SPH Astrophysics – “State of Art”.

Peter Berczik

Astronomisches Rechen-Institut (ARI),
Zentrum für Astronomie Univ. Heidelberg, Germany

berczik@ari.uni-heidelberg.de

SPHERIC 3rd, Lausanne, Switzerland, 4th – 6th June 2008

Presentation plan:

- **Gas (particle) physics in astrophysics.**
  - **Astronomical observations**
  - **N-body inspiration ☺**
- **Astrophysics SPH equations.**
- **Numerical astrophysics.**
- **Hardware accelerators.**
- **Recent multi-phase results.**

Presentation plan:

- Gas (particle) physics in astrophysics.
- Astrophysics SPH equations.
  - Basic Equations
  - Cooling Function
  - Smoothing Length
  - Self Gravity
  - Time Integration
  - SPH - test
- Numerical astrophysics.
- Hardware accelerators.
- Recent multi-phase results.

Presentation plan:

- Gas (particle) physics in astrophysics.
- Astrophysics SPH equations.
- Numerical astrophysics.
  - Computers
  - Codes
  - Results
- Hardware accelerators.
- Recent multi-phase results.

Presentation plan:

- **Gas (particle) physics in astrophysics.**
- **Astrophysics SPH equations.**
- **Numerical astrophysics.**
- **Hardware accelerators.**
  - **- GRAPE (only gravity)**
  - **- MPRACE/FPGA (gravity + SPH)**
  - **- GPU!!! ☺ (gravity + SPH)**
- **Recent multi-phase results.**

Presentation plan:

- Gas (particle) physics in astrophysics.
- Astrophysics SPH equations.
- Numerical astrophysics.
- Hardware accelerators.
- Recent multi-phase results.

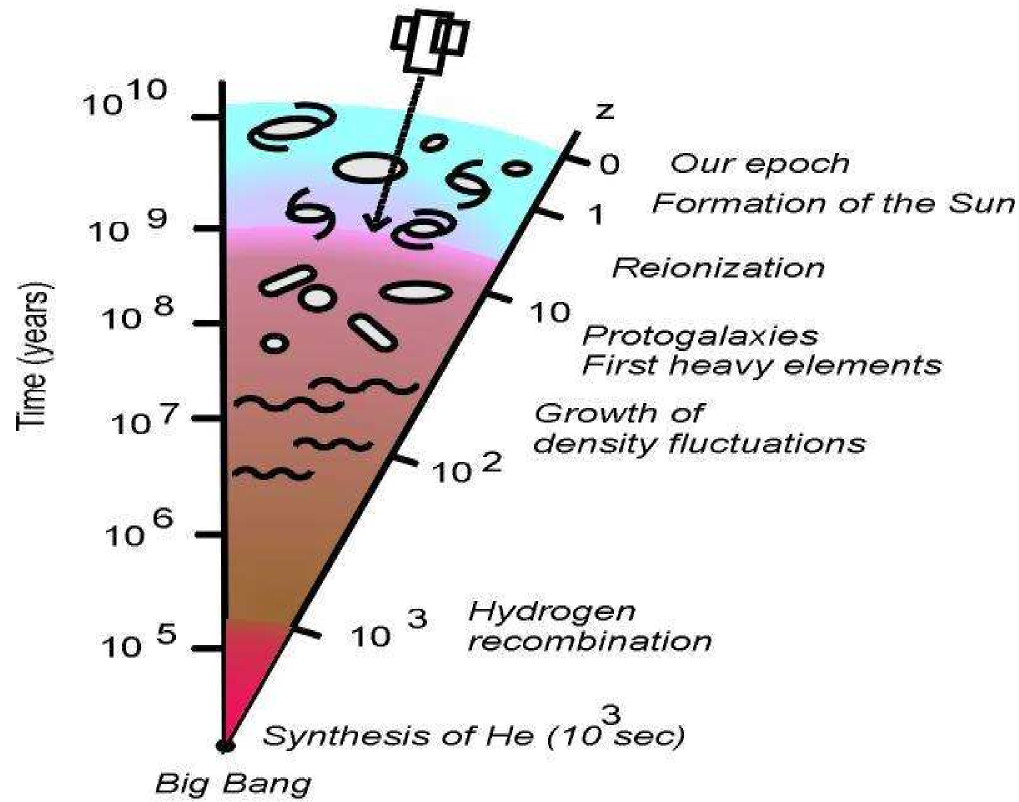
- Speedup
- Accuracy
- First results

Collaborators & Grants:

- **Naohito Nakasato** — **Univ. of Aizu, Japan**
- **Keigo Nitadori** — **Tokyo Univ., Japan**
- **Ingo Berentzen & Rainer Spurzem** — **Univ. Heidelberg**
- **G.M. Martinez, G. Lienhart, A. Kugel, R. Maenner** — **Univ. Mannheim**
- **A. Burkert, M. Wetzstein, T. Naab, H. Vasquez** — **Univ. Munich**

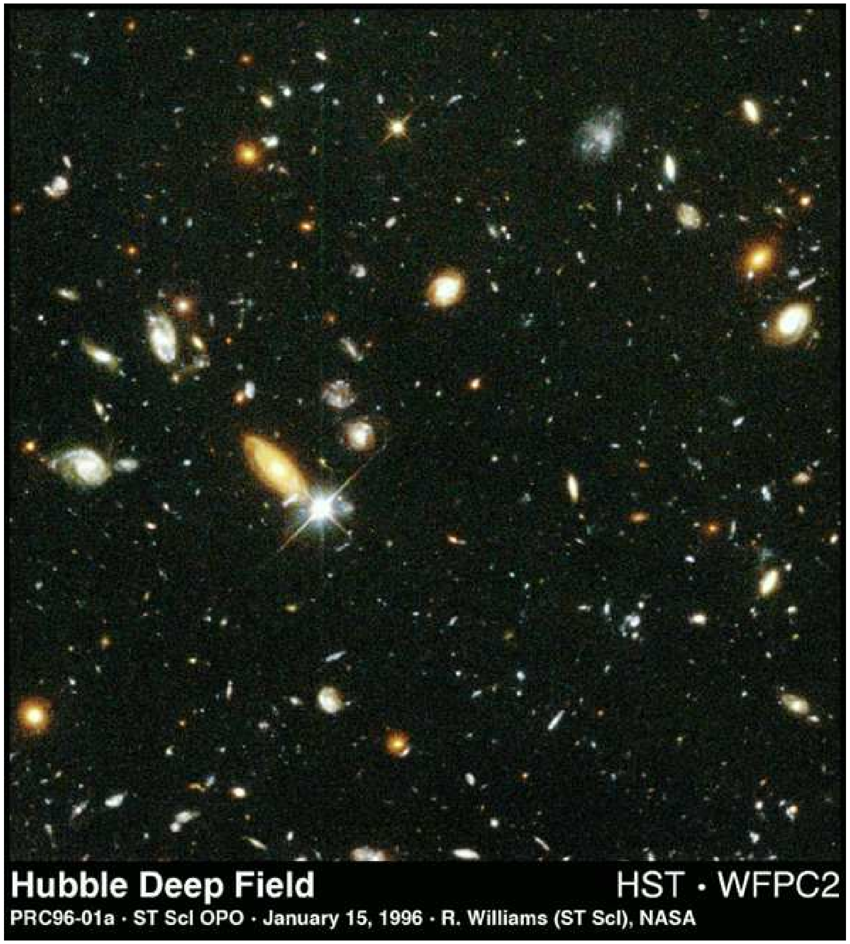
- **DFG SFB: No. 439/B11: 2005 - 2008**
- **Volkswagen/Baden-Württemberg, GRACE project: 2005 - 2008**

