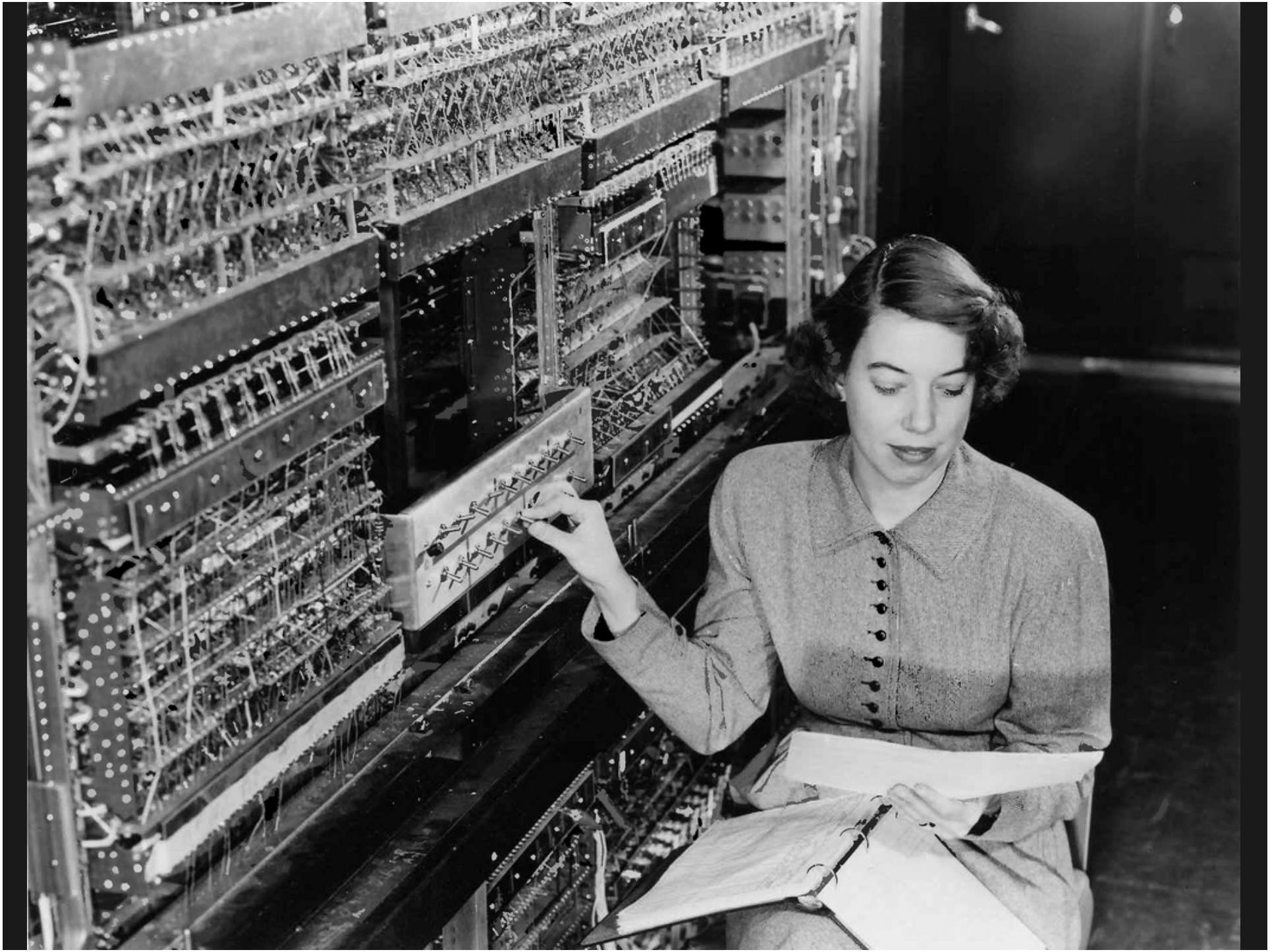
An Introduction to Parallel Supercomputing

