GRAPE-6 and beyond

Jun Makino

University of Tokyo

Talk overview

- GRAPE Project
- GRAPE-6
- GRAPE-8?
- Summary

GRAPE project

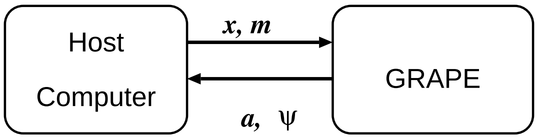
GOAL:
Design and build specialized hardware for simulation of dense stellar systems.

Rational:
You can do larger simulations (better resolution) for same amount of money.

	Year	Peak	Cost	Cost/TF
GRAPE-6	2002	64 TF	4M$	0.07
ASCI White	2001	12 TF	200M$	17
Earth Simulator	2002	40 TF	300M$	7.5
ASCI Q	2003	30 TF	200M$	6.7
IBM BG/L	2005?	400 TF?	50M$?	0.13?

Basic idea of GRAPE

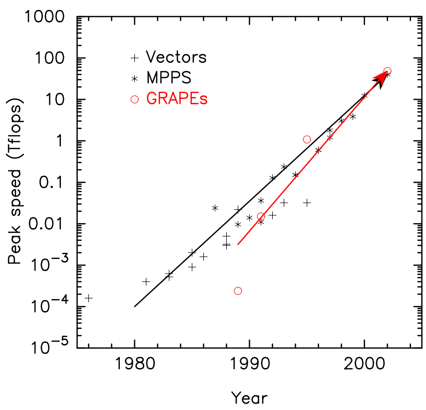
Special-purpose hardware for force calculation
General-purpose host for all other calculation



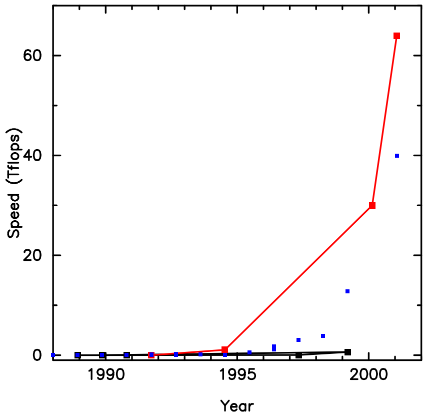
Advantages of GRAPE

- Special-purpose hardware
  - Hardwired logic → No software needed
  - Pipelined processor → High performance
- General-purpose host
  - Programmable in high-level languages
  - existing programs can be used with “small” modifications
  - can be used with Barnes-Hut tree/Aarseth scheme

Evolution of GRAPE systems



A different way to show the same data



Overall structure

- 4 clusters
- 16 processor boards per cluster
- 8 processor modules per board
- 4 chips per modules
- 6 pipelines per chip
- 60 arithmetic units per pipeline

In total, 7.5×10^5 Arithmetic units operating at 90 MHz

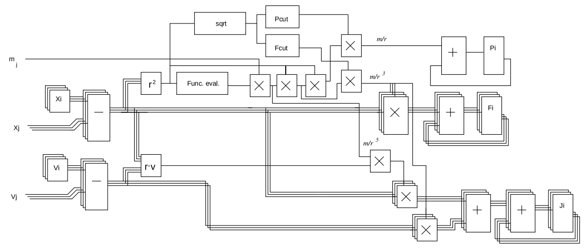
GRAPE-6 Processor Chip

- 0.25 μm custom design (Toshiba TC240E, 1.8M gates)
- 2.5 V power supply, 90 MHz operation
- 6 pipelines in a chip
- $>$ 30 Gflops (equivalent) per chip
- system clock: 22.5 MHz
- Single chip equivalent to single GRAPE-4 board

GRAPE-6

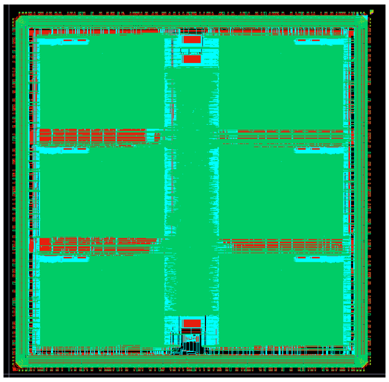
- Peak speed 64 Tflops
- Improved communication speed (2GB/s)
- Minor fixes for some "features" of GRAPE-4...