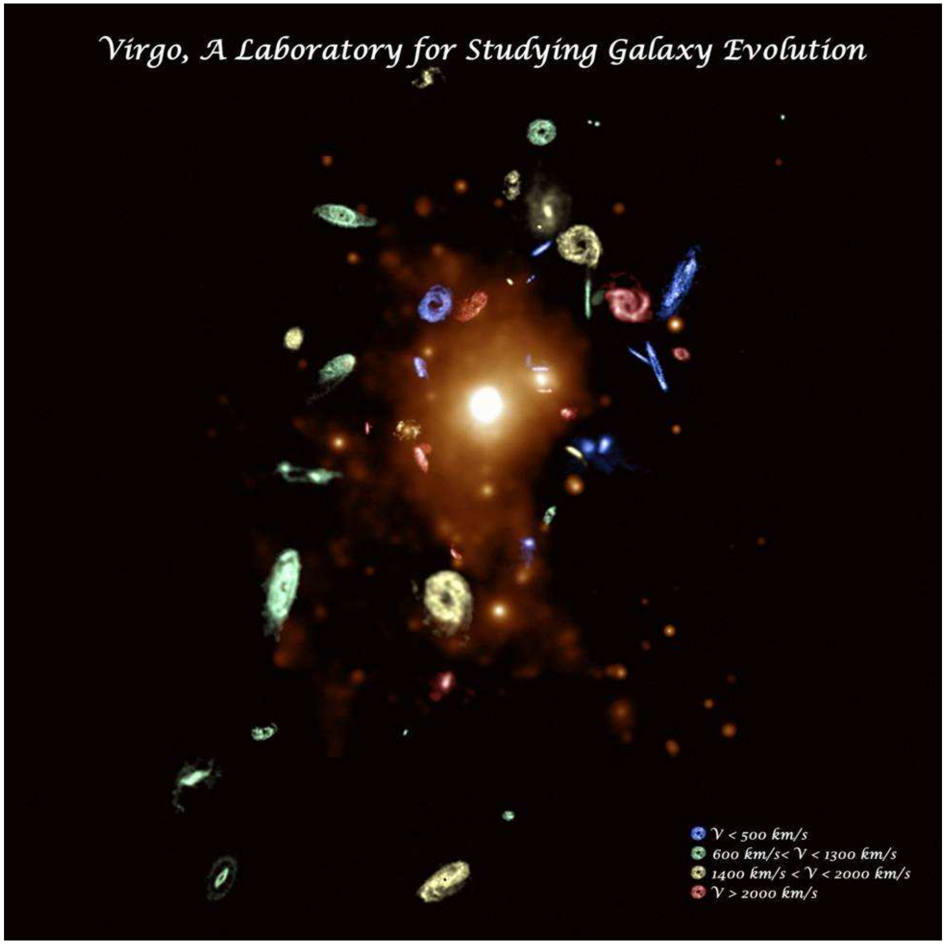
Formation of the Universe



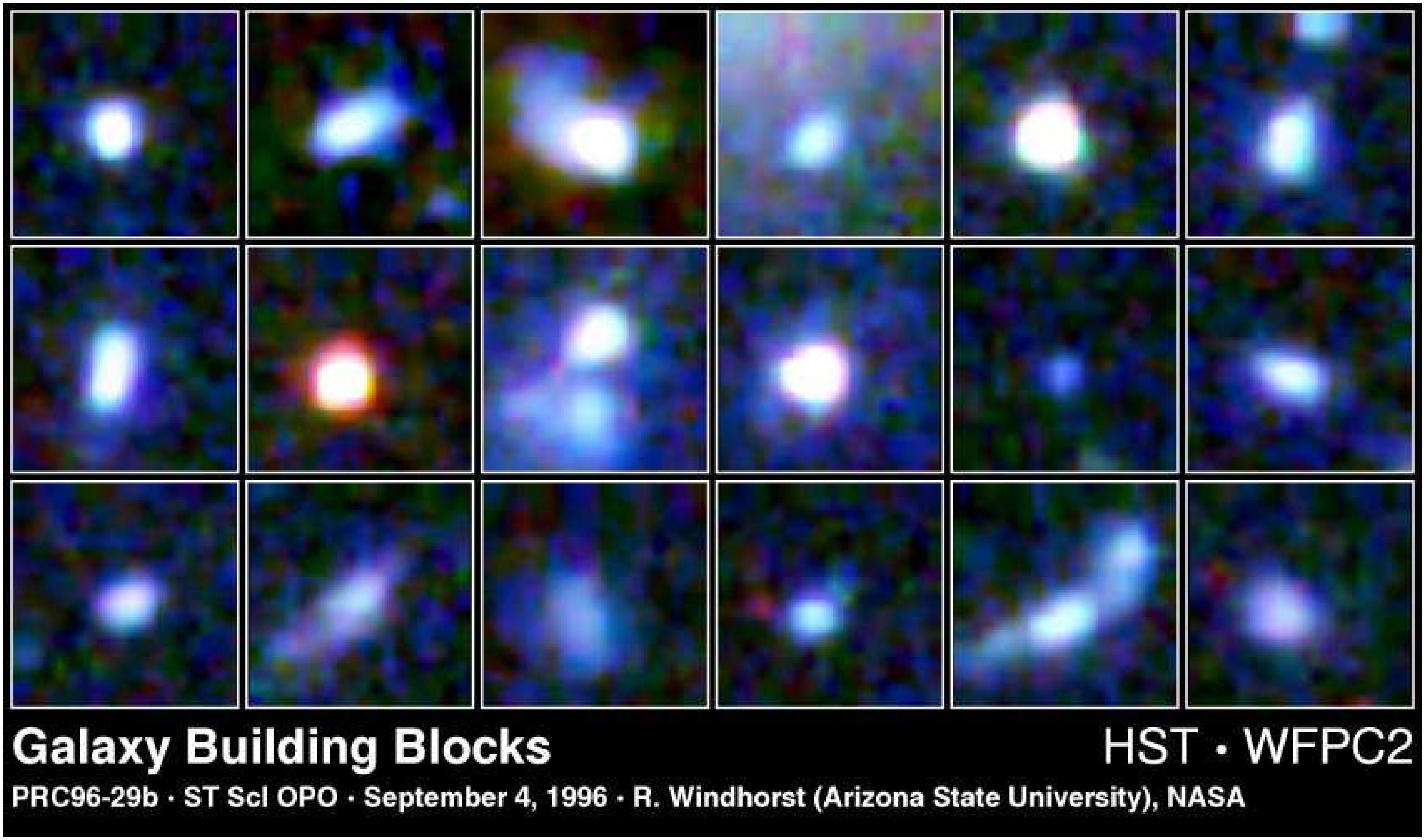
History of the Universe

Observations



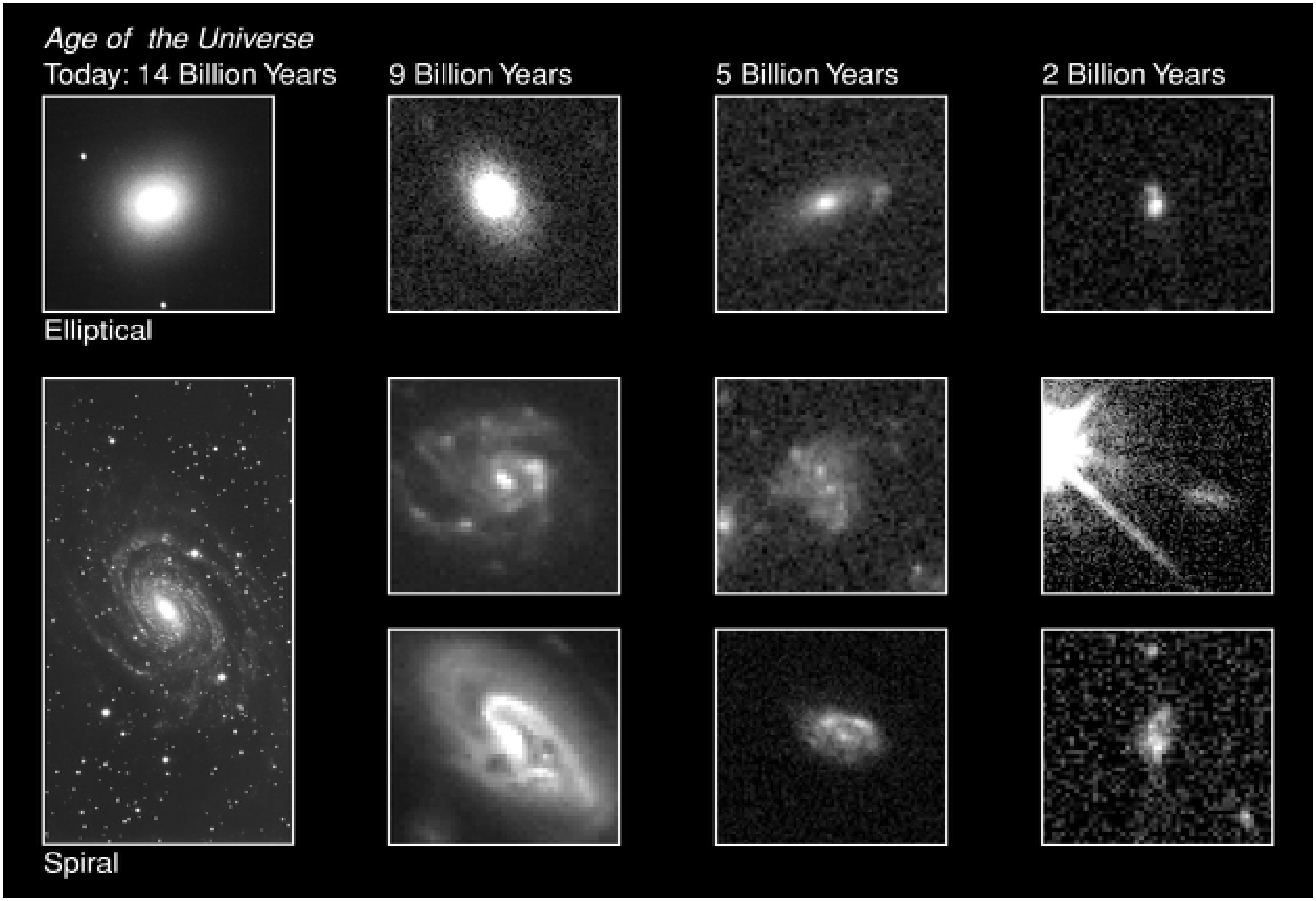


Observations



Observations

Observations

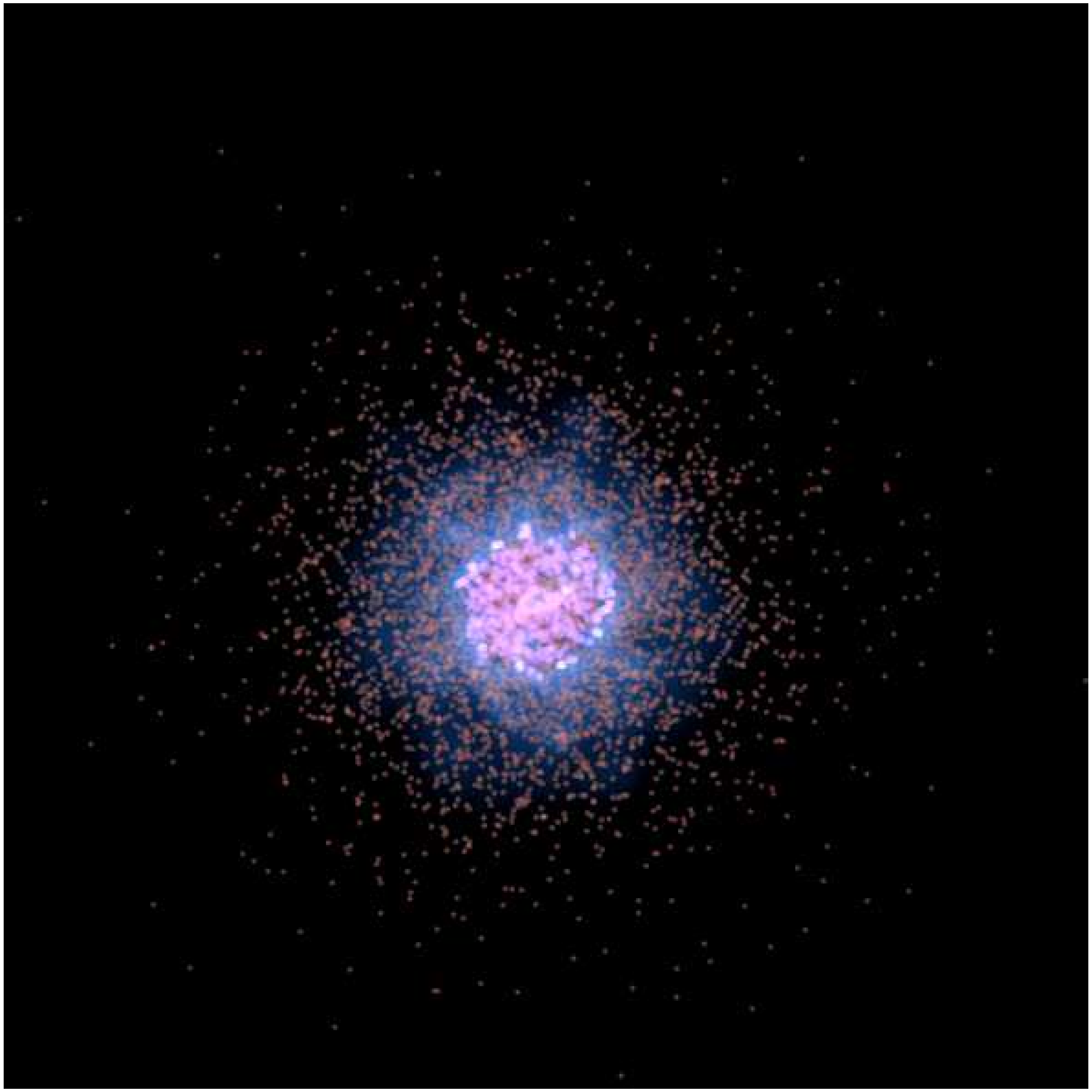


Isolated galaxy evolution

GASOLINE: Wadsley, Stadel & Quinn, 2003

http://www-hpcc.astro.washington.edu/

~few 10^6 SPH particles



Star Formation

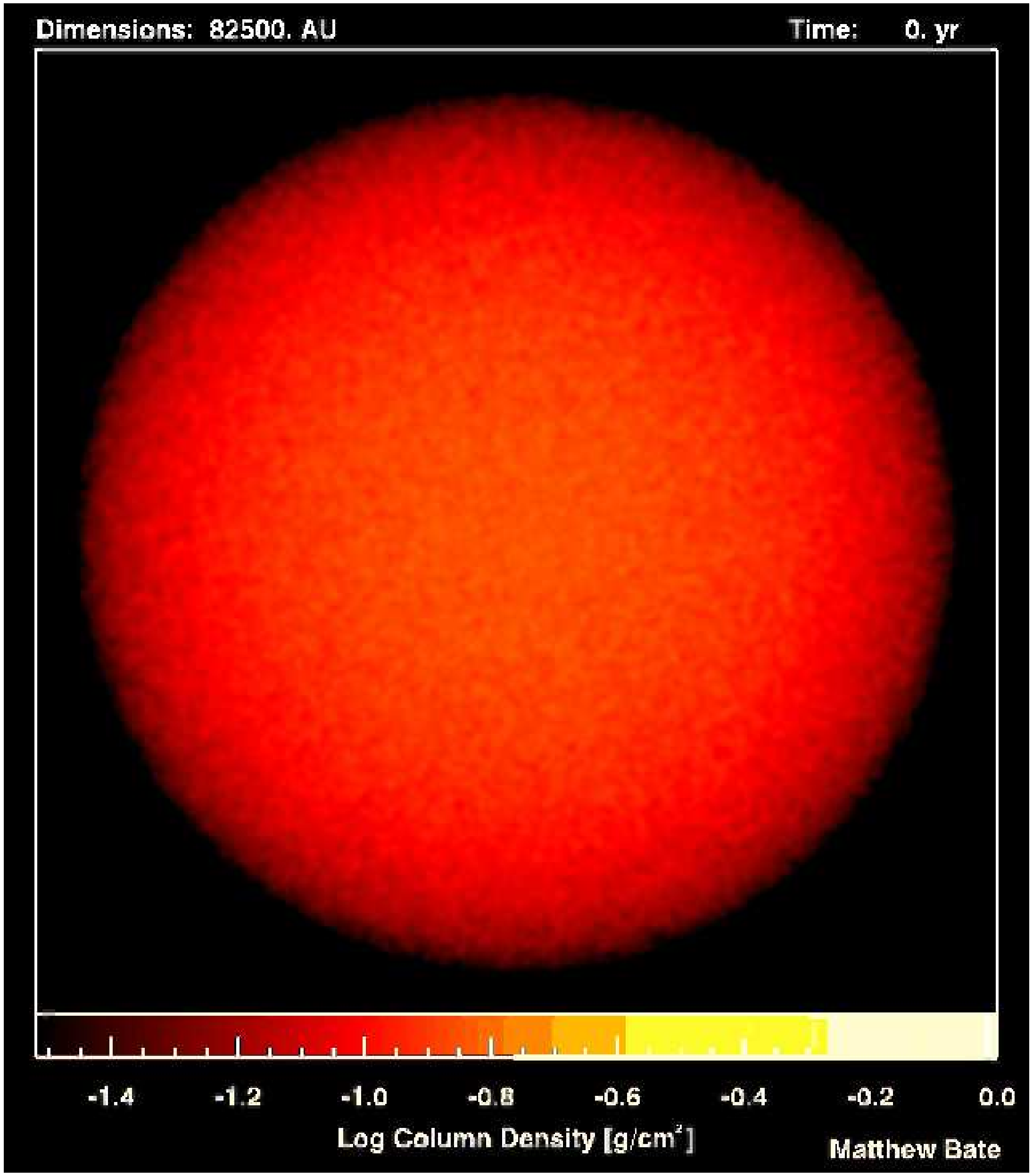
SPH: Benz, Bowers, Cameron & Press, 1990

OpenMP + Sink Particles: Bate, Bonnell & Price, 1995

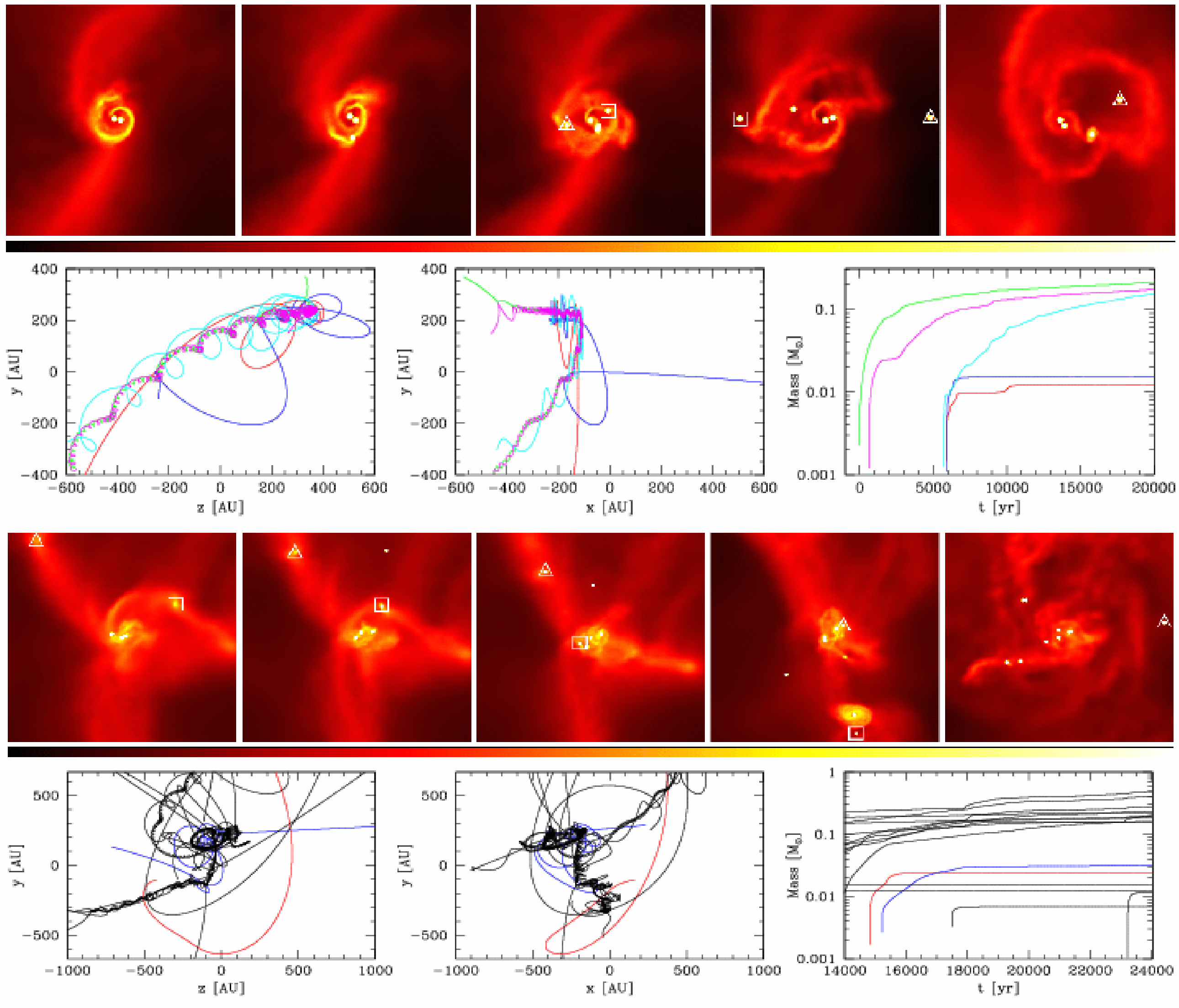
Bate, Bonnell, Bromm, 2002

The calculation required ~100k CPU hours (~11.4 years) on the SGI Origin 3800 (64 CPU) of the United Kingdom Astrophysical Fluids Facility (UKAFF).

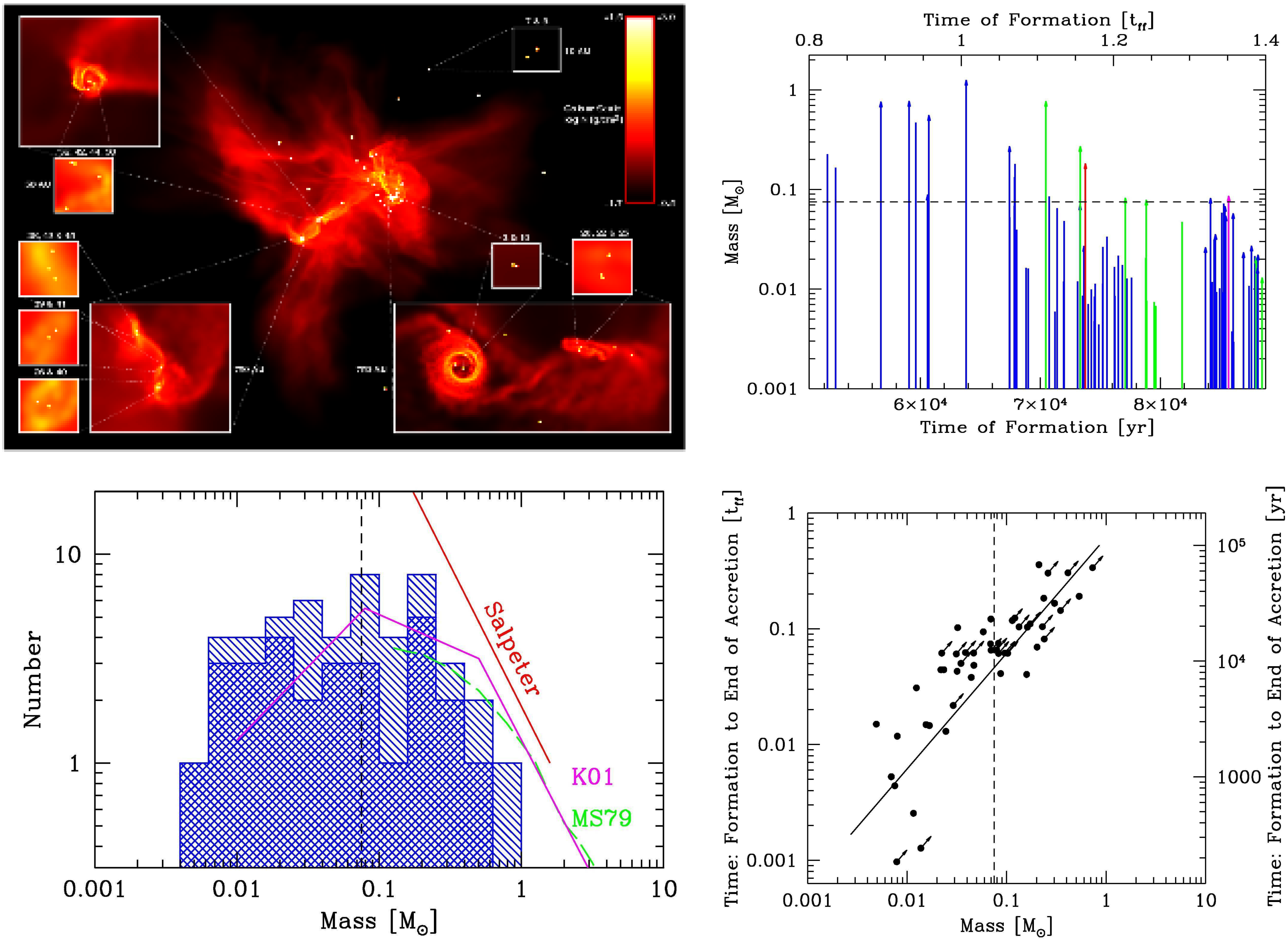
~few 10^6 SPH particles



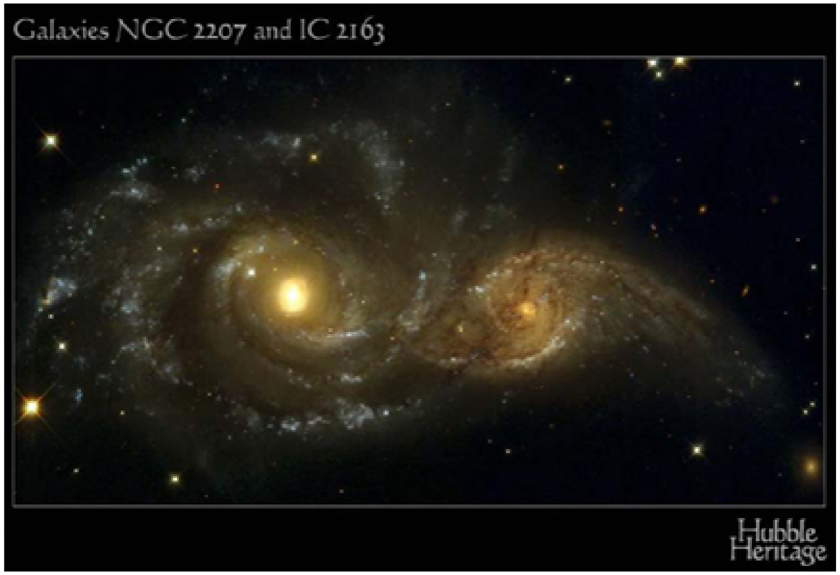
Star Formation

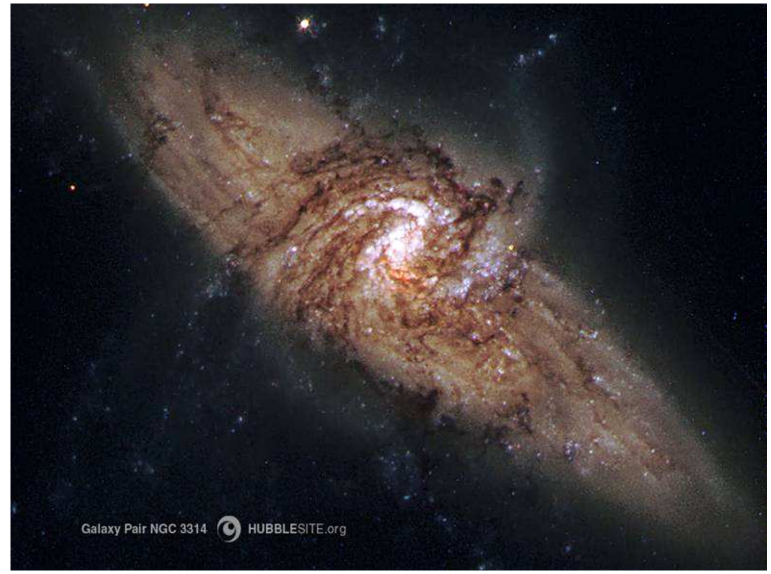


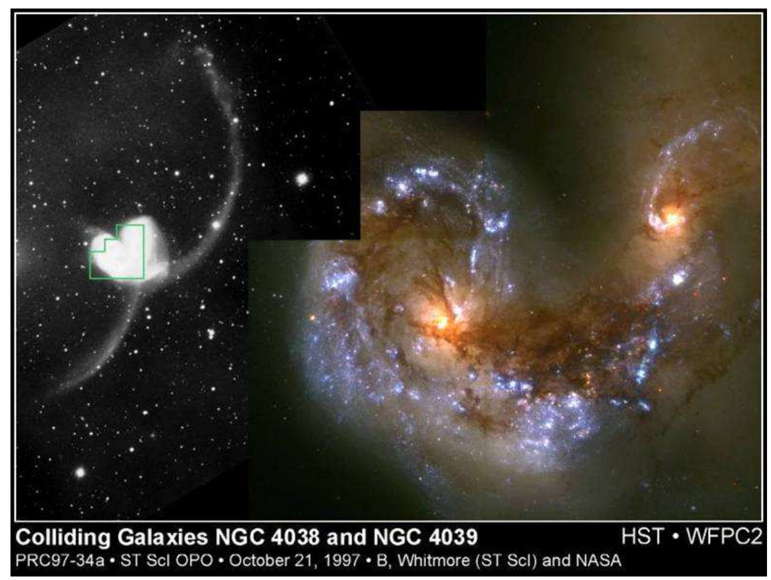
High mass stars can forms by gas (competitive) accretion!!!


Time of Formation [t$_{ff}$]
0.8
1
1.2
1.4
Mass [M$_\odot$]
1
0.1
0.01
0.001
6×10^4
7×10^4
8×10^4
Time of Formation [yr]
Number
10
1
0.001
0.01
0.1
1
10
Mass [M$_\odot$]
Salpeter
K01
MS79
Time: Formation to End of Accretion [t$_{ff}$]
1
0.1
0.01
0.001
10^5
10^4
1000
Time: Formation to End of Accretion [yr]
0.001
0.01
0.1
1
10
Mass [M$_\odot$]

Galaxy Collisions



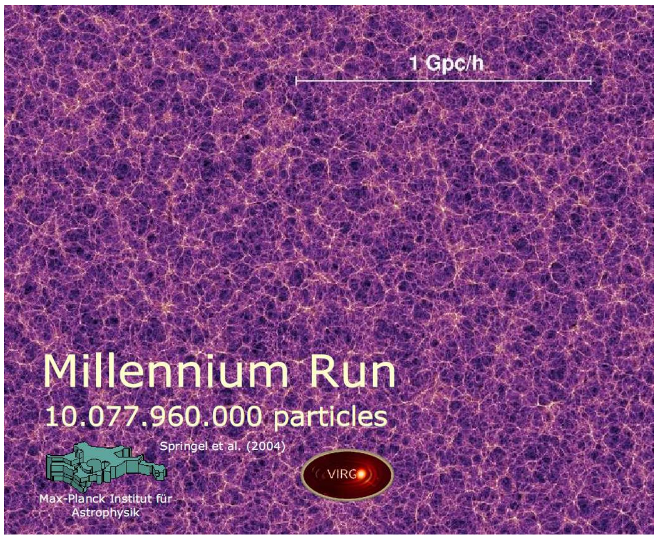




Galaxy Collisions

GADGET 2.0
Springel, 2005

http://www.mpa-garching.mpg.de/gadget/

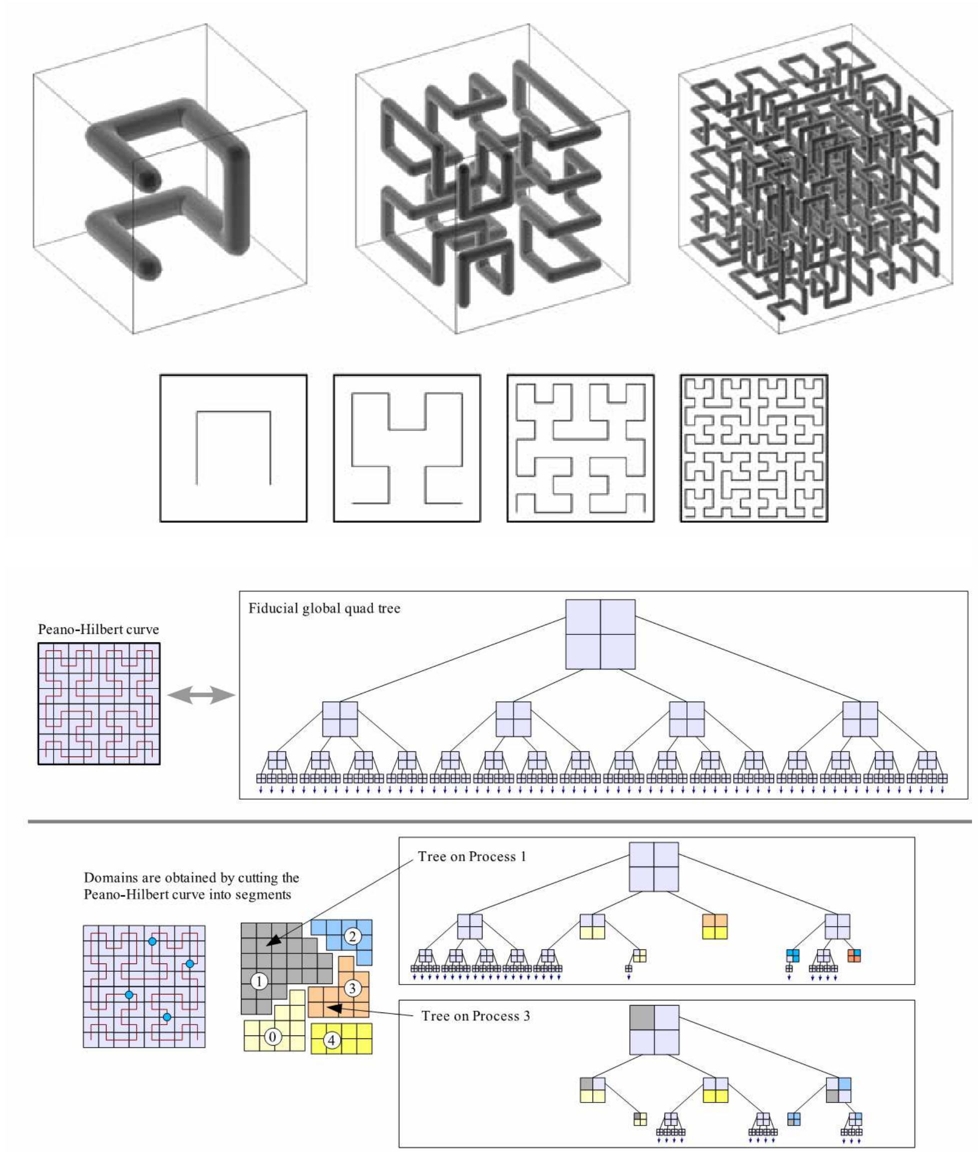


CPU time consumed

350.000 processor hours

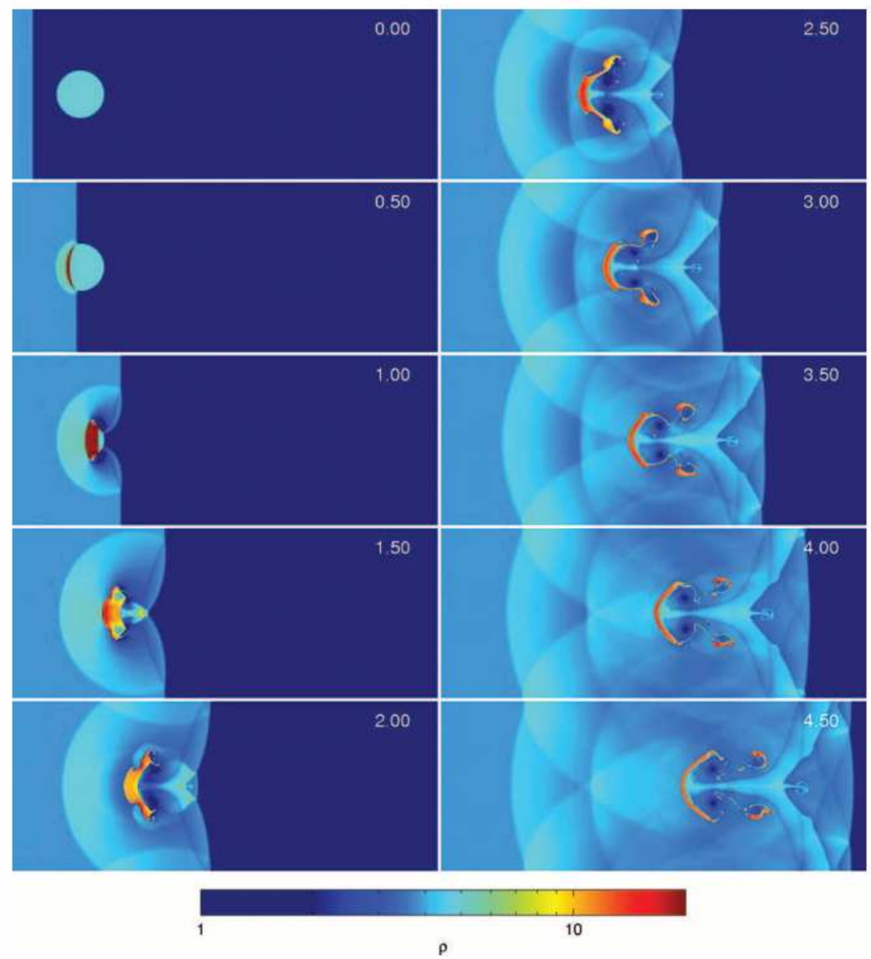
- 28 days on 512 CPUs/16 nodes
- 38 years in serial
- ~ 6% of annual time on total Regatta system
- sustained average code performance (hardware counters) 400 Mflops/cpu
- 5 x 10^{17} floating point ops
- 11000 (adaptive) timesteps

