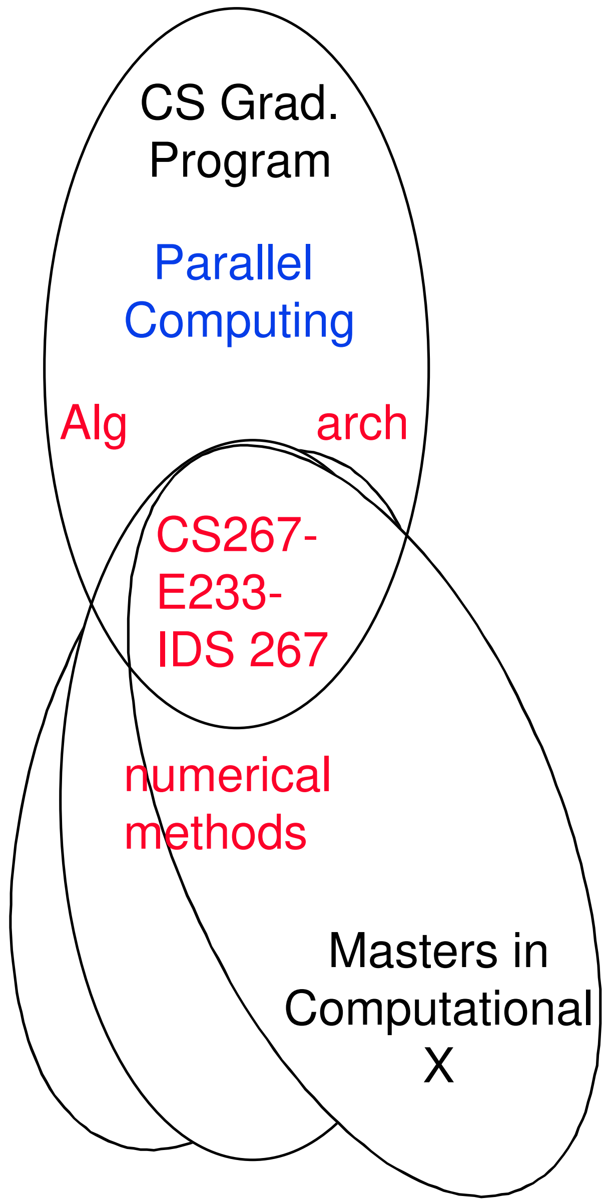
CS 267 Applications of Parallel Computers:

Teaching Parallel Computing on the Berkeley NOW

David Culler

http://now.cs.berkeley.edu/cs267/

Role of the Course in the Curriculae



Class of 97

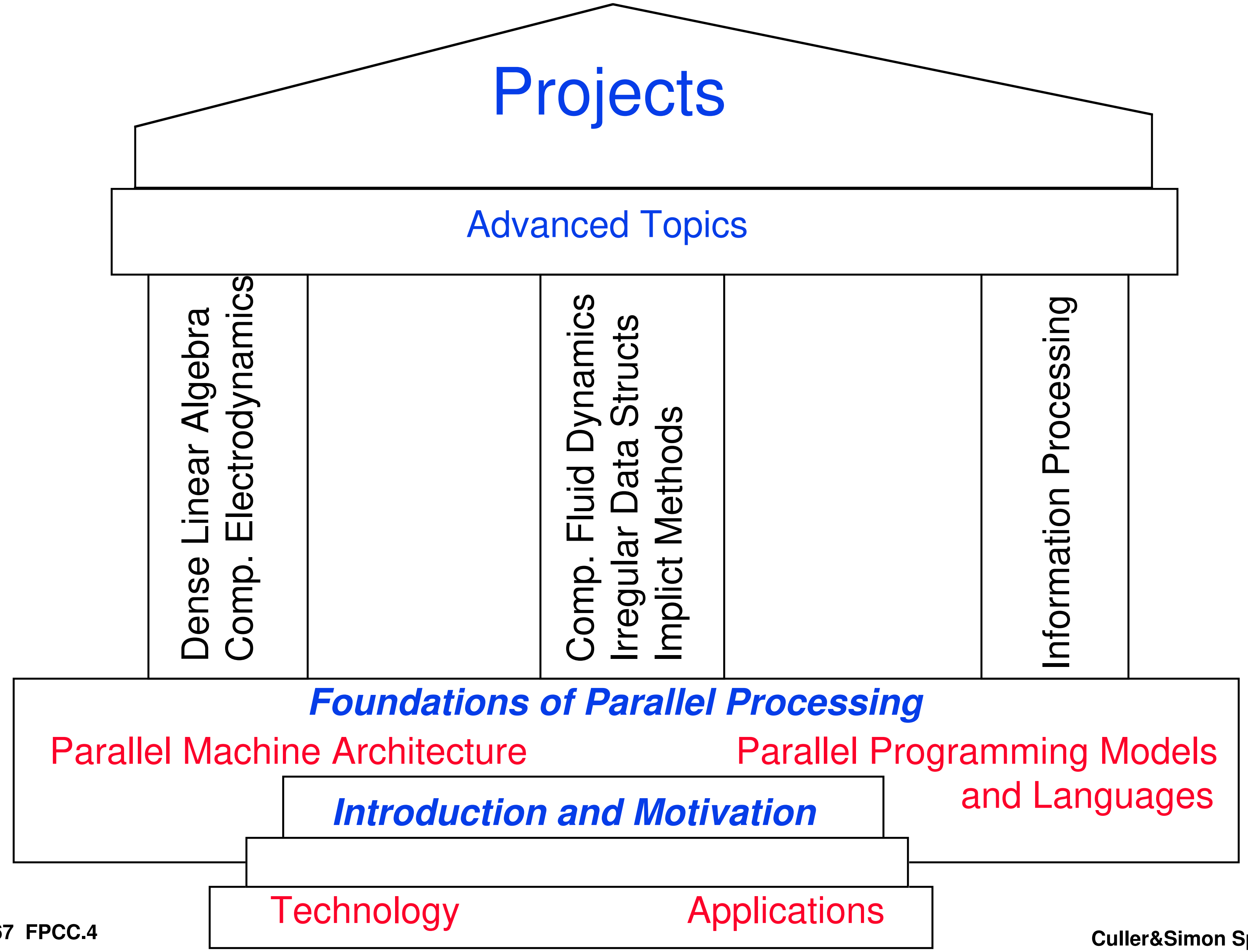
- 23 students (2 undergrad)
- Half from CS
- Rest from physics, math, materials science, chemistry, biophysics, engineering, astrophysics
- Recipe for failure:

