Cosmogrid Project

Jun Makino

Director, Center for Computational Science National Astronomical Society Observatory of Japan with: Kei Hiraki, Mary Inaba, Tomo Ishiyama (U. Tokyo), Keigo Nitadori (RIKEN), S. Portegies Zwart, Derek Groen, Stefan Harfst (Leiden University), Cees de Laat (University of Amsterdam), S. L. W. McMillan (Drexel University), and Cosmogrid project members



Talk Summary

- We performed large-scale cosmological *N*-body simulation, on a "grid" of two supercomputers connected via 10G line(s): Cray XT4 at CfCA/NAOJ and Huygens(IBM p6/575) at SARA/UvA.
- Efficiency of calculation for production runs $(2048^3 \text{ particles})$ is pretty high (better than 80%)
- (The communication throughput during the production runs is less than impressive though...)
- For daily use, the main difficulty is how to schedule two or more machines for single calculation...

Talk Overview

- Brief overview of NAOJ
- Brief overview of CfCA
- Cosmogrid Project
 - Background
 - What we tried to do
 - What we have achieved so far
 - Future directions

Brief overview of NAOJ

- National center for Japanese Astronomy
- Ground-Based observation facilities
 - Subaru Telescope
 - Nobeyama Radio Observatory
 - Okayama Observatory, and several other small telescopes
- Space Astronomy: Collaboration with ISAS/JAXA (Akari, Hinode)
- Theoretical and Computational Astronomy



Subaru Telescope

Hawaii, Mauna Kea 8.2m mirror



Only eight-meter-class telescope with a prime-focus camera (Suprime-Cam), 30 arcminutes field of view (100 times that of Hubble Space telescope) Plan to extend to 2-degree field of view (Subaru HSC)

Nobeyama Radio Observatory



Started operation in 1982. First world-class groundbased observation facility in Japan Follow-up: ALMA (US-EU-Asia joint project, observation will start in 2010?)

Other projects (incomplete list...)

- Hinode (Solar observation Satellite)
- TAMA300 (Experimental gravity Wave detector)
- VERA VLBI (Very Long Baseline Interferometer)
 - Real-time signal processing using optical network
- VSOP-2 (Space VLBI)

Center for Computational Astronomy

Two goals:

- Theory group within NAOJ
- Computer Center for Japanese (and International) Theoretical/Computational Astronomy

History

- 1965 "Domestic Computing Facilities " "for Artificial Satellites"
- 1988 ADAC (Astronomical Data Analysis center)
- 1988 Division Theoretical Astronomy
- 2006 reorganized to ADC (Astronomical Data Center) and CfCA

A unique center dedicated to "Theory and Simulation for Astronomy/Astrophysics"

Evolution of CPU power



(till 2001)

Year - log of computing speed (12=1Tflops) Open: U. Tokyo Filled: ADAC/CfCA 1/10 — 1/100 of UT

Current System

- Vector-Parallel: NEC SX-9 16CPU+8CPU
- Scalar: Cray XT4 9 cabinets (812 nodes, 28.6TF)

CfCA Cray XT4



Front panel CG



Simulation: Takayuki Saitoh (CfCA/NAOJ) Visualization :Takaaki Takeda (4D2U/NAOJ)

Cray box in ORNL, before and after 2008

before



We also have:



GRAPE-DR Accelerator with the peak DP speed of ~ 1 Tflops/card A 144-node system up and running

Hardware completed. Tuning underway.

CosmoGrid Project

- Background
- What we tried to do
- What we have achieved so far
- Future directions

Background

• In "Principle"

- We want to do large-scale simulations which are impossible on a single supercomputer, by connecting multiple supercomputers with highspeed grid.

• In "reality"

- One of the goals of "GRID" is the above. But we rarely see successful examples of one single large-scale calculation actually performed.
- If we can demonstrate a working example,
 - * We may be able to get large chunks of machine time (thus effectively make it possible to do calculations not possible in single supercomputer center)
 - * This is interesting and potentially useful research project.

Large-scale parallel simulations on Grid?

- Lots of papers claiming "we did it"
- Almost all works are on job-level parallelism, using Grid-based job schedulers. Not a single large calculation.

Why?

Large-scale parallel simulations on Grid?

- Lots of papers claiming "we did it"
- Almost all works are on job-level parallelism, using Grid-based job schedulers. Not a single large calculation.

Why?

Because Grid is a "parallel computer" which is the most difficult to use

- Extremely small communication bandwidth (1/1000 of a typical IB cluster)
- Impossibly large communication latency (1000 times that of typical IB network)

Large-scale parallel simulations on Grid? (cont'd)

So why to try the most difficult to use environment?

- To get more cycles
- We will improve simulation algorithms

Latency-tolerant algorithms which are not bandwidth-demanding High efficiency and high scalability on future large-scale parallel machines

What we did

- Performed Cosmological *N*-body simulation
- On a grid of Cray XT4 of CfCA/NAOJ and Huygens (IBM p6/575) at SARA/UvA connected by 10G network
- \bullet Test calculation with 256³ particles: successfully completed
- 2048³: Succeeded to run several timesteps on Grid.
- Currently developing softwares to use more than to machines in EuroGrid.

What is a cosmological *N*-body simulation?

