cannot wait until everything is fixed to jump
Suggestions, Recommendations, Questions

• If you are in the business of scientific computing
  – Learn MPI if you have not already (you are behind…)
  – OpenMP will help you for a while…
  – Pay attention to emerging
    • Research in decomposition methods in MDO
    • New parallel architectures (streaming supercomputing?)
    • New software development and integration paradigms
      (Python, Grid, integration frameworks, etc.)

• This is an exercise in managing complexity - one has to make the difficult look easy.
• One day we may all think in parallel…or not!
Pyre Distributed services

Workstation

Front end

Compute nodes

Launcher

Monitor

Journal

solid

fluid

Michael Aivazis, Caltech