Pete Beckman

Argonne National Laboratory



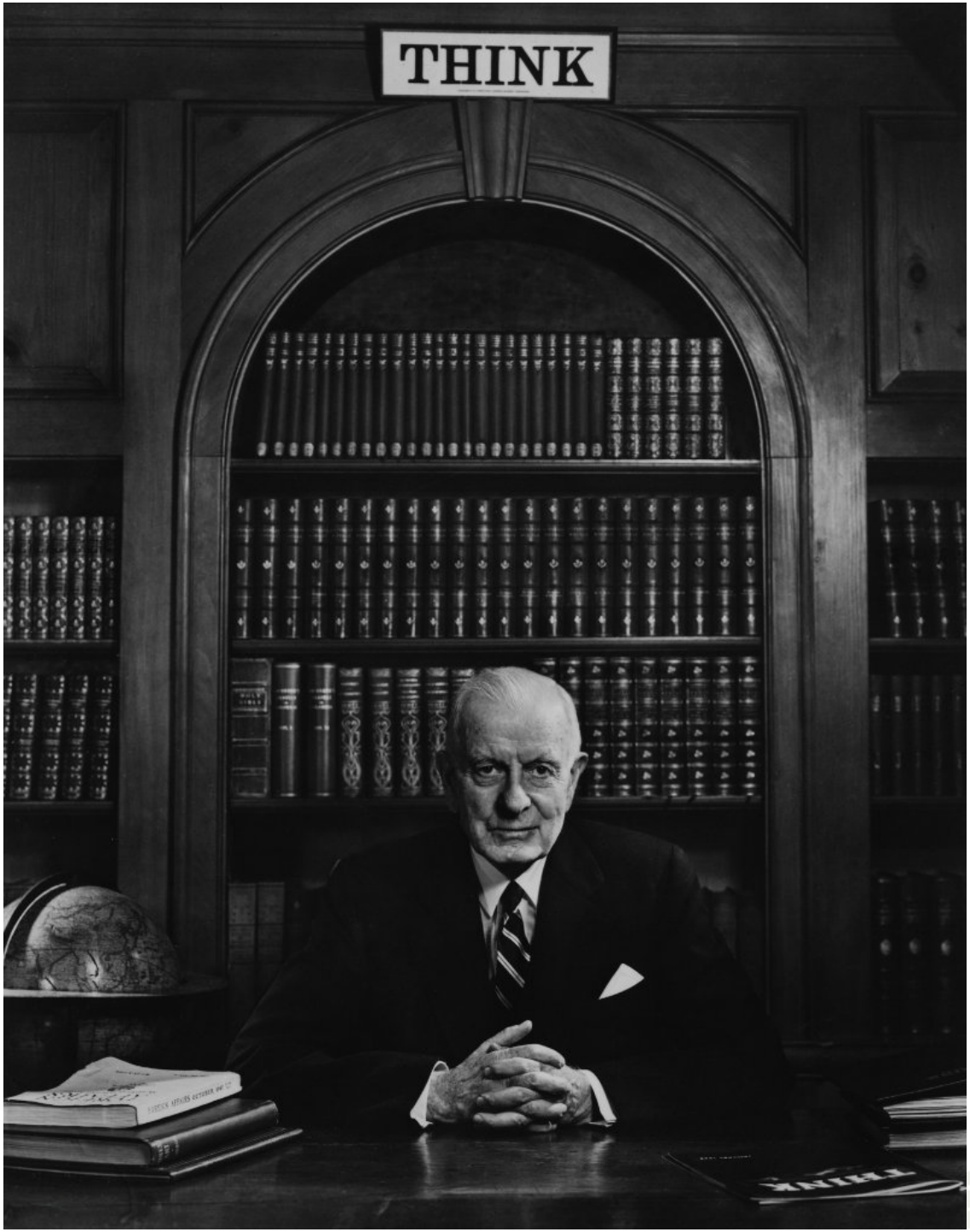
MCS Division meeting c. 1983

- "If our R&D is going to be relevant ten years from now, we need to shift our attention to parallel computer architectures"
- "Los Alamos has a Denelcor HEP: let's experiment with it"

The ACRF Summer Institutes

- Sponsored by NSF
- Held each September, 1987-1989
- Mornings: long lectures by distinguished speakers
- Afternoons: hands-on experience with Argonne software on ACRF machines
- Some speakers: Gordon Bell, Bill Buzbee, Josh Fisher, Dave Kuck, Neil Lincoln, Chuck Seitz, Larry Smarr, Burton Smith, Guy Steele, Don Austin, Mani Chandy, Arvind, Tom DeFanti, David Gelernter, John Gurd, Ken Kennedy, Alex Nicholau, others
- Admission by application and review, about 20 students each summer

And now….. 2013 ATPESC!