  too much math for the CS types +

  too much programming for the disciplinary science types

- Team taught with Horst Simon @ NERSC

Course Architecture



Resources

- NOW
  - 100 Sun Ultrasparcs + Myrinet + Active Messages + Glunix
    - 167 MHz, 128 MB, 2x2.3 GB
  - MPI, HPF, Split-C
- Enterprise 5000
  - 4 8-way SMPs
    - 167 MHz, 256 MB, ...
  - MPI, P-Threads
- Cray T3D and Cluster of SMPs for projects

Lecture Components

- **Intro**
  - technology and applications
  - modeling perspective on the parallel programming challenge
- **Architecture**
  - small-scale SMPs - caches
  - large-scale parallel machines - communication
- **Programming Models and Languages**
  - Message Passing - MPI
  - Data Parallel Programming - F90, HPF
  - Structured Shared Address - Split-C
  - Shared Address - PThreads and ANL
  - Cannonical mistakes by example
- **Dense Linear Algebra**
  - computational electrodynamics -> LU factorization
  - all the TFLOPS tricks and techniques
  - software engineering for portable parallel libs - ScaLapack

Lecture Components (cont)

- **Project Proposals by external faculty**
  - field trips to visualization lab and SGI
- **Computational Fluid Dynamics**
  - unstructured meshes -> graph partitioning -> recursive spectral partitioning
- **Multidimensional FFTs**
- **Data Management**
  - Parallel sorting under LogP
  - External disk-to-disk sort
- **Computer Science Techniques**
  - Synchronization algorithms
  - Distributed Data structures (oct trees)
  - Connected Components
- **Computational Science Methods**
  - Structured Grid Iterative Solvers
  - Sparse Matrix Methods

Reading Materials

- 3 on-line texts
  - Jim Demmel's notes from CS267 Spring 1996
  - Culler and Singh's, Parallel Computer Architecture - alpha draft
  - Ian Fosters, "Designing and Building Parallel Programming"
    - bookstore should have hardcopies
- Papers
- the web ....

Programming Assignments

- High Performance Matrix Multiply on single processor
  - learn how critical is locality
  - instruction level parallelism & register usage
  - partitioning for locality ~ parallelism
  - performance is a challenge on any machine
- Sharks and Fish World
  - spectrum of basic problems in range of programming models
- 3D FFT Performance Competition
  - anything goes
- Projects

Sharks and Fish

- 1. Fish move continuously subject to an external current and Newton's laws. **(EP)**
- 2. Fish alone move continuously subject to gravitational attraction and Newton's laws. **(N-body)**
- 3. Fish alone play the "Game of Life" on a square grid. **(Local)**
- 4. Fish alone move randomly on a square grid, with at most one fish per grid point. **(Synchronization)**
- 5. Sharks and Fish both move randomly on a square grid, with at most one fish or shark per grid point, including rules for fish attracting sharks, eating, breeding and dying. **(Dist. Sim)**
- 6. Like Sharks and Fish 5, but continuous, subject to Newton's laws.

	1	2	3	4	5	6
Matlab	x	x	x	x	x	
HPF	x	x				
MPI	x	x	x			
Split-C	x	x				
Sun Threads	x	x				

3D FFT Competition

- Motivated by ASCI proposal effort
  - validate AMR based turbulence via brute-force FFT approach
- Previous programming competitions
  - sorting, traveling salesman
- Algorithm and language was open
- Used great Java demo to learn about FFT
- Fundamentally learn about all-to-all personalized communication
- Learned to debug parallel numeric code
  - visualization tools
- Not enough algorithmic variation

3D FFT results

Projects

- SVD on the T3E (Lanzsos + Sparse Matrix-vector Mult)
- 3D FFTs on Network of SMPs
- Two-Dimensional Quality Mesh Generation and Delaunay Triangulator in Parallel
- Sparse LU on Distributed Memory Multiprocessors and Nonradial Oscillations in Rotating Stars
- Parallel Collision Finding and Cost-Effective Mints for E-Commerce Micropayments
- Communication Patterns for Block-Structured Scientific Applications on KeLP
- Parallelizing the Time-Dependent Schrodinger Equation
- Parallel Name Servers
- Parallel Isosurfacing in Span Space on the Cray T3E
- Parallel Renderer

What students should get out of the course

- In depth understanding of:

(1) how to apply parallel computers to demanding problems

(2) requirements of parallel applications and their programmers

(3) hardware, software, theory and practice of parallel computing

What it does for the research

- CMAM developed for Sp 92
- Split-C came out of the experience and used in S93
- Mantis parallel debugger is S94
- Inktomi was a class project
- Many other projects became theses and publications
- Sp97 had 25 users with common deadline pounding on NOW
  - developed MPI over Active Messages
  - HPF over MPI
  - Glunix reservation system