GRAPE-6 Processor pipeline



Calculates gravitational force, its first time derivative and potential.

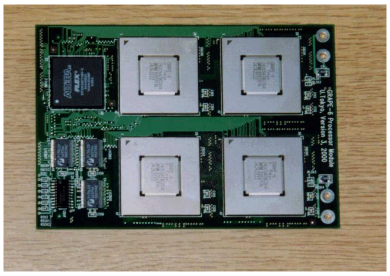
GRAPE-6 Processor Chip



GRAPE-6 vs Intel P4

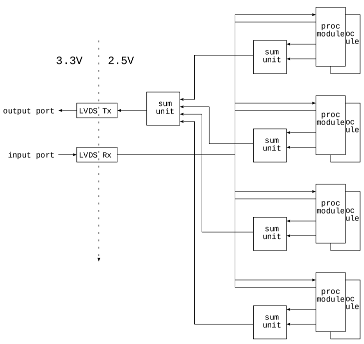
	G6	P4
Design Rule (μm)	0.25	0.13
No. transistors	7M	40M
Clock freq.	90M	3.2G
Power Consumption	15W	$\sim$ 100W
No. ALUs	$\sim$ 400	2 (SSE2)
Peak speed(GF)	31	6.4

GRAPE-6 processor module



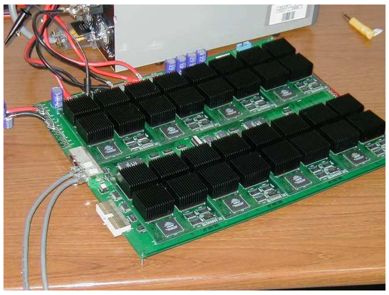
- 4 processor chips + Memory chips.
- Integrates all functions of GRAPE-4 processor board.

The GRAPE-6 Processor board

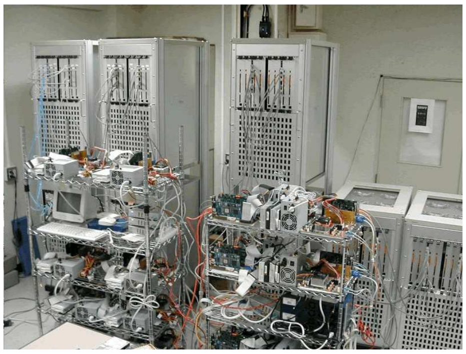


- Single board with 32 chips
- Semi-serial (LVDS) inter-face (350MHz clock, 4 wires)
- Two input ports and one output port
- Result from 32 chips are summed by a reduction tree

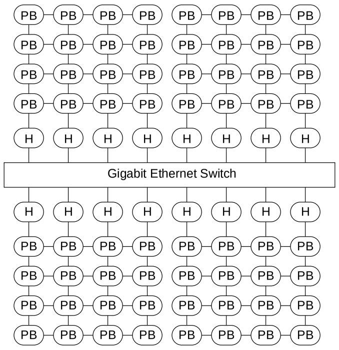
The GRAPE-6 processor board



The 64-Tflops GRAPE-6 system



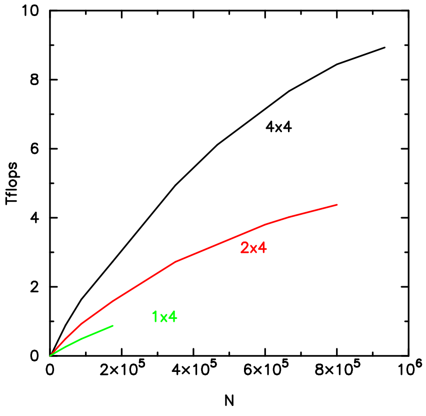
The full 64 Tflops GRAPE-6 system



- 4-host, 16-board "block" with dedicated network
- 4 "blocks" connected through GbE network

For parallel code:
Host-Host communication not needed within block
Inter-block comunication can be done in parallel