GADGET 2.0 details

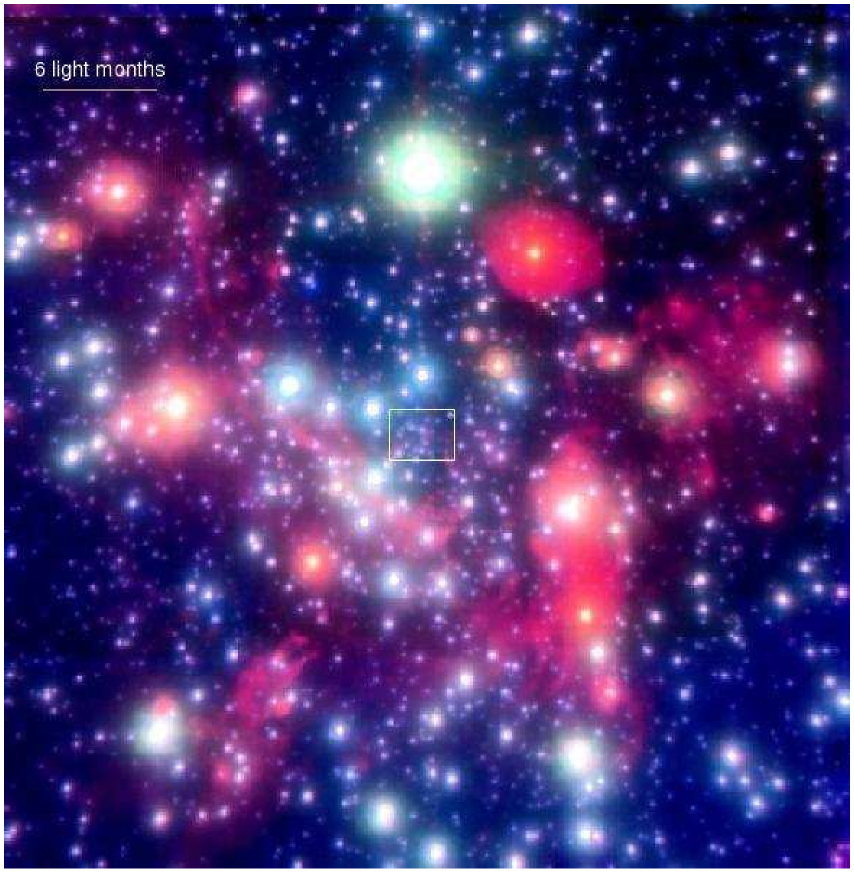


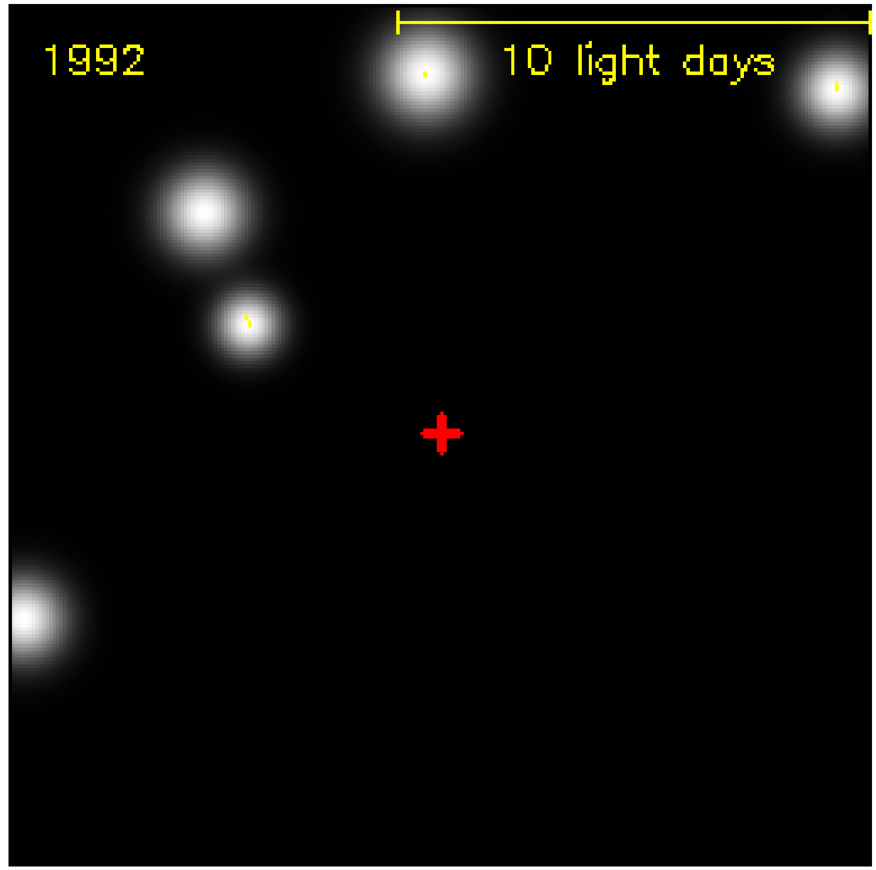
1 TByte RAM needed

16 x 32-way Regatta Node
64 GByte RAM
512 CPU total



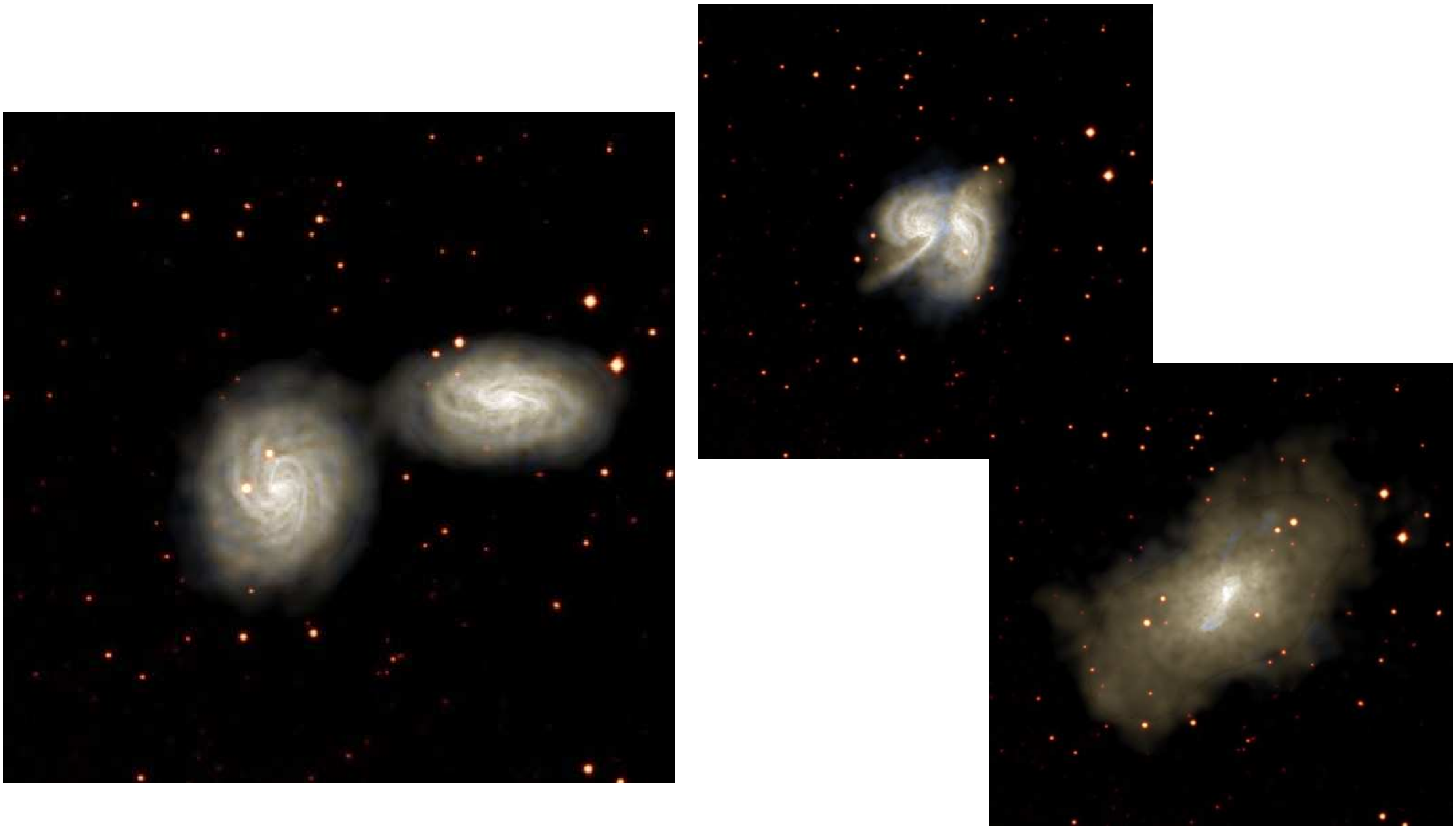
BH's in galaxies (MW - Sgr A*)





Galaxy Collisions ≈ BH's collisions

Mergers of Galaxies & MBH's [Begelman, Blandford & Rees, 90's]

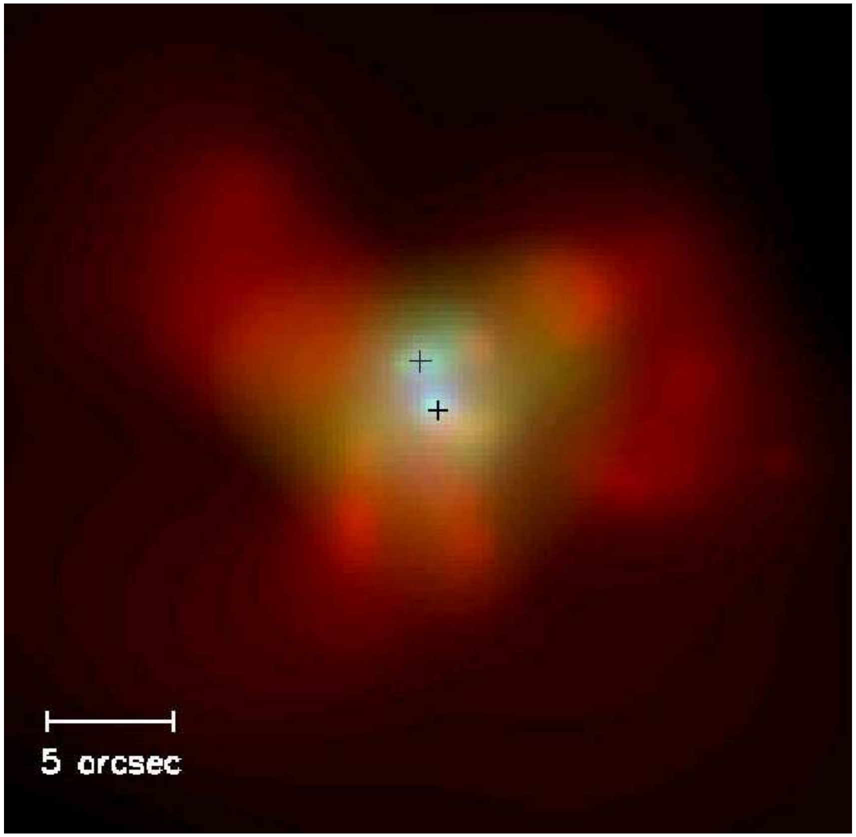


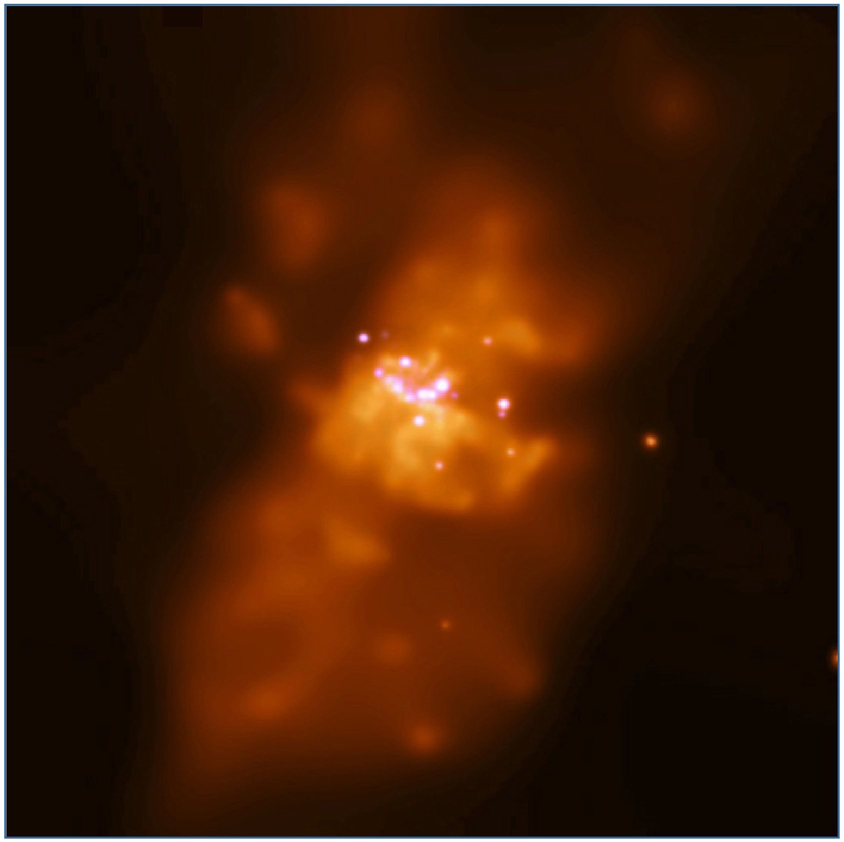
Galaxy Collisions ≈ BH's collisions

Multiple Massive Black Holes
NGC6240
strong ongoing merger…
Komossa et al. 2002

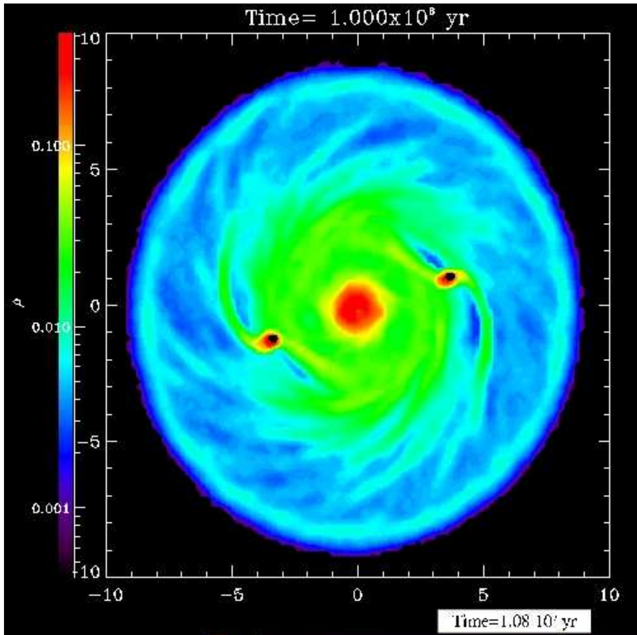
Two AGN in each of Nuclei separation
~1kpc Chandra X-Ray

M82: The bright spots in the center are supernova remnants and X-ray binaries. The luminosity of the X-ray binaries suggests that most contain a black hole. A close encounter with a large galaxy, M81, in the last 100 Myr is thought to be the cause of the starburst activity. Ebisuzaki et al. 2002