Big bang



After Big bang, "hot" universe starts to expand As the universe expand, the temperature drops

Density fluctuations starts to grow through gravitational instability \leftarrow We simulate this process Animations: (a)(b)

What can we learn from simulations?

- How galaxies (or what accounts for about 85% of its mass: dark matter halos) formed
- Theoretical predictions for mass and size distribution of galaxies, spacial distribution (statistical properties)

By comparing these theoretical predictions with observations, we can get some knowledge on what makes up the dark matter:

- Mass of one particle, total mass
- If there are any interactions other than gravity
- etc etc...

Why Grid?

- State-of-the-arts calculations are truly of large scale ($N\sim 10^{10})$
- Number of steps is rather small (less than 10^5)
- Size of calculation limited by CPU speed, several minutes per timestep.
- Parallelization very well studies and understood.

In other words,

- Very long latency (more than 100 msec) is okay, since one timestep takes more than 100 seconds and communication occurs less than 10 times per step.
- Amount of communication is also small: $O(N^{2/3})$.
- We do want to do large calculation. If using Grid helps to get more cycles, then...

Grid Structure



The network environment itself is similar to the experimental setup used by Hiraki and coworkers. To that setup, we connected two supercomputers at the both end.

Network performance

- latency (ping rtt) 280ms
- Bandwidth: Cray XT4's 10G card is the bottleneck. S2IO Xframe with PCI-X interface, connected to AMD 8131.
- Cray-Cray loopback test (loop at Chicago or Amsterdam): up to 6Gbps



Calculation code

- MPI within a site
- One of MPI nodes is dedicated to communication. All communications with the other site is through this "communication node"
- \bullet This "communication node" actually talks with another I/O node with TCP/IP
- \bullet I/O nodes in two sites communicate with TCP/IP



Development of the calculation code

- In-site parallelization: used our existing code (Developed by T. Ishiyama)
- Collection of data to the "communication node": Fairly easy
- Communication between I/O nodes: Various performance tuning required. Prof. Hiraki knows every tricks.
- Need to dynamically change the domain decomposition to reach good load balance (extension of what is done in in-site parallel code)

Nothing difficult, just a lot of bookkeeping...

How real experiments took place

- We ask collaborators to set up the network for two weeks or so.
- Lots of troubles happen and ping starts to work after ... days.
- More troubles happen in supercomputers...

International collaborative experiments are very difficult.

Simulation completed so far

 $N = 256^3$ run Communication performance ($N = 2048^3$, 1024 cores at both ends)

	Data size(MB)	Time (sec)	Fraction $(\%)$
Calculation		350	86
Grid for FFT	70	5.8	1.4
Particles exchange	770	51	13

- The calculation is already pretty efficient (communications less than 15% of total time)
- A factor of 10 reduction of communication time should be possible.
- For larger N, relative cost of communication decreases (as $N^{-1/3}$)

Grid of two machines in different continents is actually useful for large-scale cosmological N-body simulations.

Caveats

Practical difficulties

- Both machines are operated with job-queuing systems. How do we schedule two machines to start the calculation at the same time?
- Can we exclusively use fast network for extended period? (maybe note so severe: 1Gbps is fine)

Scheduling issue is critical if we are to do production runs, not just a few experiments. Other users would complain if we give Grid jobs special priority.

Possible solution for scheduling

(Not implemented yet, just ideas)

- Schedule two machines independently.
- When both machines are available do grid calculation
- When only one machine is available continue with non-grid run
- When the number of machines changes, migrate the workload dynamically.

Theory is simple, but...

Summary



- We developed an *N*-body code for cosmological simulation using NAOJ XT4 and UvA p6/575 connected by 10G network.
- Performance of TCP/IP communication is more than enough
- In real calculation, TCP/IP performance is not impressive, but still sufficient.
- In other words, we have shown this kind of calculation can be run efficiently on multi-continent Grid.
- Practical issues: Do we (managers of computer centers) want this kind of Grid jobs? How to schedule? Should applications be modified to keep other users happy?

Numerical methods

- Wide range in both space and time
 - Spatial: Gpc to 10 km (radius of neutron stars)
 - Time: Gyr to milliseconds

Numerical integration should ideally cover these ranges.

Time domain: Individual timestep



(Aarseth 1963)

- Each star has its own time and timestep
- Event-driven integration: — star with minimum $t_i + \Delta t_i$ is selected

Requirements for integration scheme

- High-accuracy predictor necessary
- Variable stepsize necessary
- Cannot use scheme which require the calculation of acceleration at intermediate points (eg: Runge-Kutta)
 - Linear Multistep method OK
 - Runge-Kutta not OK
 - Symplectic schemes not OK

Space domain

How do we calculate the right-hand side of the equation of motion?

For a while we forget about the individual timestep scheme...

Widely used method: Barnes-Hut treecode Widely know method: Fast-multipole method (FMM)

Basic idea for tree method and FMM



\mathbf{FMM}

- Tree: aggregate stars which exert the forces
- FMM: aggregate both side

How do we aggregate — Barnes-Hut tree

Use tree structure

- First make a cell with all stars in it
- Recursively subdivide the cells to 8 subcells
- Stop if there is small enough stars



•		•	•	•	•			•	•
•	•				•	٠			
•		••		•			•	•	
		•					•		
•	•	•		•		•	•		
_									
	•		•			•			•