Understand the Model

POOMA Project: 1996
John Reynders

Parallel Platform Paradox

"The average time required to implement a moderate-sized application on a parallel computer architecture is equivalent to the half-life of the latest parallel supercomputer."

"Although a strict definition of "half-life" could be argued, no computational physicist in the fusion community would dispute the face that most of the time spent implementing parallel simulations was focused on code maintenance, rather than on exploring new physics. Architectures, software environments, and parallel languages came and went, leaving the investment in the new physics code buried with the demise of the latest supercomputer. There had to be a way to preserve that investment."

Pete's Investment Recommendations

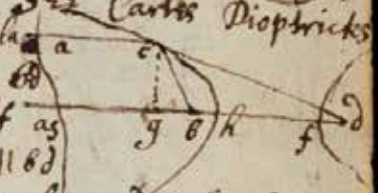
- **OPM (Other People's Math (libraries))**
- **Encapsulation**
  - Parallelism & Messaging & I/O
- **Embedded Capabilities**
  - Debugging
  - Performance Monitoring
  - Correctness Detection
  - Resilience
- **The Two Workflow Views**
  - Science: (problem setup, analysis, etc.)
  - Programmer: (mod, testing, document, commit)
- **Automation**
  - A+ Build system, nightly test and build, configuration
  - Embedded versioning and metadata
- **Community: web, tutorial, email, bug tracking, etc**

Future Trends: Invest Wisely

Trending Up	Trending Down
Asynchrony, Latency Hiding	Block synchronous
Over Decomp & Load Balancing	Static partitioning per core
Massive Parallelism	Countable parallelism
Hierarchical Memory	Flat Memory
Expensive Data Movement	Expensive flops
Fault / Resilience Strategy	Pure checkpoint/restart
Low BW to Storage, in-situ analysis	Save all, let the viz guys sort it out

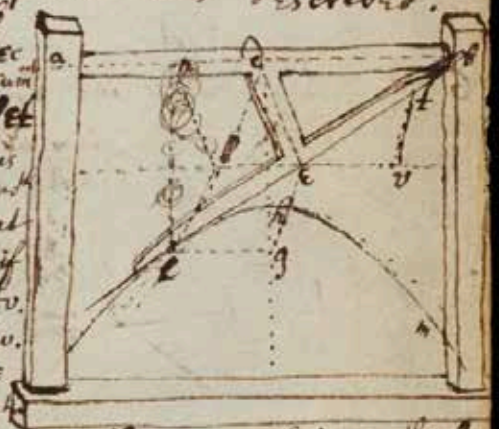
Of Refractions.

1 If ye ray ac bee refracted at the center of ye circle acdg towards d & ab ⊥ be ⊥ gc ∥ ed. Then suppose ab : ed :: d : e. See Cartes Dioptricks



2 If there be an hyperbola the distance of whose foci bd are to its transverse axis hf as d to e. Then ye ray ac ∥ bd is refracted to ye exterior focus (d). See C: Diopt

3 Having ye proportion of d to e, or bd : hf. The Hyperbola may bee thus described.



1 Upon ye centers a, b let ye instrument adbtec bee moved in wch instrum observe yt ad ⊥ de ⊥ cet & yt the beame cet is not in ye same plane wth adbe but intersects it at ye angle tev so yt if tv ⊥ ev, then d : e :: et : tv. Or d : e :: Rad : sine of ∠tev. Also make de = $\frac{q}{2}$ i.e half ye transverse diameter. Then place the fiducial side of ye plate chm in the same plaine wth ab. & moving ye instrument adbect to & fro its edge cet shall cut or weare it into ye shape of ye desired Parabola. Or the plate chm may bee filed away untill ye edg cet exactly touch it every where.

2 By the same proceeding Des=Cartes concave Hyperbolicall wheele may bee described by beeing turned wth a chissell dtec whose edge is a streight line inclined to the axis of the mandrill by ye ∠tev wch angle is found by making d : e :: et : tv :: Rad : sine of etv.

3 By the same reason a wheele may be turned Hyperbolically concave ye Hyperbola being convex. Or a Plate may bee turned Hyperbolically concave