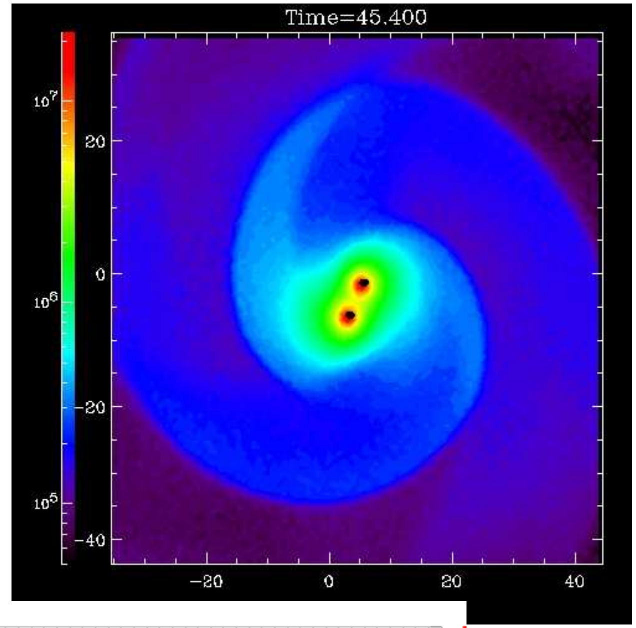


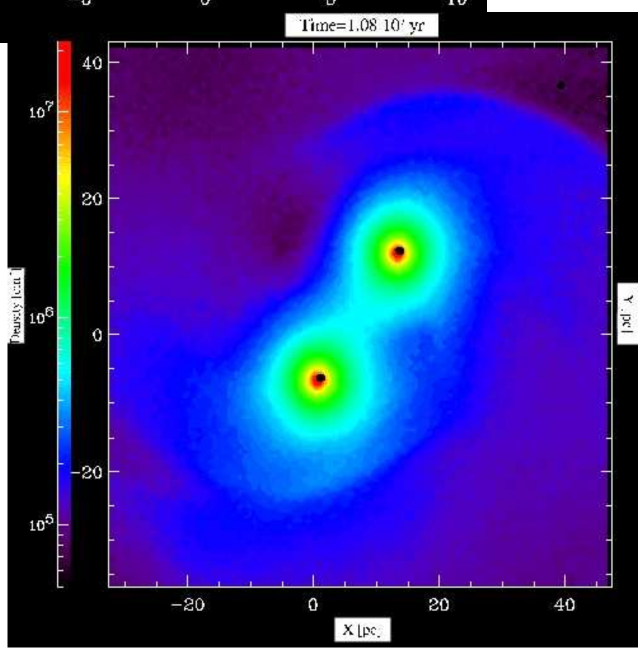


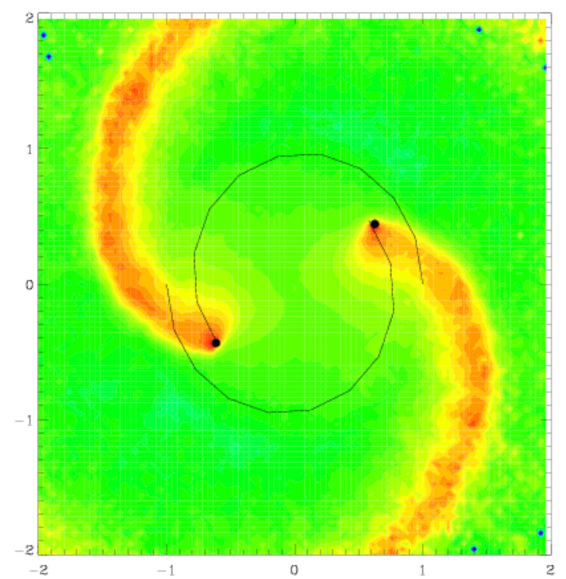
Galaxy Collisions ≈ BH’s collisions



GADGET 2.0 Simulations





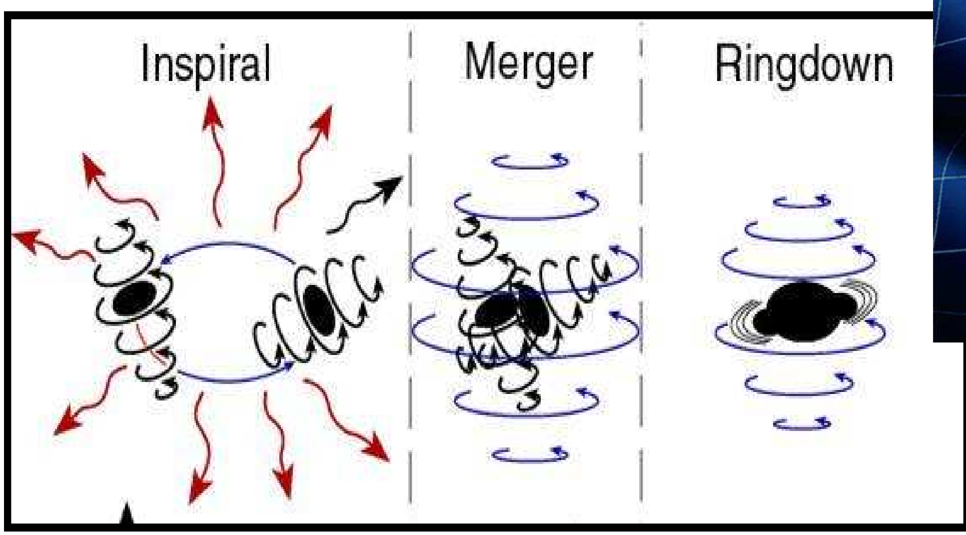


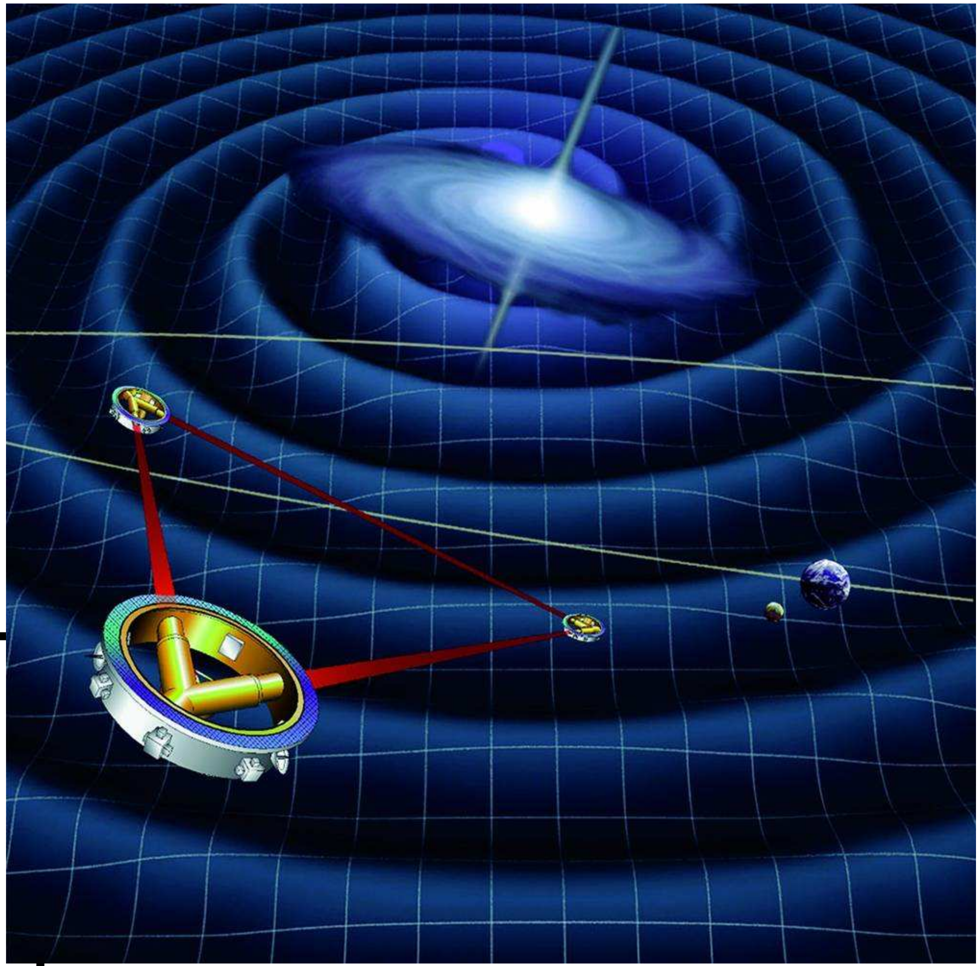
Future Observations

Gravitational Wave Detection - LISA

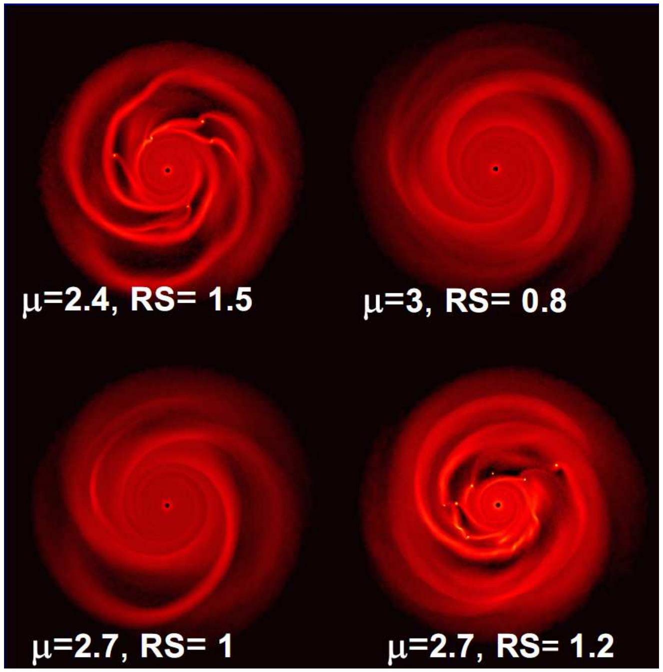
Two of the strongest potential sources in the low-frequency (LISA) regime are:

- Coalescence of binary supermassive black holes
- Extreme-mass-ratio inspiral into supermassive black holes



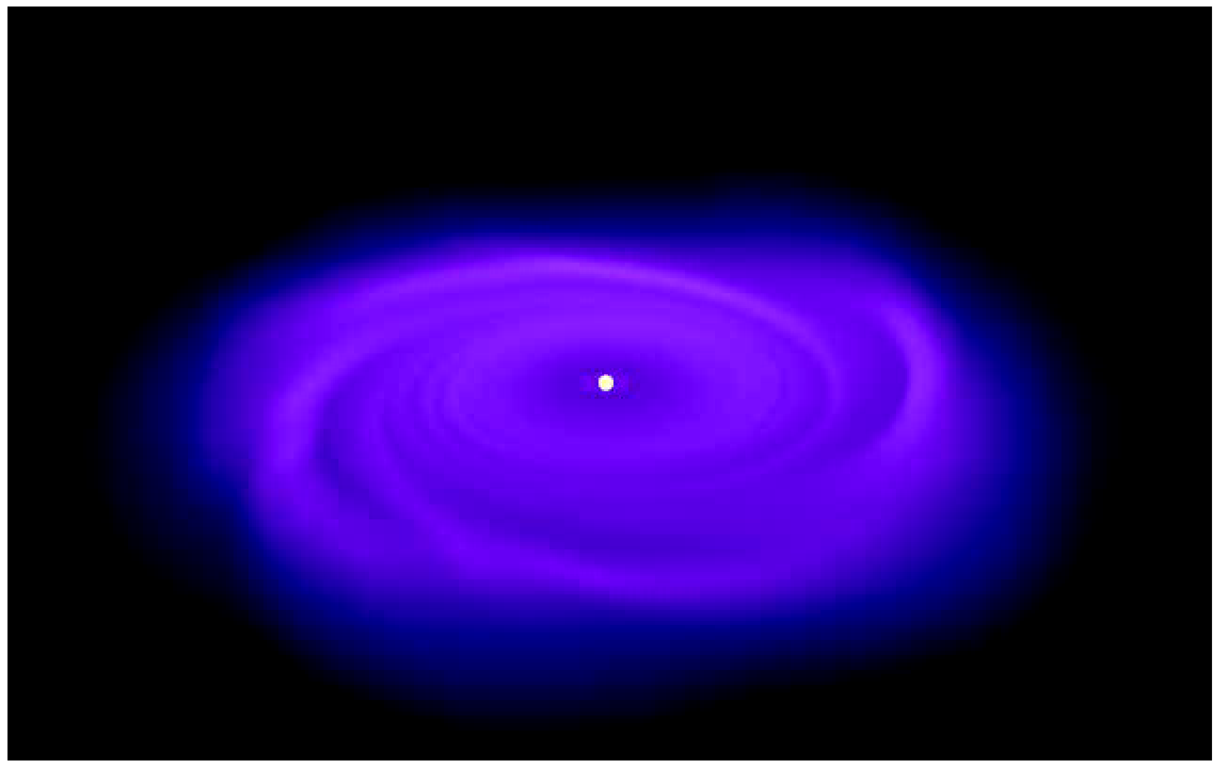


Proto-Planet formation

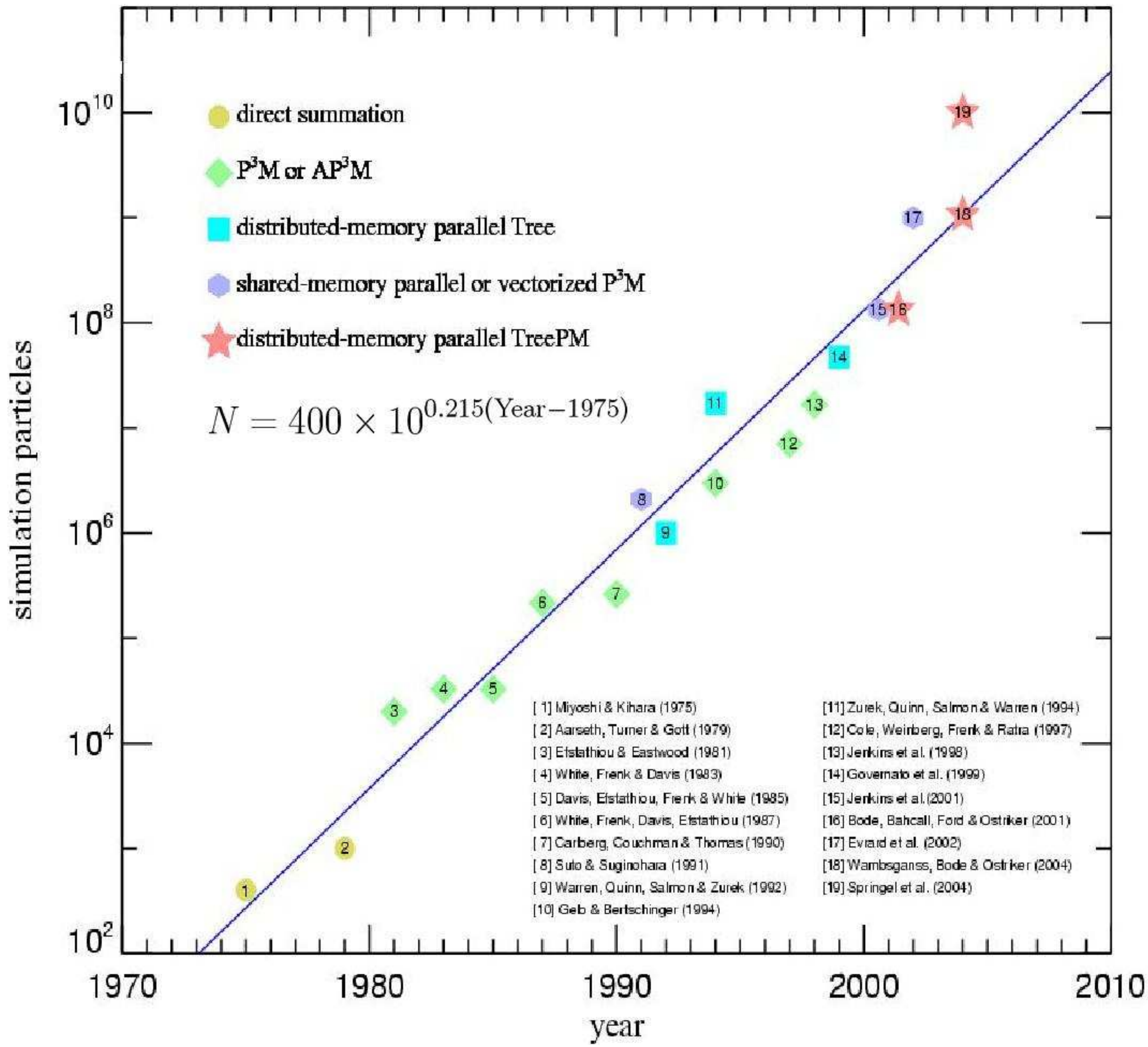


GASOLINE
Mayer, Lufkin et al. , 2006

Mayer et al. , 2002, 2003, 2004



Largest astrophysical N-body simulations



Father of numerical Astrophysics…
…with 200 light bulbs

LUMA METALL

Dissertation Univ. Lund (Schweden) 1937:
A study of double and multiple galaxies.
Galaxies often in groups and pairs.
Satellit galaxies distributed unevenly. [Holmberg-Effect]

The Astrophysical Journal, Nov. 1941

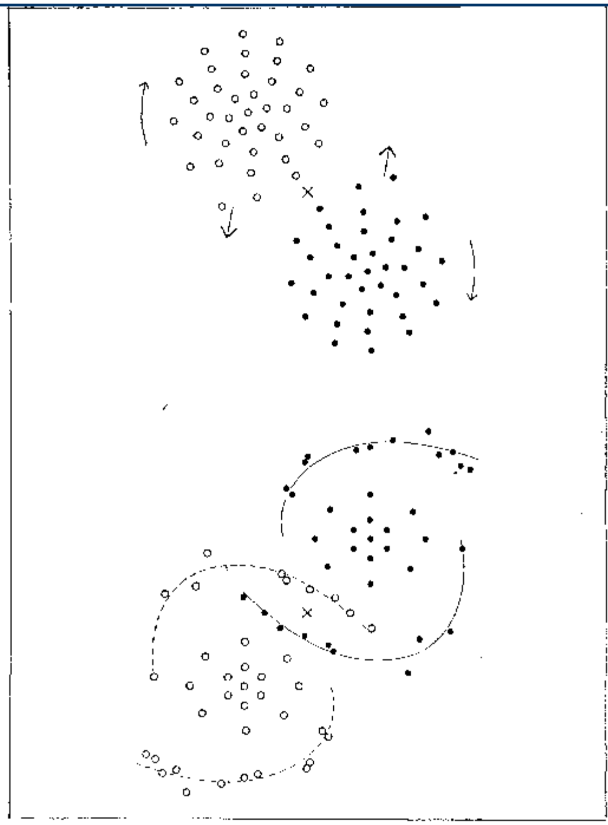


FIG. 4a

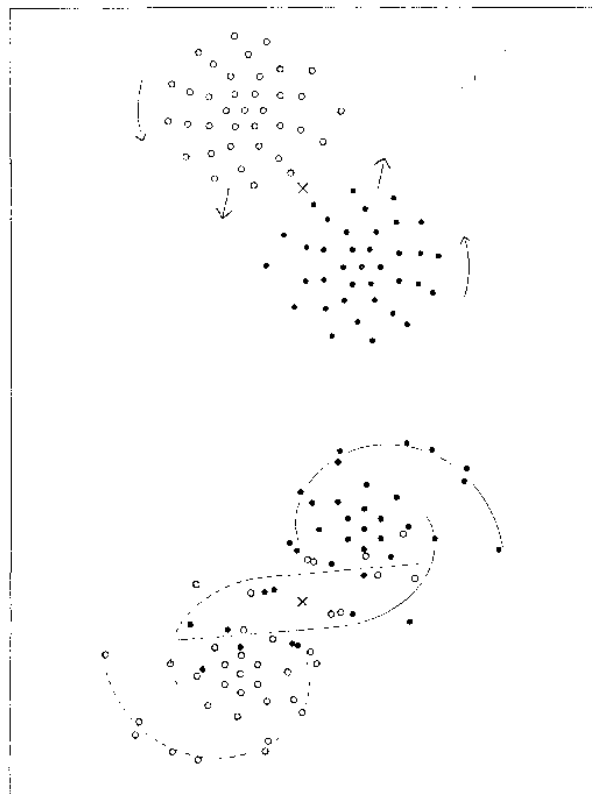


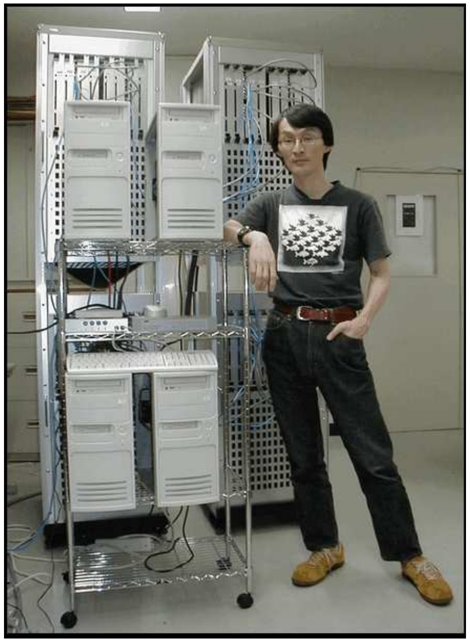
FIG. 4b

Erik Holmberg (1908-2000)

Nowadays real supercomputers...

ASCI-Q (LANL)
~30 Tflops
~250M USD

Earth Simulator
~40 Tflops
~350M USD



48 x GRAPE6
~48 Tflops
~3M USD

Makino, 2002

GRAPE Gordon Bell prizes

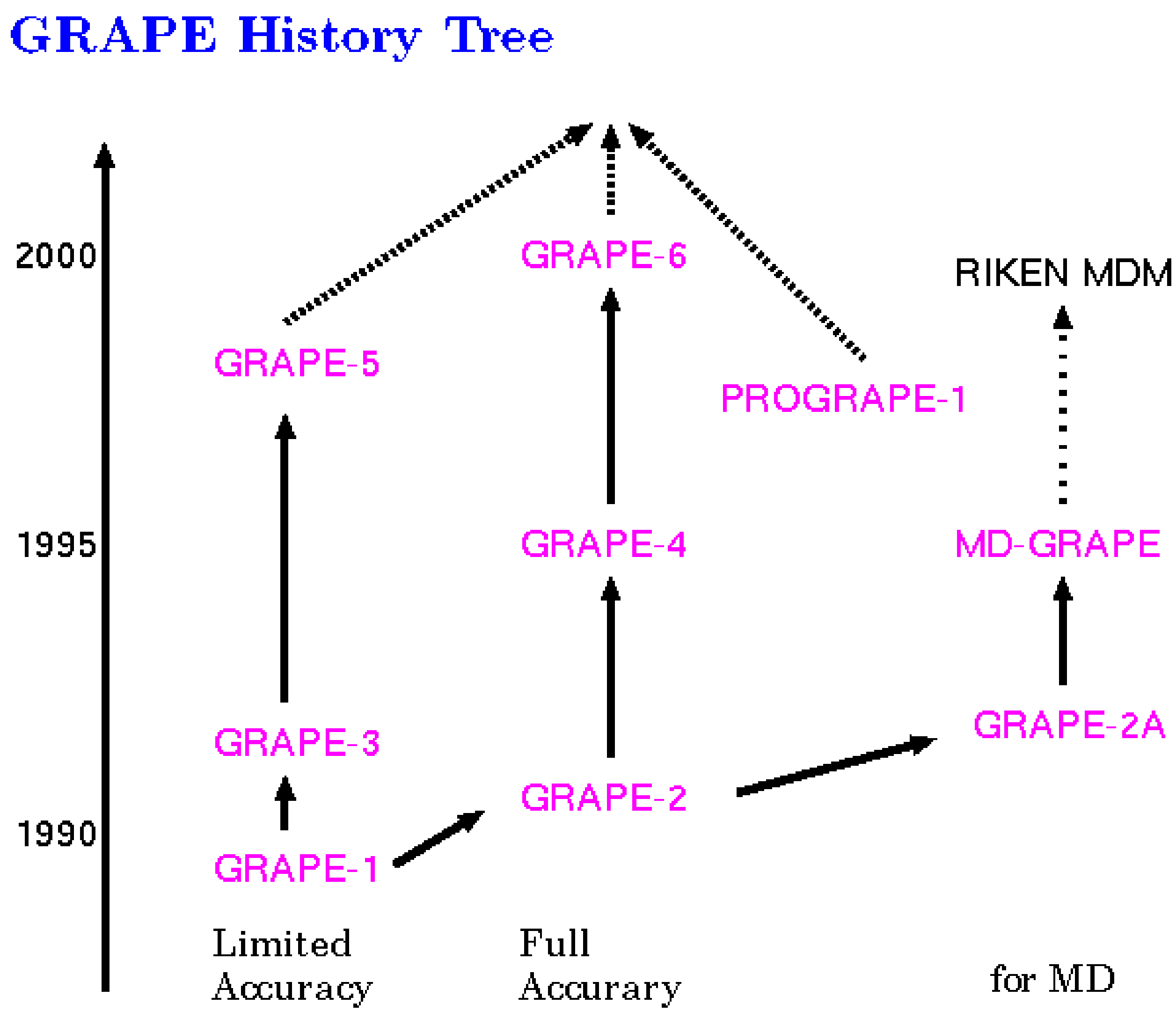
2001: 11.58Tflops (G6)