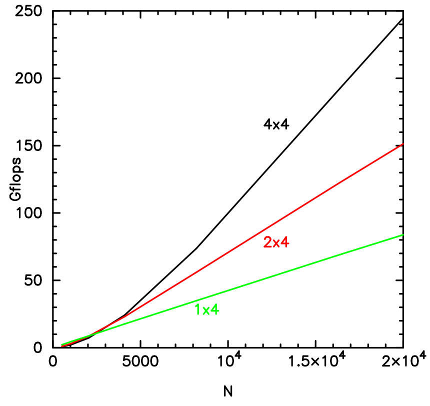
Performance of GRAPE-6



16-board, 8-board and 4-board system with 4, 2 and 1 hosts, respectively

Pentium 4 hosts connected by 100BT Ethernet (MPICH/p4)

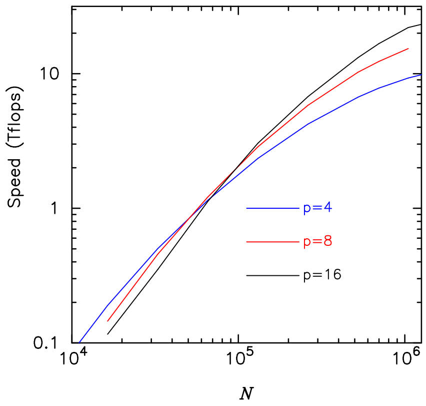
A closer look



Even with slow MPICH, parallel overhead is small if $N > 10^4$.

Host overhead is still large...

Performance (more than 4 hosts)

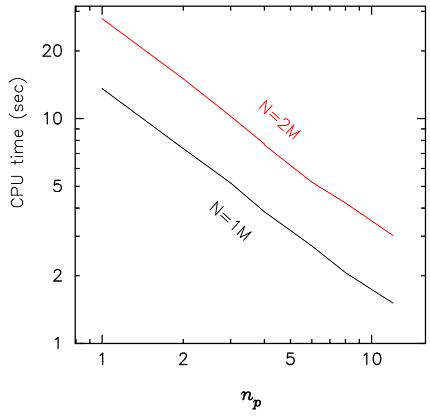


4, 8, 16-node systems (16, 32, 64 Tflops peak)

Hosts: AMD Athlon XP 1800+, GbE (MPICH/p4)

Linux 2.4.16
gcc 3.2

Treecode Performance



CPU time in seconds

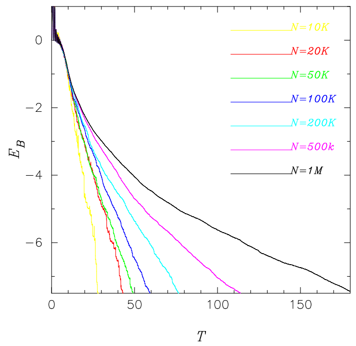
1-12 nodes, 1M/2M particles, $\theta = 0.75$
Plummer model

GRAPE-6 Scientific achievements

- Not easy to find new results on collisional systems which is not done on GRAPE-4/6
- Highlights: Talks today (Shara, McMillan, Aarseth, Baumgardt ...)

Good enough reason to continue with GRAPE (I think).

BH binary (Makino and Funato 2003)



Details will be given in JD11...

Current limitations with GRAPE-6

(not necessarily related to hardware)

- Performance of large codes (NBODY4, Kira) limited by the speed of the single host (reads: Beowulf is very cheap)
- PCI speed (133 MB/s) will be the bottleneck in a few years.
- Direct summation inefficient with large number of particles
- hardware reliability/software trouble ....

R & D directions

- parallel code (application/communication library)
- faster host interface
- hardware support/new algorithm for tree/neighbour scheme
- faster hardware — GRAPE-8

GRAPE-8?

What can we build in 5 years from now?

- Technology: 90-65 nm
  - 6-10 times more transistors
  - 2-3 times faster clock

Assuming we can use the same amount of money for the mass production of the chip, 1–3 Petaflops is within reach.

Hardware Design challenge

Major issues

- What host computer should we use?
- How should we connect hosts and GRAPEs?
- How should we organize the memory architecture?
- How can we get funded?