2000: 1.34Tflops (G6)

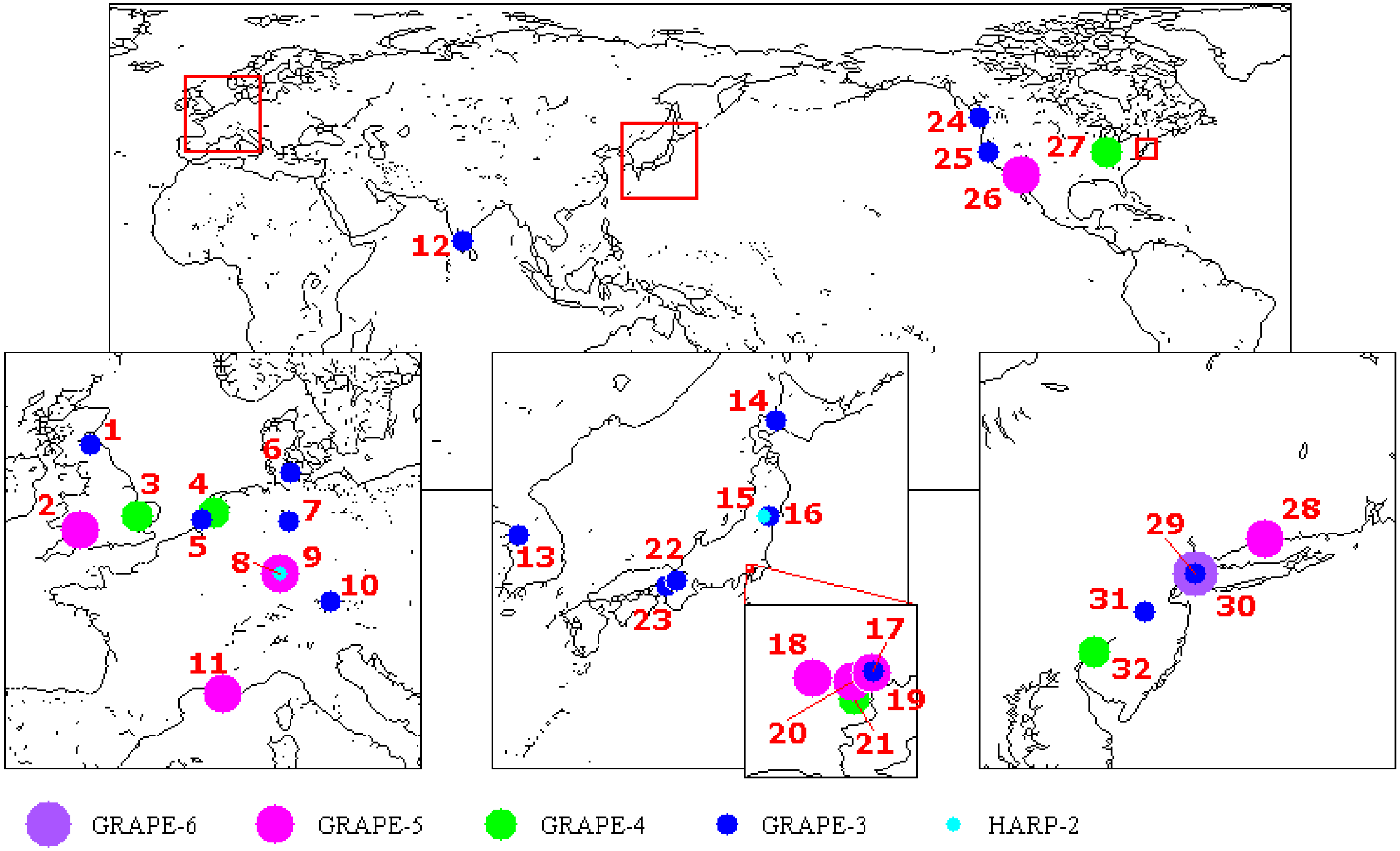
1999: $7/Mflops (G5)

1996: 333Gflops (G4)

GRAPE history tree



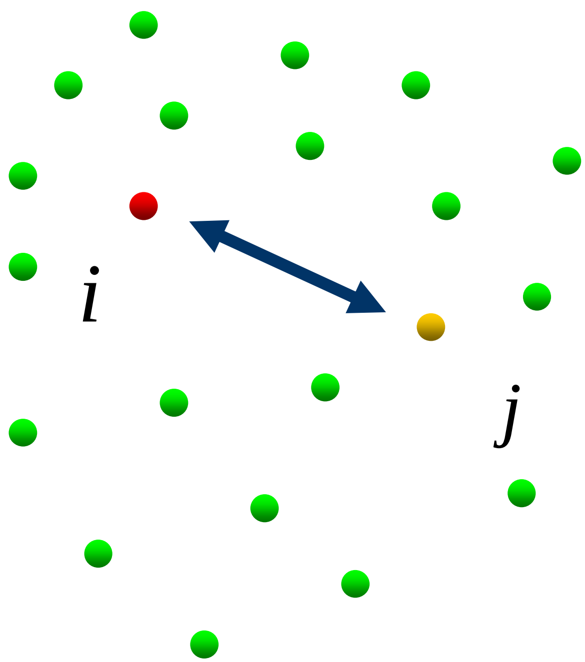
GRAPE's all over the World



* The size of the symbol indicates the top speed of the GRAPE installed.
* If an institute has different versions of GRAPEs, the latest version's symbol is shown.

http://www.astrogrape.org

Basic idea of any N-body code



m_j

t_i

$\vec{r}_i$

$\vec{v}_i$

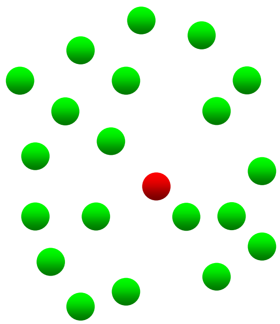
$$\frac{d^2\vec{r}_i}{dt^2} = \vec{a}_i$$

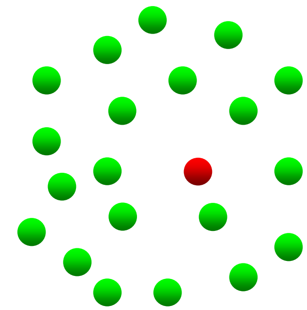
$$\propto N \cdot (N-1)$$

$$\vec{r}_{ij} = \vec{r}_i - \vec{r}_j$$

$$\vec{f}_{ij} = -\frac{G \cdot m_j}{(r_{ij}^2 + \varepsilon^2)^{3/2}} \vec{r}_{ij}$$

Basic idea of any N-body code





$$\vec{v}_i^{\,new} = \vec{v}_i^{\,old} + \vec{a}_i \cdot \frac{\Delta t}{2}$$

t^{old}

$$\vec{r}_i^{\,new} = \vec{r}_i^{\,old} + \vec{v}_i^{\,new} \cdot \Delta t$$

$t^{new} = t^{old} + \Delta t$

$\vec{r}_i^{\,old}$

$$\vec{a}_i = -\sum_{j=1;\ j\neq i}^{N} \frac{G \cdot m_j}{(r_{ij}^2 + \varepsilon^2)^{3/2}} \vec{r}_{ij}$$

$\vec{r}_i^{\,new}$

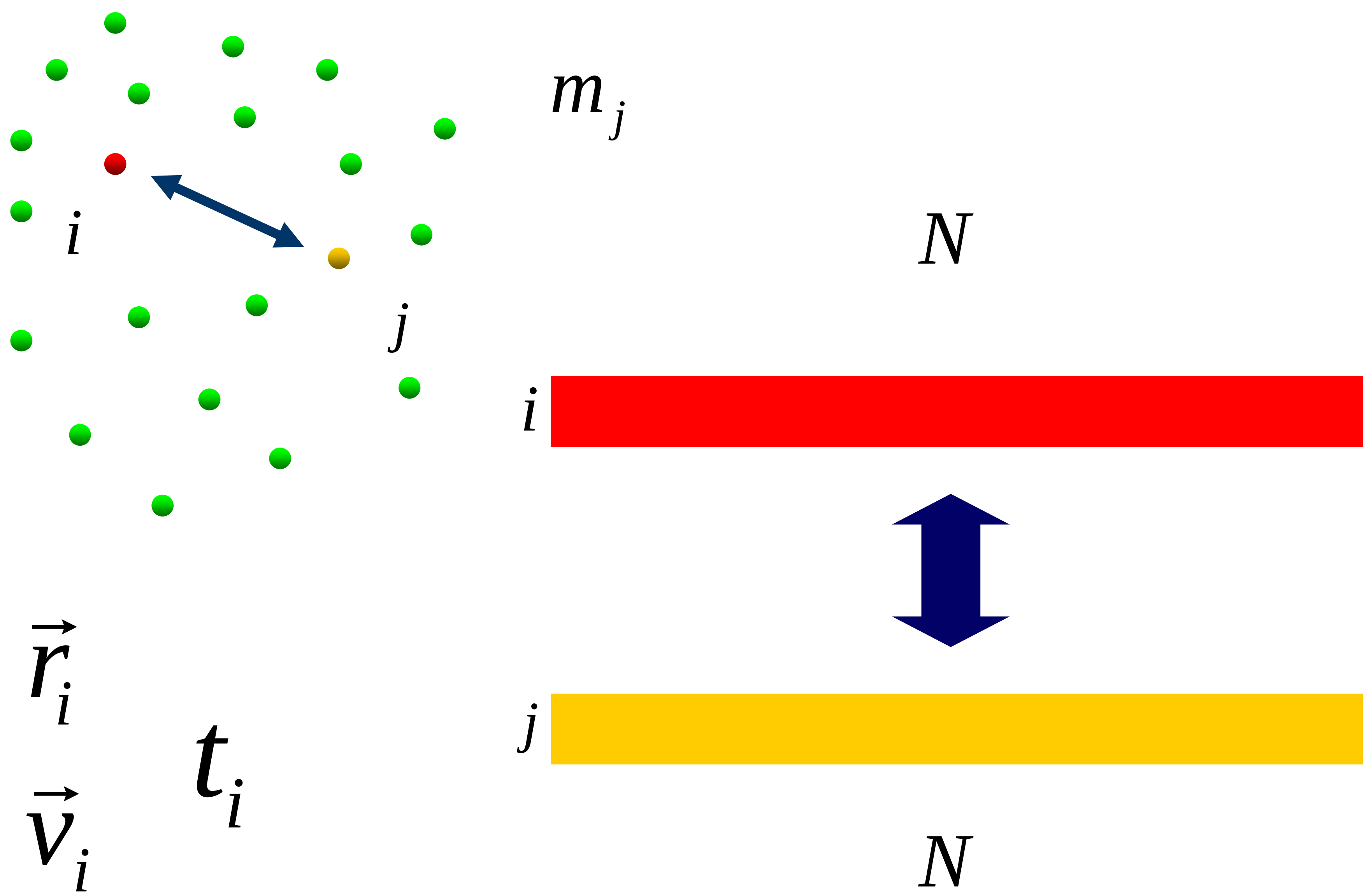
$\vec{v}_i^{\,old}$

$\vec{v}_i^{\,new}$

$$\vec{v}_i^{\,new} = \vec{v}_i^{\,new} + \vec{a}_i \cdot \frac{\Delta t}{2}$$

$\vec{r}_{ij} = \vec{r}_i - \vec{r}_j$

Basic idea of any N-body code

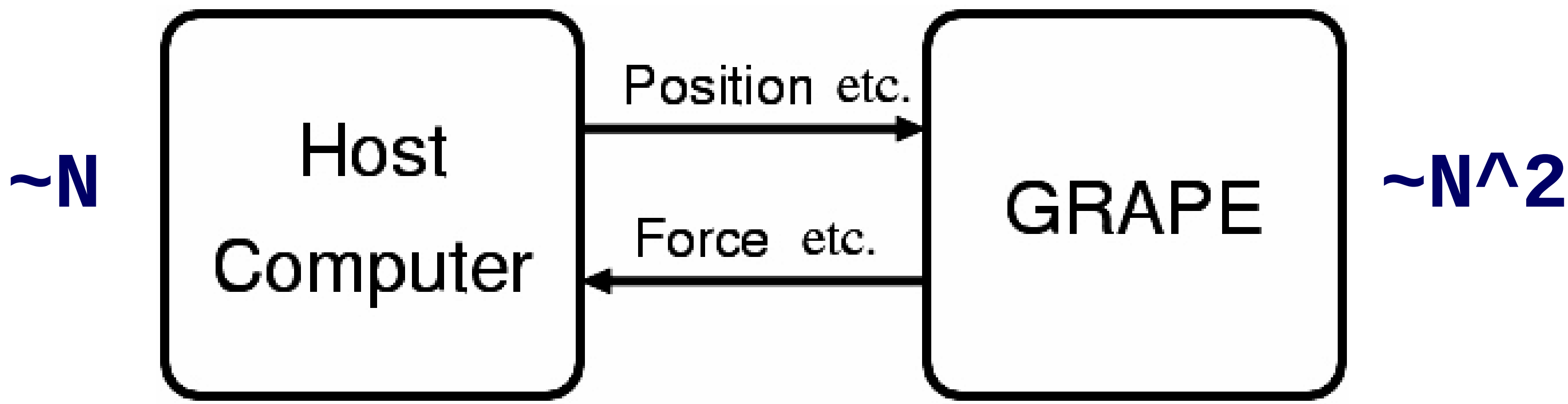


Basic idea of any N-body code

```
C       OBTAIN THE “FULL” FORCE FOR ALL BODY’S.
C       ……….
        DO 10 I = 1,N
          AX(I) = 0.0
          AY(I) = 0.0
          AZ(I) = 0.0
          DO 20 J = 1,N
            IF (J.EQ.I) GO TO 20
            DX_IJ = X(I) - X(J)
            DY_IJ = Y(I) - Y(J)
            DZ_IJ = Z(I) - Z(J)
            DR2 = DX_IJ*DX_IJ + DY_IJ*DY_IJ + DZ_IJ*DZ_IJ + EPS2
            TEMP = M(J)/( DR2*SQRT(DR2) )
            AX(I) = AX(I) – TEMP*DX_IJ
            AY(I) = AY(I) - TEMP*DY_IJ
            AZ(I) = AZ(I) - TEMP*DZ_IJ
20        CONTINUE
10      CONTINUE
C       ……….
```

$$\vec{a}_i = -\sum_{j=1;\, j\neq i}^{N} \frac{G \cdot m_j}{(r_{ij}^2 + \varepsilon^2)^{3/2}} \vec{r}_{ij}$$

Basic idea of any GRAPE N-body code:



$$\vec{a}_i = \sum_{j=1;\ j\neq i}^{N} \vec{f}_{ij} \qquad \vec{f}_{ij} = -\frac{G \cdot m_j}{(r_{ij}^2 + \varepsilon^2)^{3/2}} \vec{r}_{ij}$$

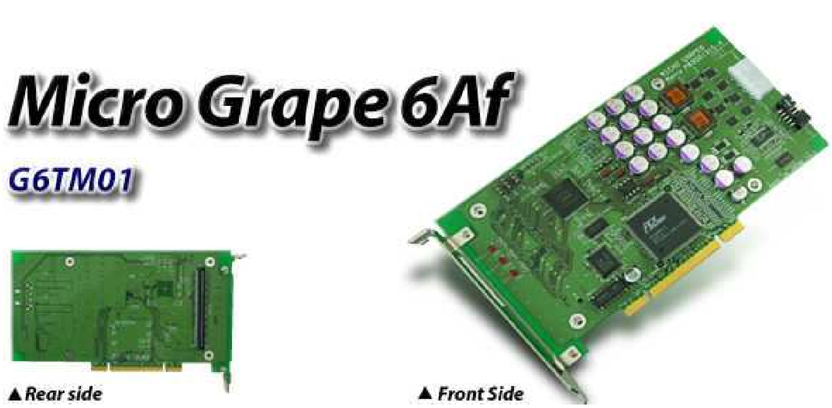
Commerce GRAPE6a boards

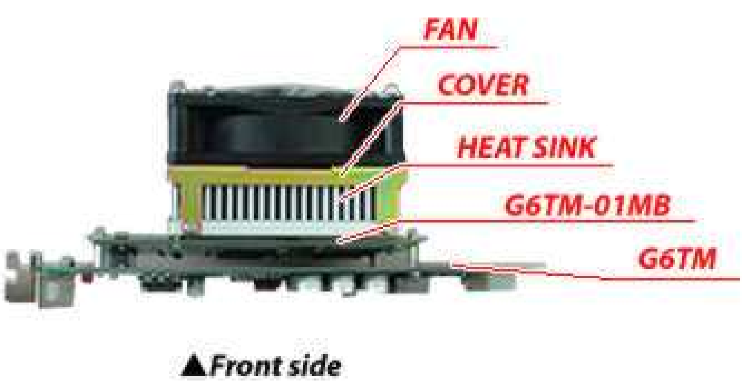
http://www.metrix.co.jp

Profile

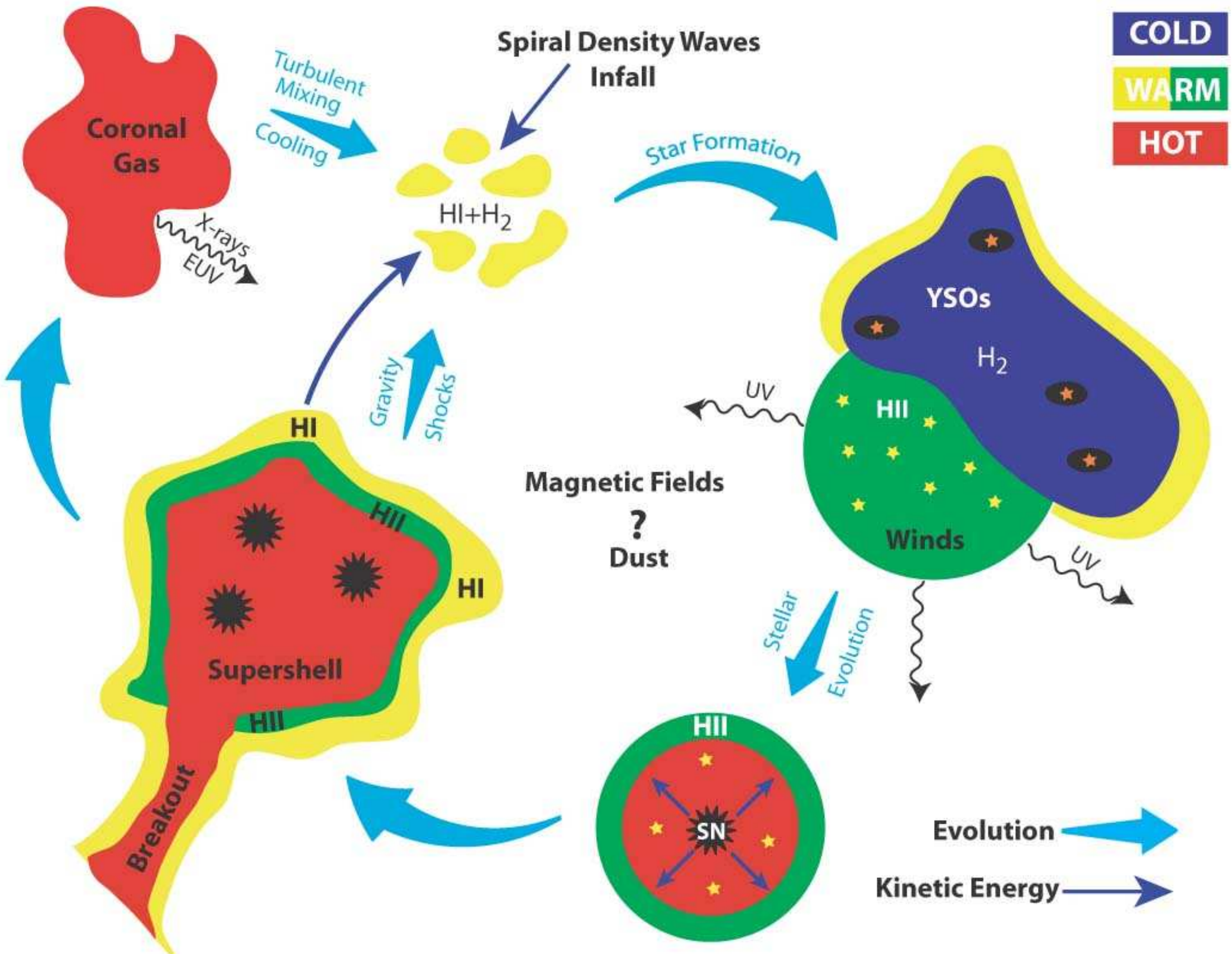
Hamamatsu Metrix

浜松メトリックス株式会社





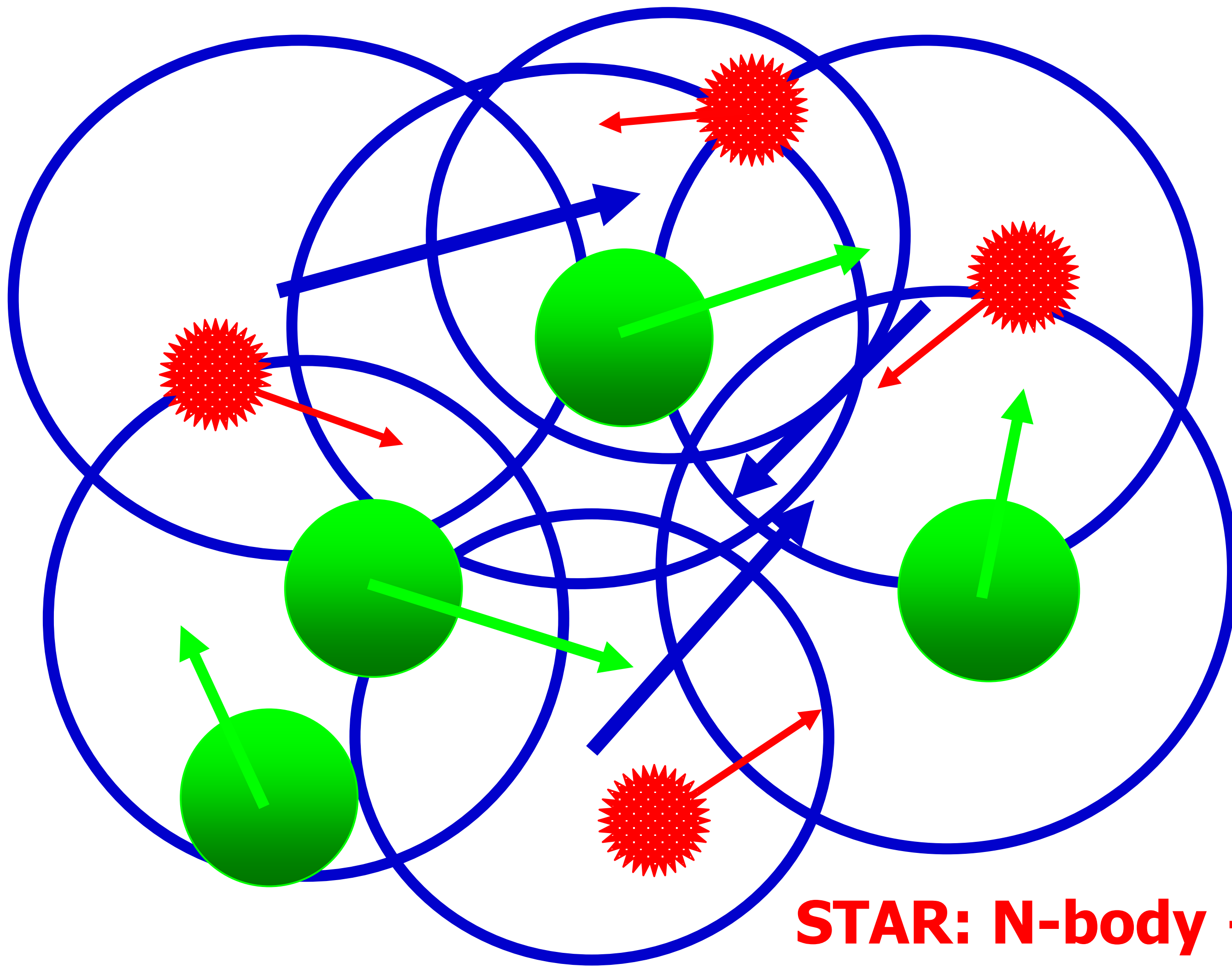
ISM "Ecology"



Tumlinson, 2004: astro-ph/0411249

Our Multi-Phase GRAPE SPH code

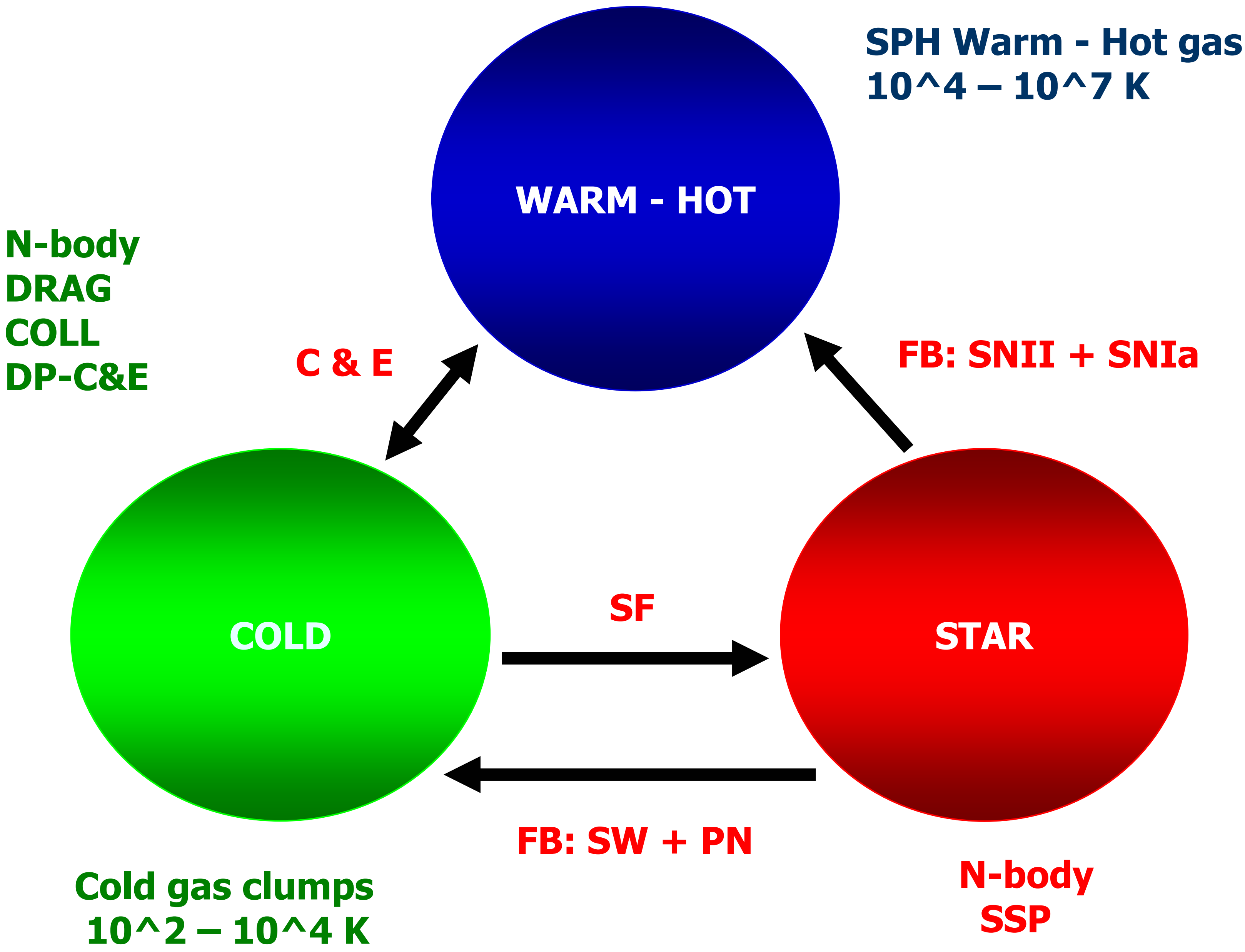
WARM - HOT: SPH



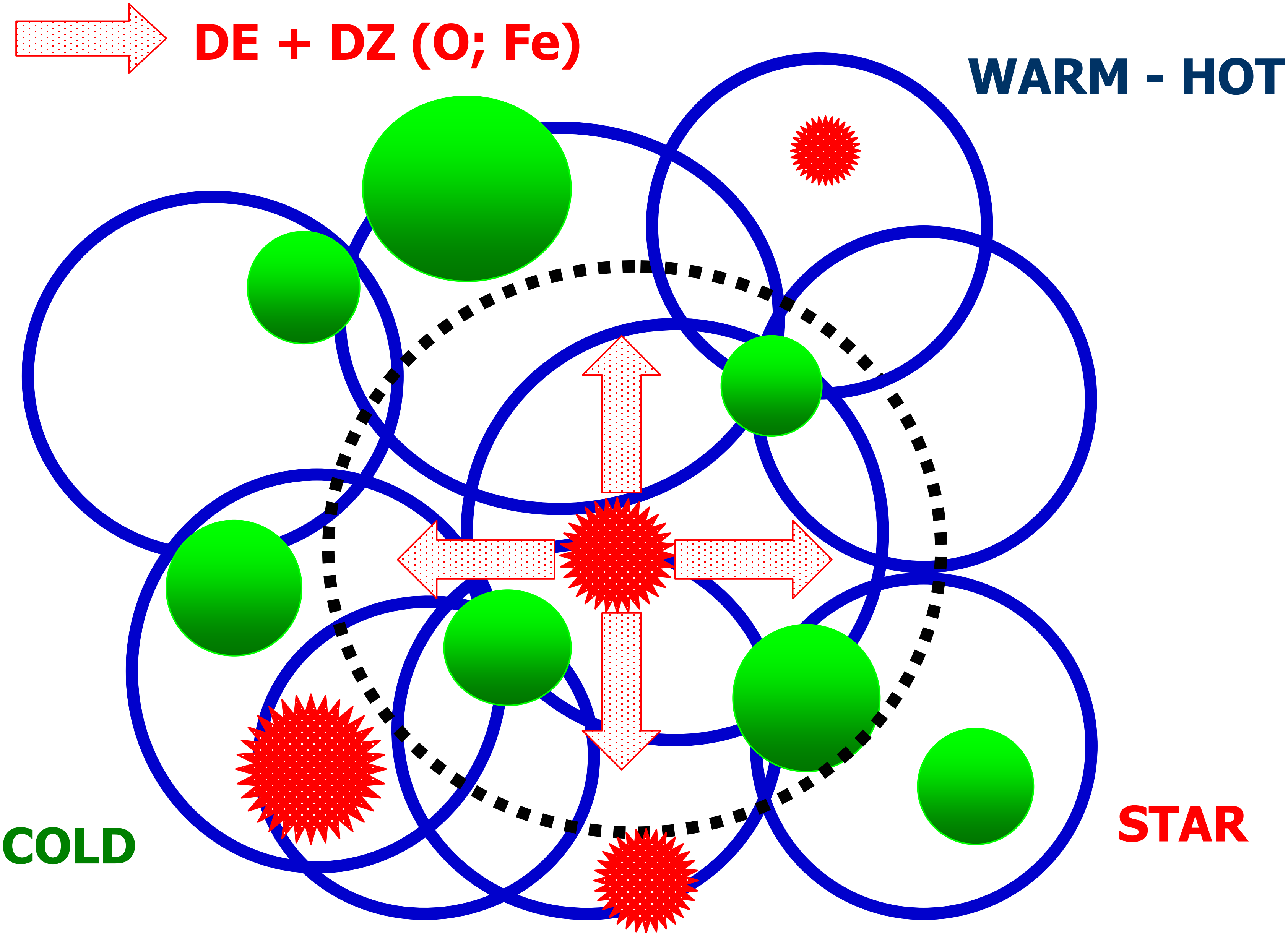
STAR: N-body + SSP

COLD: N-body + Viscosity

Our Multi-Phase GRAPE SPH code



Our Multi-Phase GRAPE SPH code



Basic Equations

Monaghan, 1977; Lucy, 1977

$$\rho_i \equiv \sum_{j=1}^{N} m_j \cdot W_{ij}$$

$$\frac{d\vec{v}_i}{dt} = -\sum_{j=1}^{N} m_j \cdot \left(\frac{P_i}{\rho_i^2} + \frac{P_j}{\rho_j^2} + \tilde{\Pi}_{ij} \right) \cdot \vec{\nabla}_i W_{ij} - \underline{\vec{\nabla}_i \Phi_i} - \underline{\vec{\nabla}_i \Phi_i^{ext}}$$

$$\frac{du_i}{dt} = \frac{1}{2}\sum_{j=1}^{N} m_j \cdot \left(\frac{P_i}{\rho_i^2} + \frac{P_j}{\rho_j^2} + \tilde{\Pi}_{ij} \right) \cdot \left(\vec{v}_i - \vec{v}_j\right) \cdot \vec{\nabla}_i W_{ij} + \underline{\frac{\Gamma_i - \Lambda_i}{\rho_i}}$$

$$P_i = (\gamma - 1) \cdot \rho_i \cdot u_i$$

Basic Equations

Monaghan & Gingold, 1983

$$\Pi_{ij} = \begin{cases} \left[-\alpha \cdot c_{ij} \cdot \mu_{ij} + \beta \cdot \mu_{ij}^2\right] / \rho_{ij} & \text{if } (\vec{r}_{ij} \cdot \vec{v}_{ij}) < 0 \\ 0 & \text{else} \end{cases}$$

$$\mu_{ij} = \frac{h_{ij} \cdot (\vec{v}_i - \vec{v}_j) \cdot (\vec{r}_i - \vec{r}_j)}{\left|\vec{r}_i - \vec{r}_j\right|^2 + \varepsilon \cdot h_{ij}^2}$$

$\alpha = 1$

$\beta = 2$

$\varepsilon = 0.01$

$$\rho_{ij} = \frac{1}{2}\left(\rho_i + \rho_j\right) \quad h_{ij} = \frac{1}{2}\left(h_i + h_j\right) \quad c_{ij} = \frac{1}{2}\left(c_i + c_j\right)$$

Basic Equations

Monaghan & Lattanzio, 1985

$$W(r;h) = \frac{1}{\pi \cdot h^3} \cdot \begin{cases} 1 - \frac{3}{2}\left(r/h\right)^2 + \frac{3}{4}\left(r/h\right)^3 & 0 \le r/h < 1 \\ \frac{1}{4}\left(2 - r/h\right)^3 & 1 \le r/h < 2 \\ 0 & 2 \le r/h \end{cases}$$

Hernquist & Katz, 1989

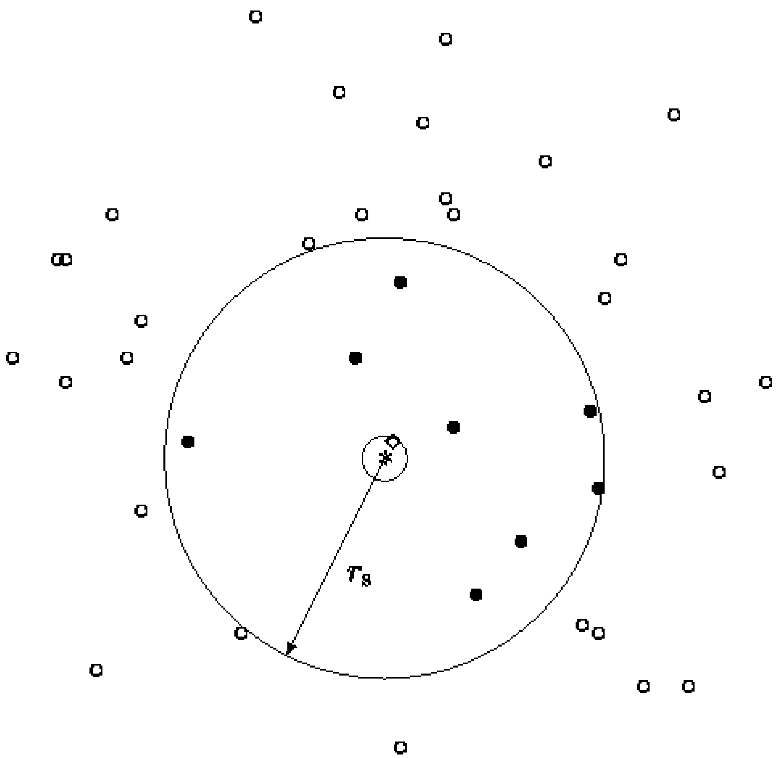
$$W_{ij} = W\left(\left|\vec{r}_i - \vec{r}_j\right|; h_{ij}\right)$$

Balsara, 1995; Steinmetz, 1996

$$\tilde{\Pi}_{ij} = \frac{1}{2}\left(f_i + f_j\right) \cdot \Pi_{ij}$$

$$f_i = \frac{\left|\left(\vec{\nabla} \cdot \vec{v}\right)_i\right|}{\left|\left(\vec{\nabla} \cdot \vec{v}\right)_i\right| + \left|\left(\vec{\nabla} \times \vec{v}\right)_i\right| + \varepsilon \cdot c_i / h_i}$$

Define smoothing length



$$\left|r_{ij}\right| \leq 2 \cdot \max\left(h_i; h_j\right)$$

Inside "2•h" N_B = const =50

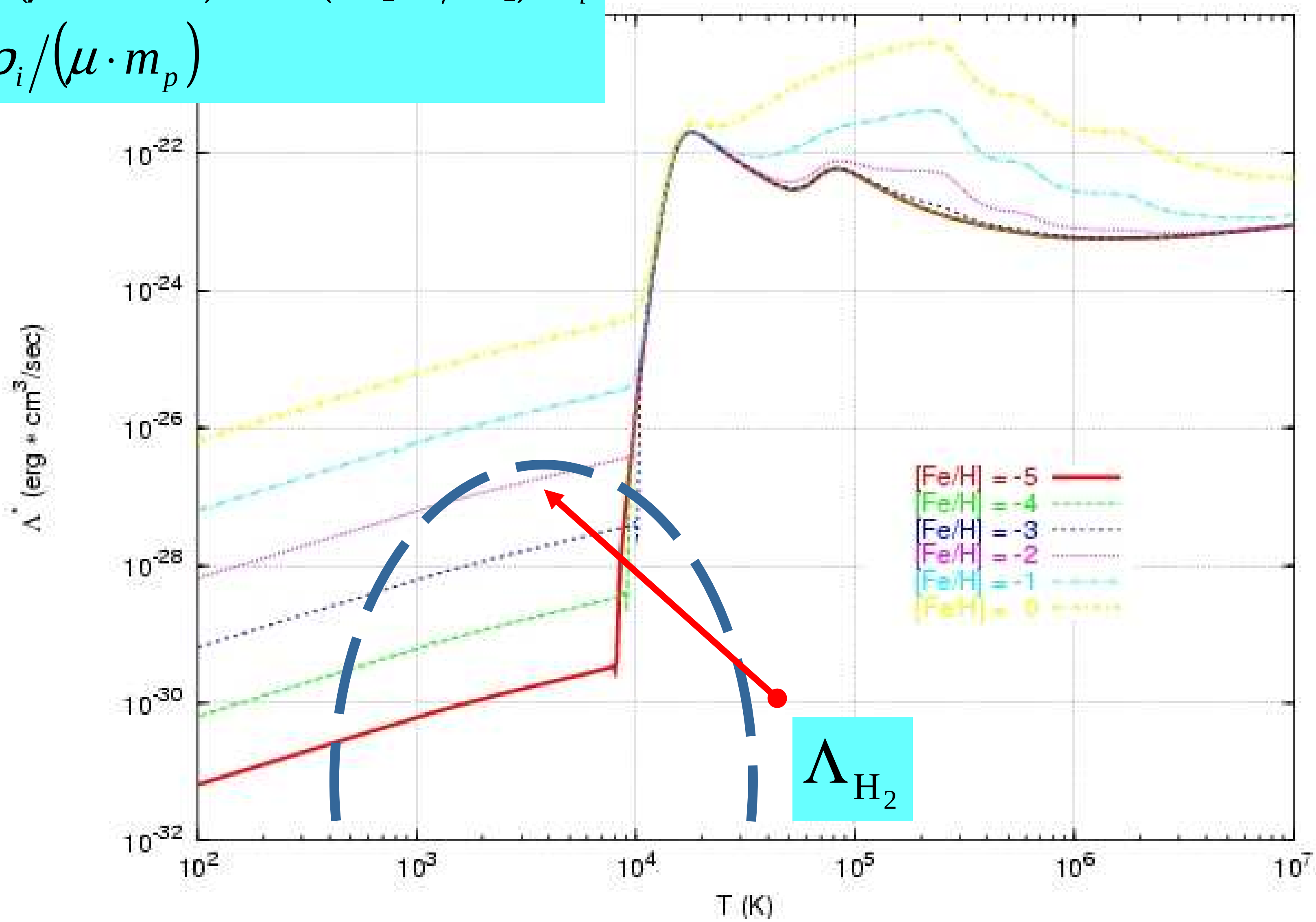
Using the ANN with kdtree

http://www.cs.umd.edu/~mount/ANN/

Delgarno & McCray, 1972; Sutherland & Dopita, 1993

$$\Lambda \equiv \Lambda(\rho, u, Z, \ldots) \cong \Lambda^*(T, [Fe/H]) \cdot n_i^2$$

$$n_i^2 = \rho_i / (\mu \cdot m_p)$$



Integrator

Predictor step:

$$\begin{cases} \vec{v}_i^{\,p} = \vec{v}_i^{\,n} + \vec{a}_i^{\,n} \cdot \Delta t \\ \vec{r}_i^{\,p} = \vec{r}_i^{\,n} + (\vec{v}_i^{\,n} + \vec{v}_i^{\,p}) \cdot \dfrac{\Delta t}{2} \\ u_i^{\,p} = u_i^{\,n} + \dot{u}_i^{\,n} \cdot \Delta t \end{cases}$$