Chip Initial Cost — The end of special-purpose computer?

Initial cost increases as semiconductor technology advances

	Year	Design Rule	Approx. Initial cost
G4	1992	1 μm	0.25M$
G6	1997	0.25 μm	1.5M$
G8	2004?	90 nm	$>$ 3M$

Difficult to justify big money for narrow range of applications...

Solution for the initial cost problem?

Possible approaches

- widen application range
  - Molecular Dynamics
  - Bouldary value problem (*e.g.*, Helmholz eq.)
  - ...
- reduce initial cost
  - FPGA
  - FPGA “Hardcopy”

Far from ideal...
A completely new approach might be necessary.

Summary

- 64 Tflops GRAPE-6 is up and running.
- Multiple-petaflops GRAPE-8 seems technically feasible, but getting funded is not easy...

Host Parallelization

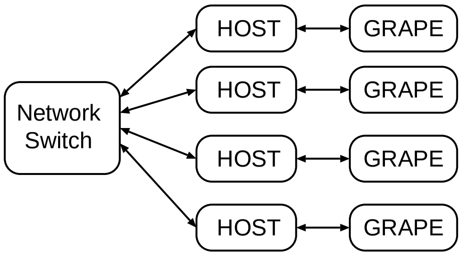
Major challenge: We need parallel host to serve GRAPE-6

- GRAPE-6 is 100 times faster than GRAPE-4
- single CPU would become only 10 times faster in 5 years

We have to use 10 hosts in parallel for GRAPE-6
How?

A Naive approach

Connect multiple GRAPEs and multiple hosts



Works fine for treecode
Does not work for direct summation

Two methods for parallelization

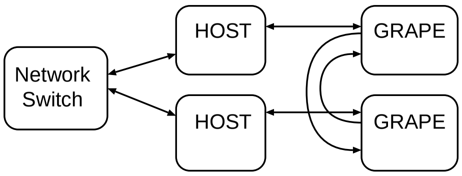
- Each host has complete copy of the system. Time integration is distributed.
- Each host has its own share of particles.

In either case, each GRAPE (and therefor each host) need the information of all particles to calculate the forces on its particles.

The communication bandwidth of a single host limits the scalability

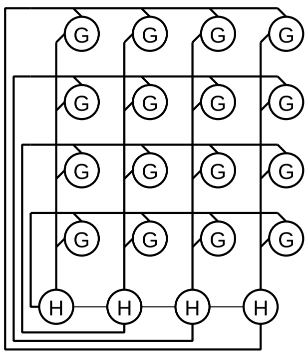
Solution in GRAPE-6

Each GRAPE box has multiple input ports from which it can receive the data from all hosts in parallel.



For n hosts, GRAPE has n input ports.

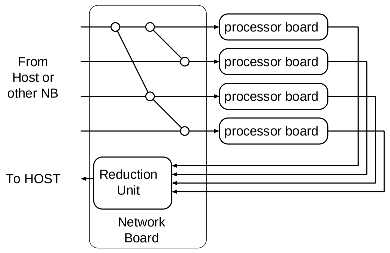
Network structure



No host-host communication required (except for synchronization)

Low-latency message-passing interface is desirable
SCore? GAMMA?

A more flexible system

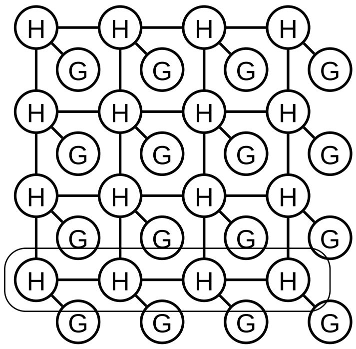


Can be configured for the broadcast from a single host, direct connections from n hosts, and multicasts from $n/2$ hosts etc.

An alternative solution

- Configure n^2 hosts as a $n \times n$ grid.
- All hosts other than a single row of n hosts are used just as the network nodes described before.
- Cheap PCs woth GbE are fine
- Final GRAPE-6 partly uses this approach

The host network solution



This approach can be used for paralleliza- tion on MPPs with- out GRAPE.

Much simpler than the Hyper-Systolic stuff...