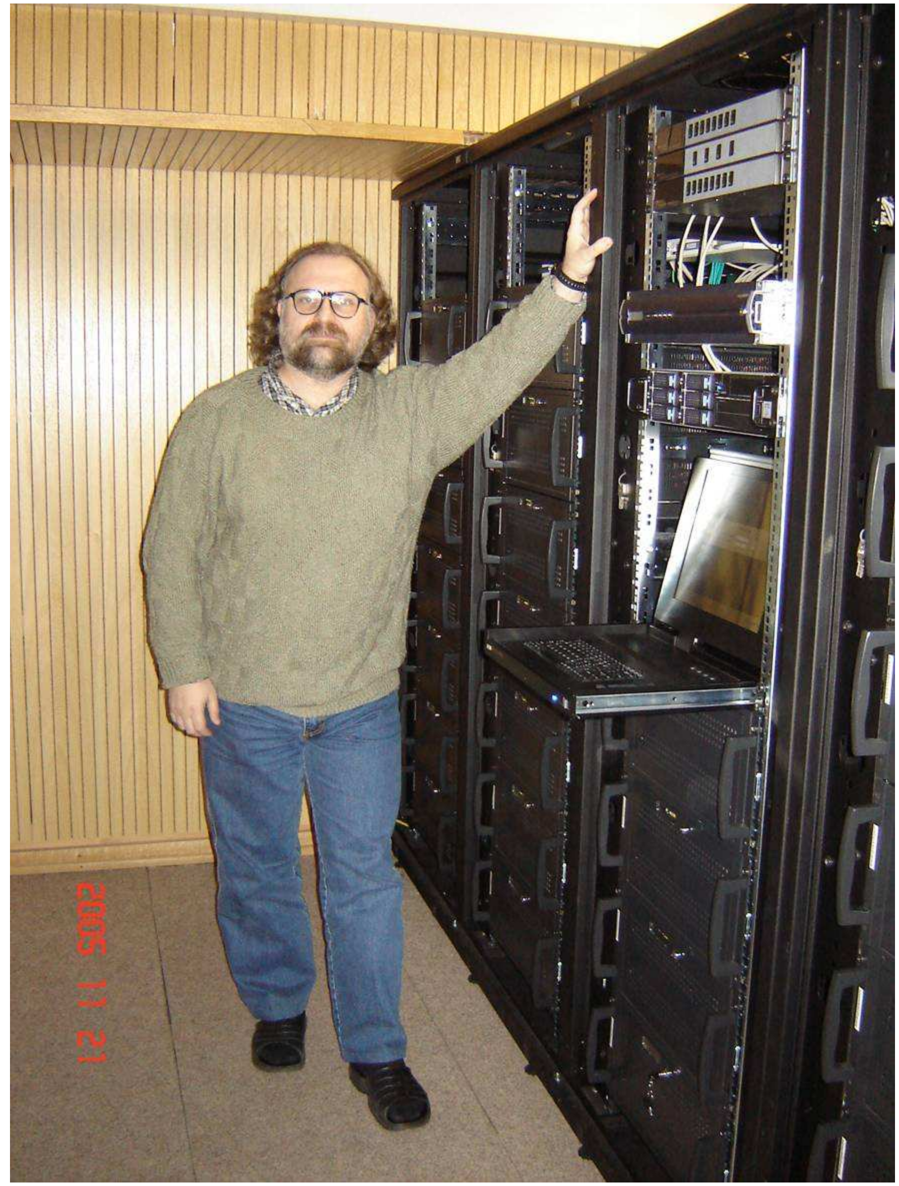
Corrector step:

$$\begin{cases} \vec{v}_i^{\,n+1} = \vec{v}_i^{\,n} + (\vec{a}_i^{\,n} + \vec{a}_i^{\,p}) \cdot \dfrac{\Delta t}{2} \\ \vec{r}_i^{\,n+1} = \vec{r}_i^{\,n} + (\vec{v}_i^{\,n} + \vec{v}_i^{\,n+1}) \cdot \dfrac{\Delta t}{2} \\ u_i^{\,n+1} = u_i^{\,n} + (\dot{u}_i^{\,n} + \dot{u}_i^{\,p}) \cdot \dfrac{\Delta t}{2} \end{cases}$$

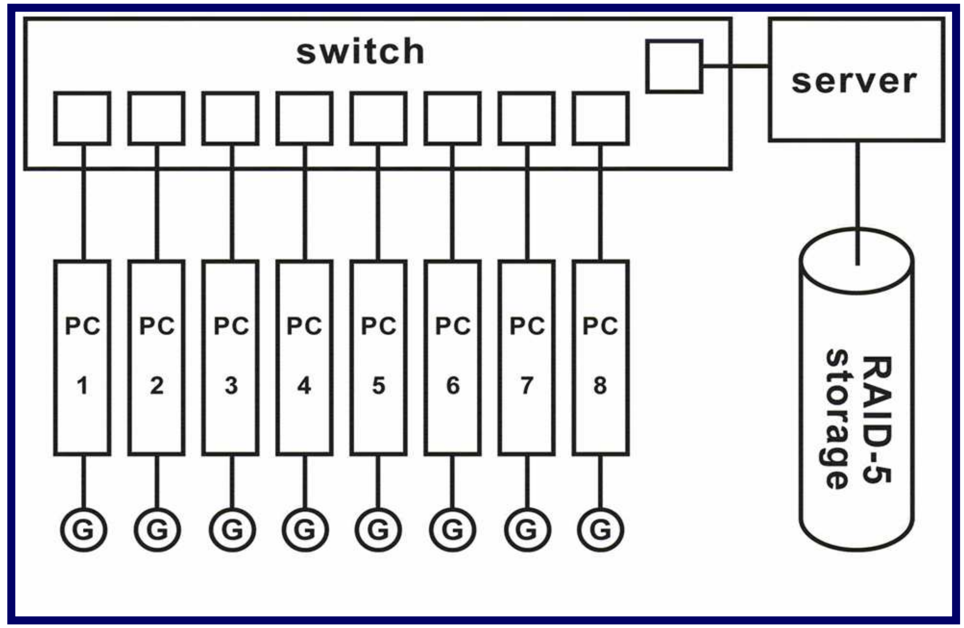
Time step:

$$\Delta t = 0.1 \cdot \min\left(\sqrt{\frac{2 \cdot h_i}{|\vec{a}_i|}}; \frac{h_i}{|\vec{v}_i|}; \frac{h_i}{c_i}; \frac{u_i}{\dot{u}_i} \right)$$

RIT & ARI 32 node GRAPE6a clusters

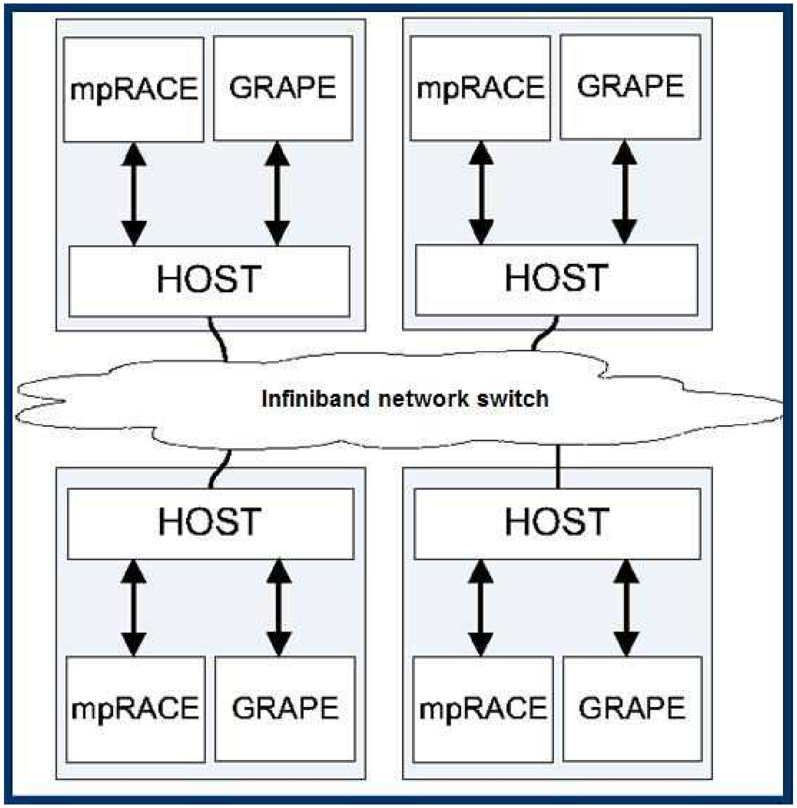


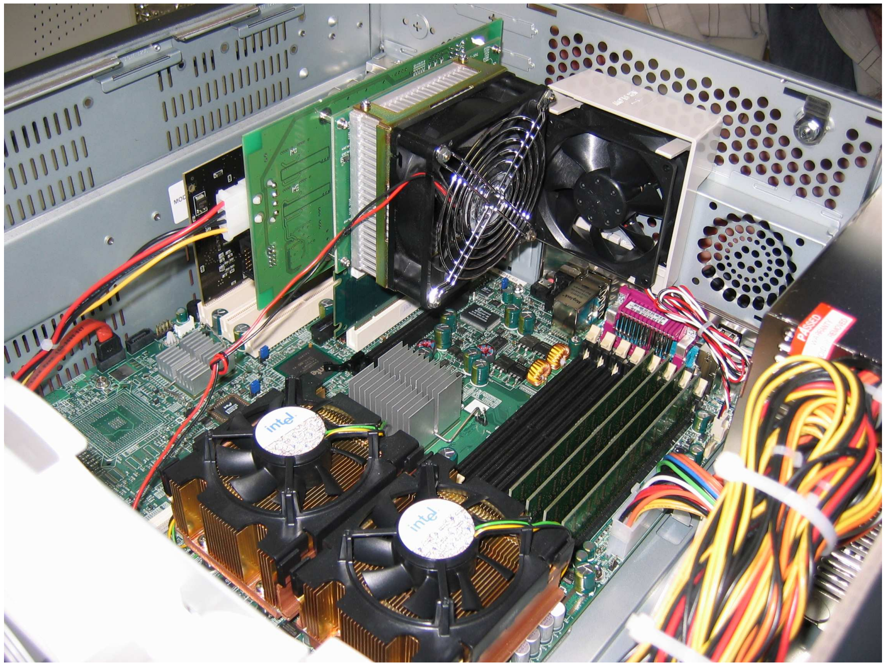
MAO 8+1 node GRAPE6 blx64 cluster

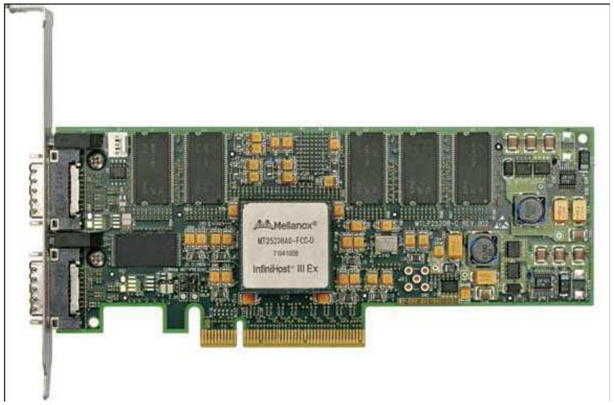


- 9 x2 dual-core Xeon 2.0 GHz
- 9 GRAPE6 blx64
- 5 TB RAID
- Infiniband switch (2x10 Gb/s)
- Speed: ~1 Tflops
- N up to 2M
- Cost: ~100k EUR
- Funding: NASU

ARI 32 node GRAPE6a cluster:





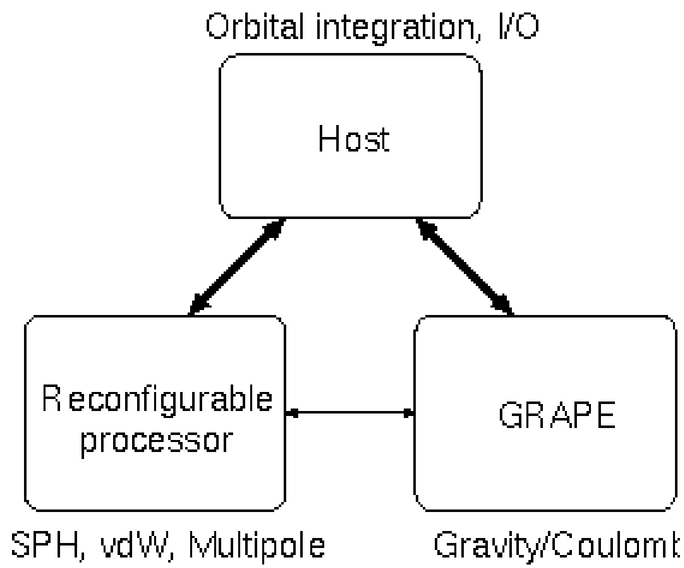


32x2 64 bit-Xeon P4, 3.2 GHz (~2 Gfps)
32 GRAPE6a (~120 Gfps)
32 FPGA-MPRACE (~20 Gfps)
3.5 TB RAID5 disk system
Infiniband, dual port network (~20 Gb/s)

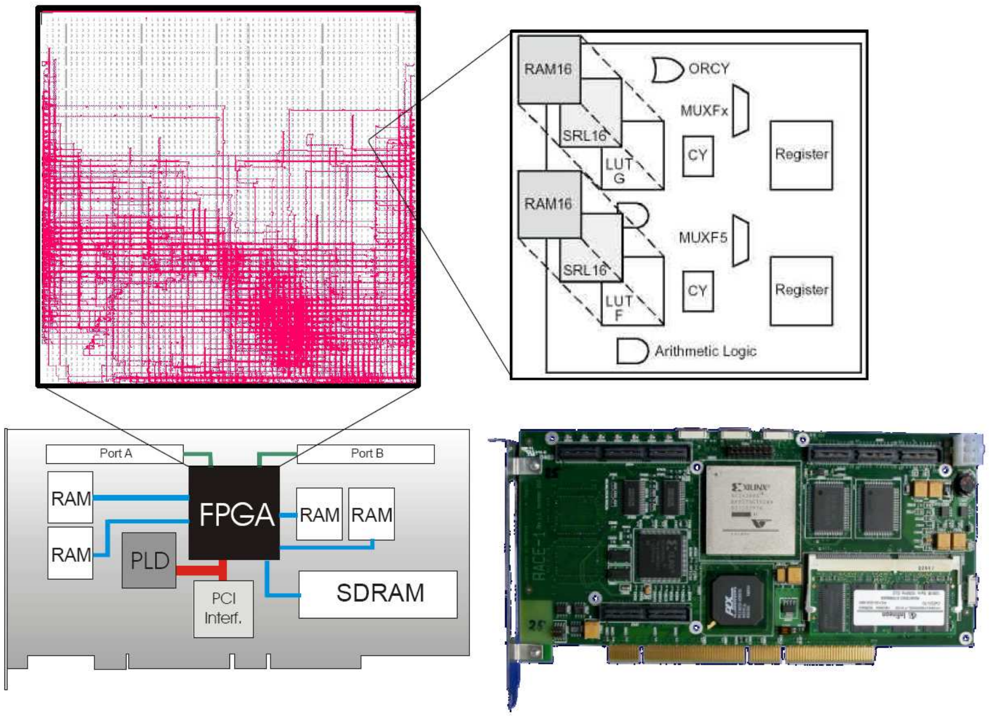
Summary speed: ~4 Tfps
N (direct summation) up to 4M

Volkswagen/Baden-Württemberg ~400k EUR

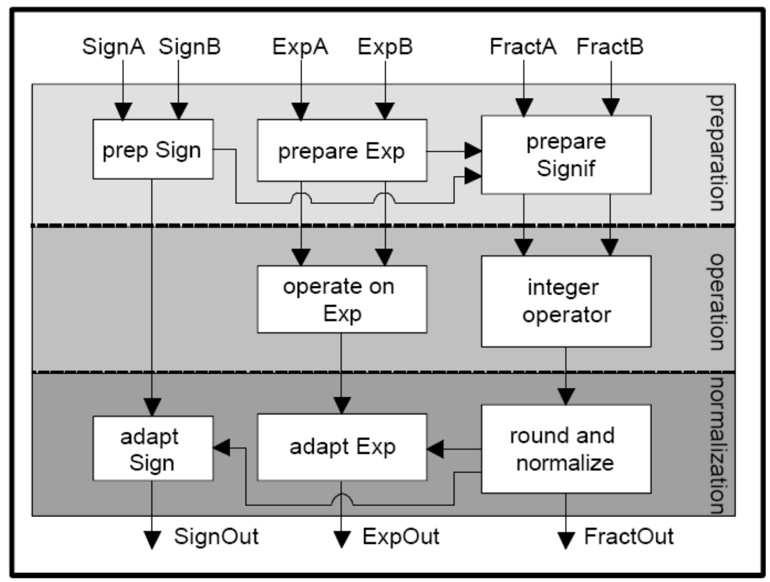
GRACE=GRAPE + MPRACE:

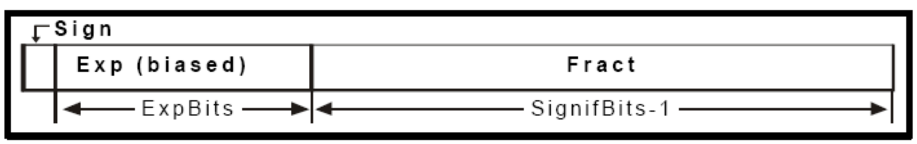


MPRACE FPGA board



FP arithmetic:
16 or 24 mantissa

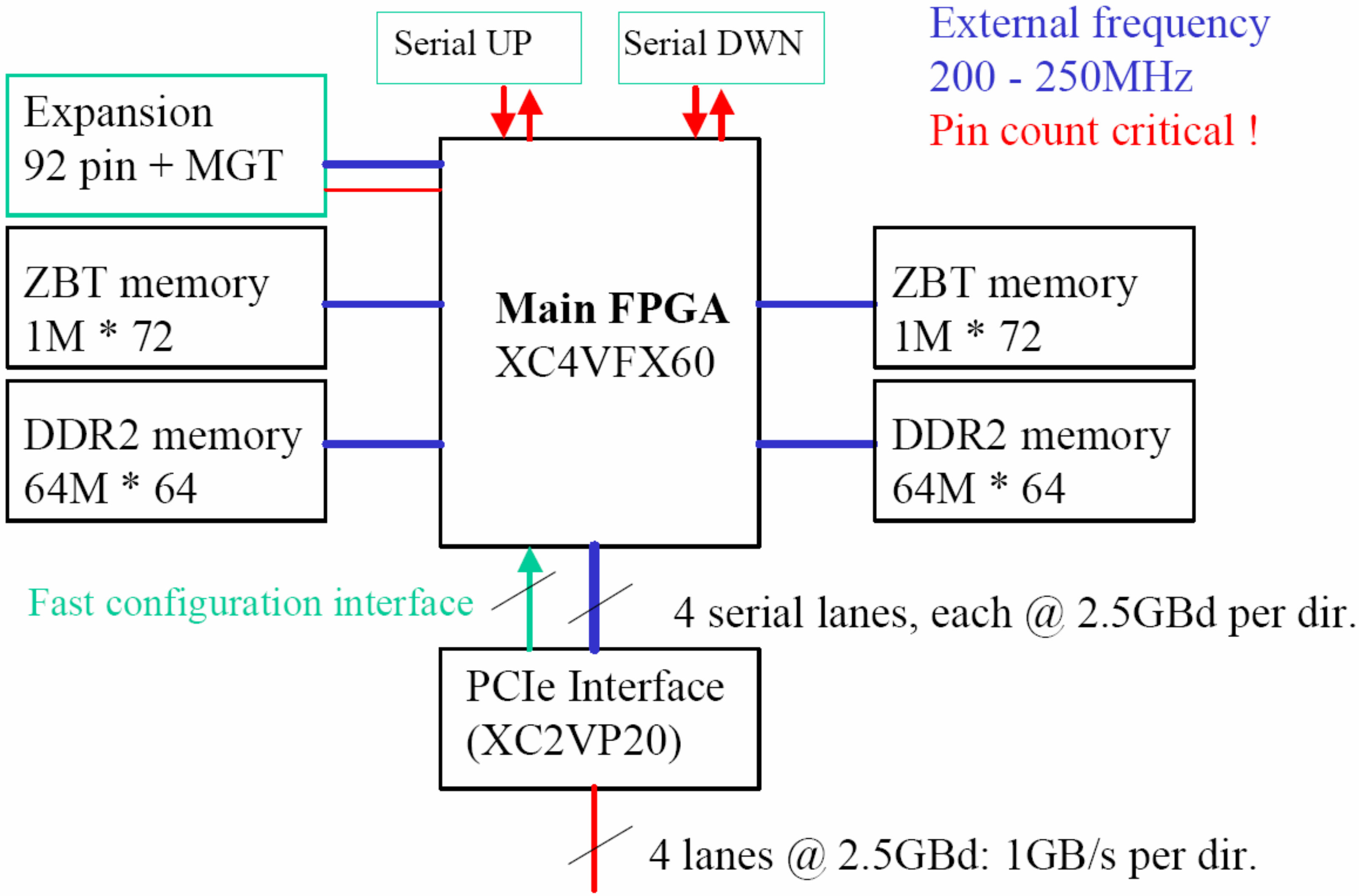




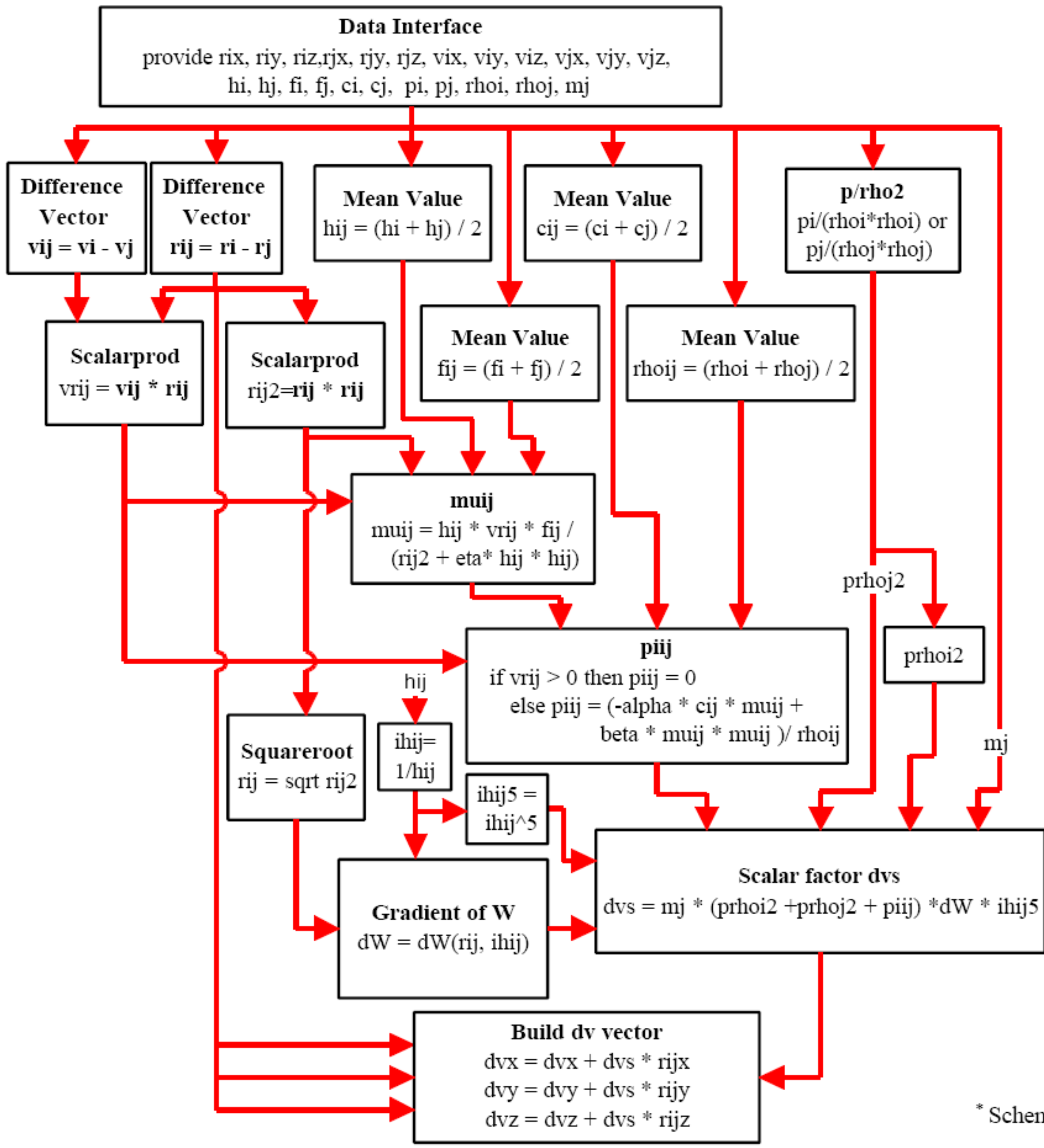
$$f = \begin{cases} +1 \ \text{for Sign} = 0 \\ -1 \ \text{for Sign} = 1 \end{cases} \cdot \left(1 + \text{Fract} \cdot 2^{-(\text{SignifBits} -1)}\right) \cdot 2^{\text{Exp} - \text{bias}}$$

$$\text{with bias} = 2^{(ExpBits - 1)} - 1$$

MPRACE FPGA board

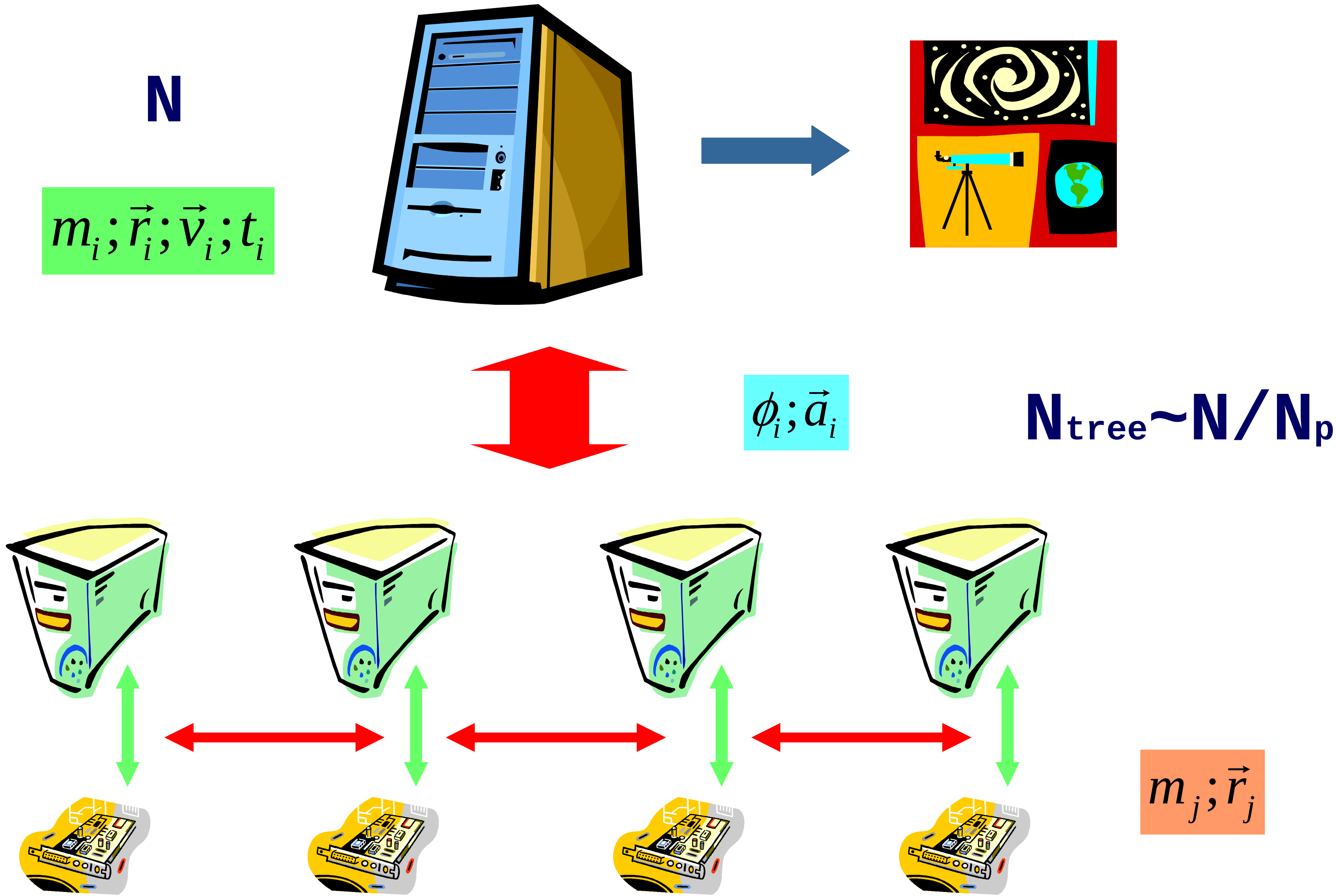


Pressure force pipeline



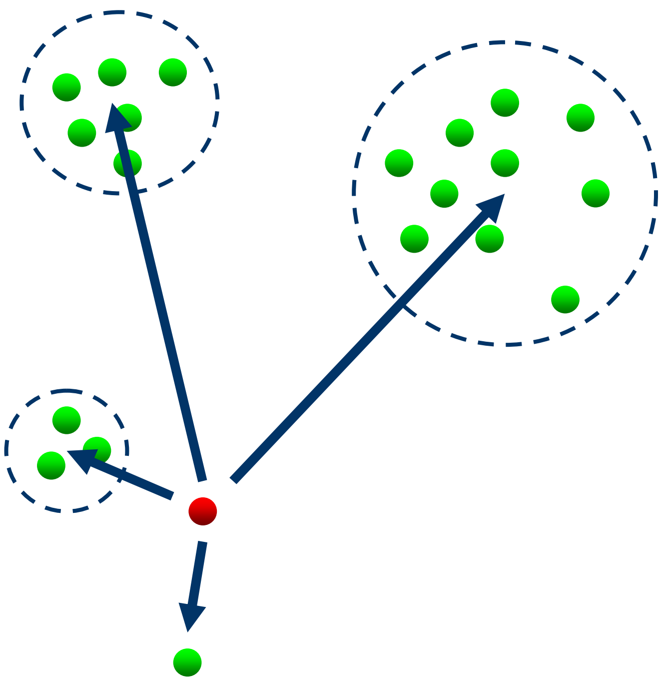
* Scheme doesn't show energy term

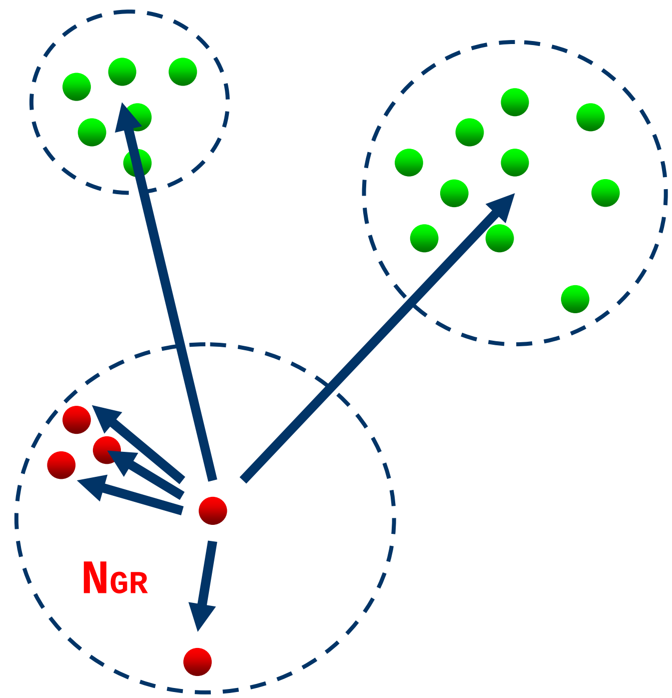
Parallel TREE gravity on the cluster



Parallel TREE gravity on the cluster

Jun Makino: TREE+GRAPE code



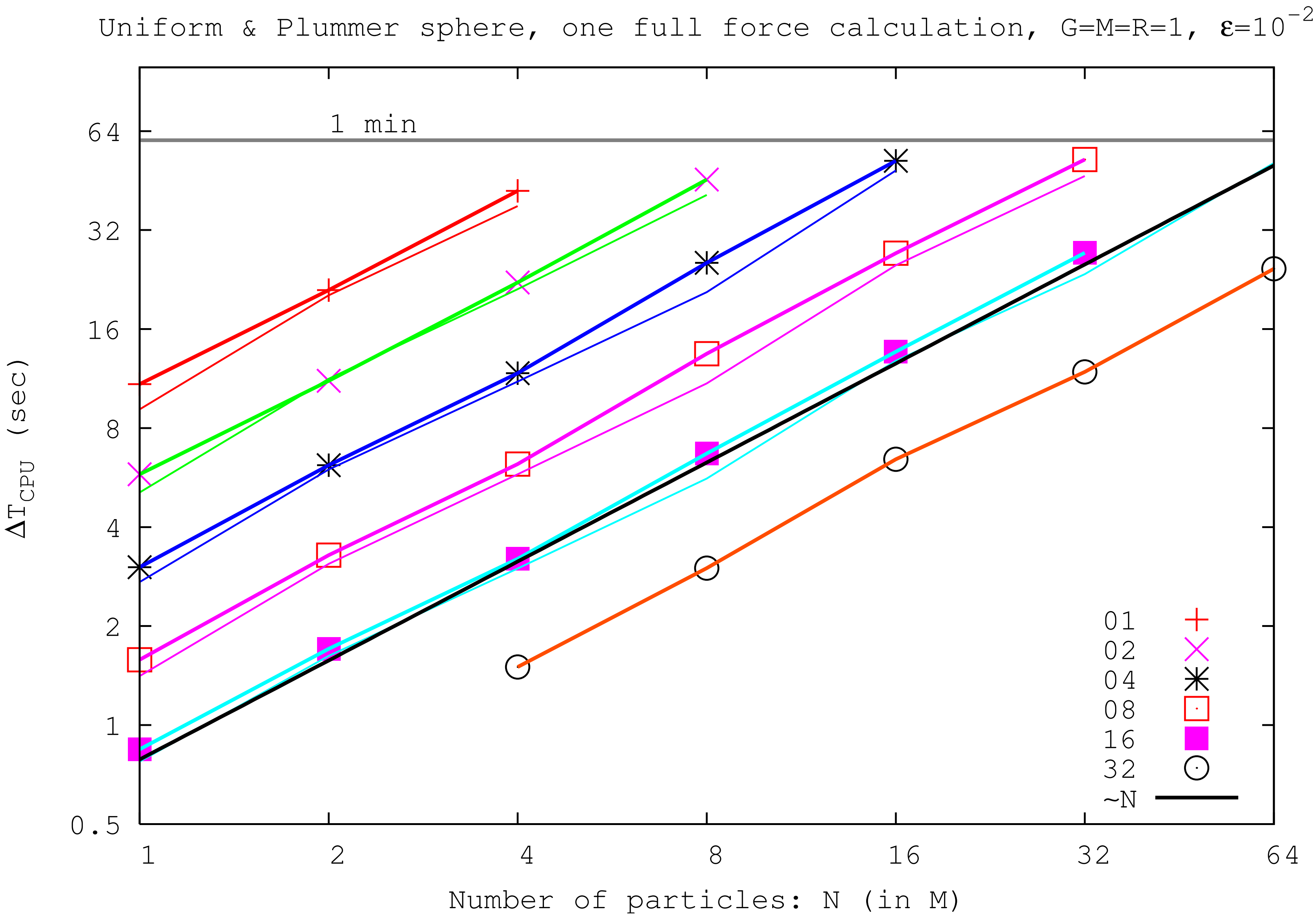


Makino, PASJ, 43, 621 (1991)

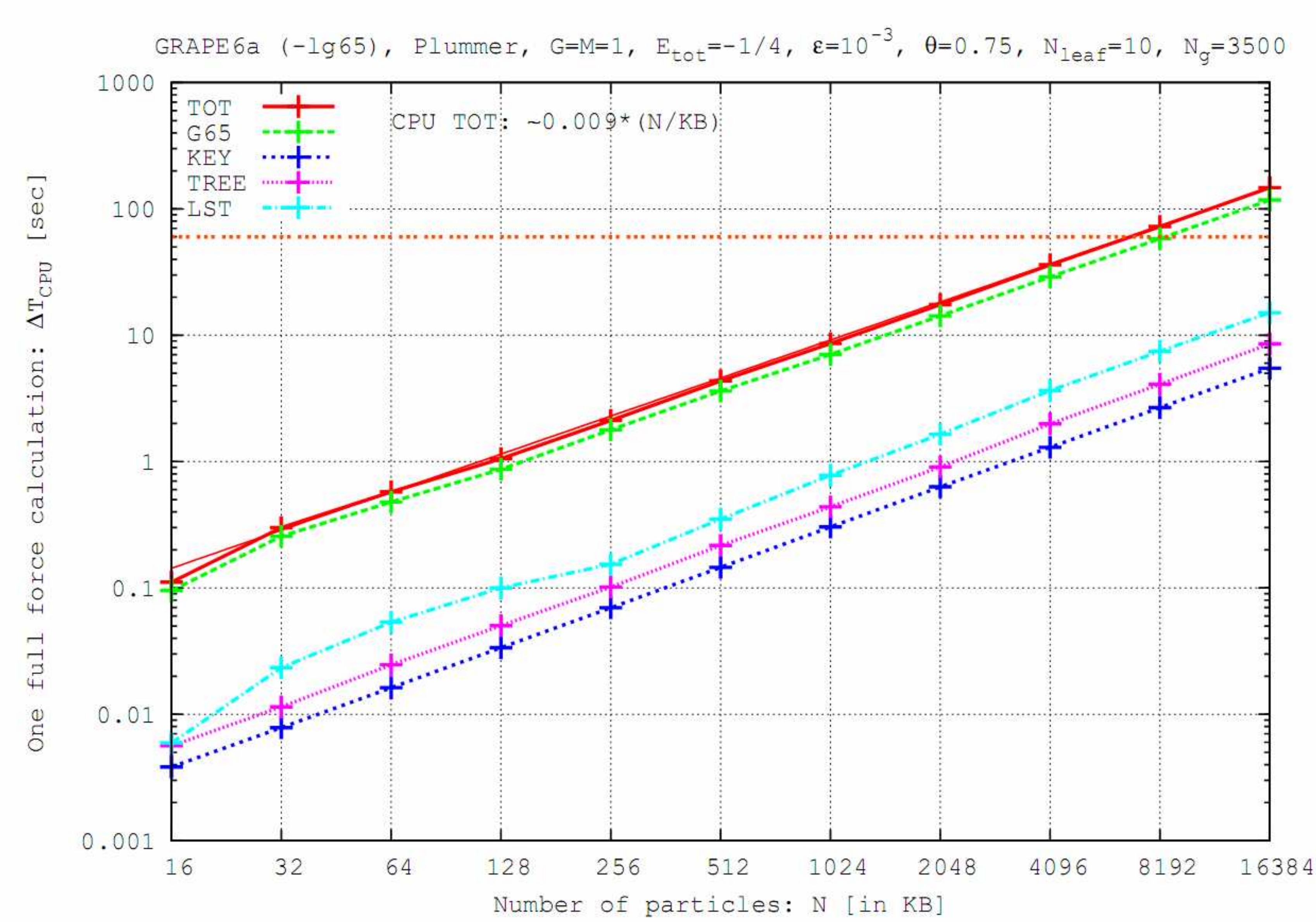
Inter. list on host ~N
Inter. list length -> short…

Makino, PASJ, 56, 521 (2004)
Fukushige, Makino & Kawai, PASJ, 57, 1009 (2005)
One interaction list is shared among NGR particles!
Inter. list on host ~N/NGR
Inter. list length -> larger…

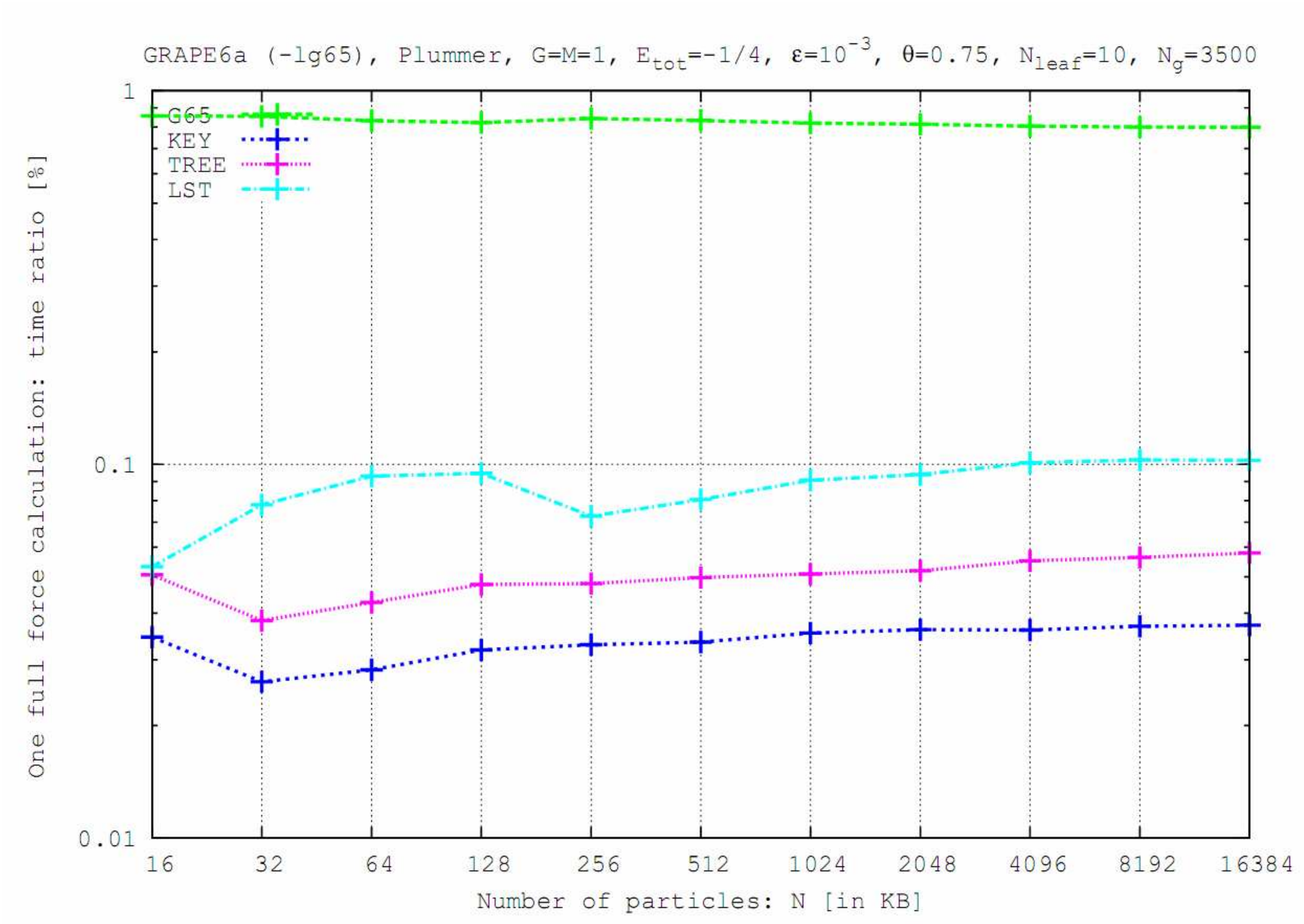
Parallel TREE gravity on the cluster



Parallel TREE gravity on the cluster



Parallel TREE gravity on the cluster



SPH - test

Adiabatic collapse of a cold gas sphere.

Evrard, 1988
Steinmetz & Muller, 1993
Carraro et al., 1998
Springel et al., 2001

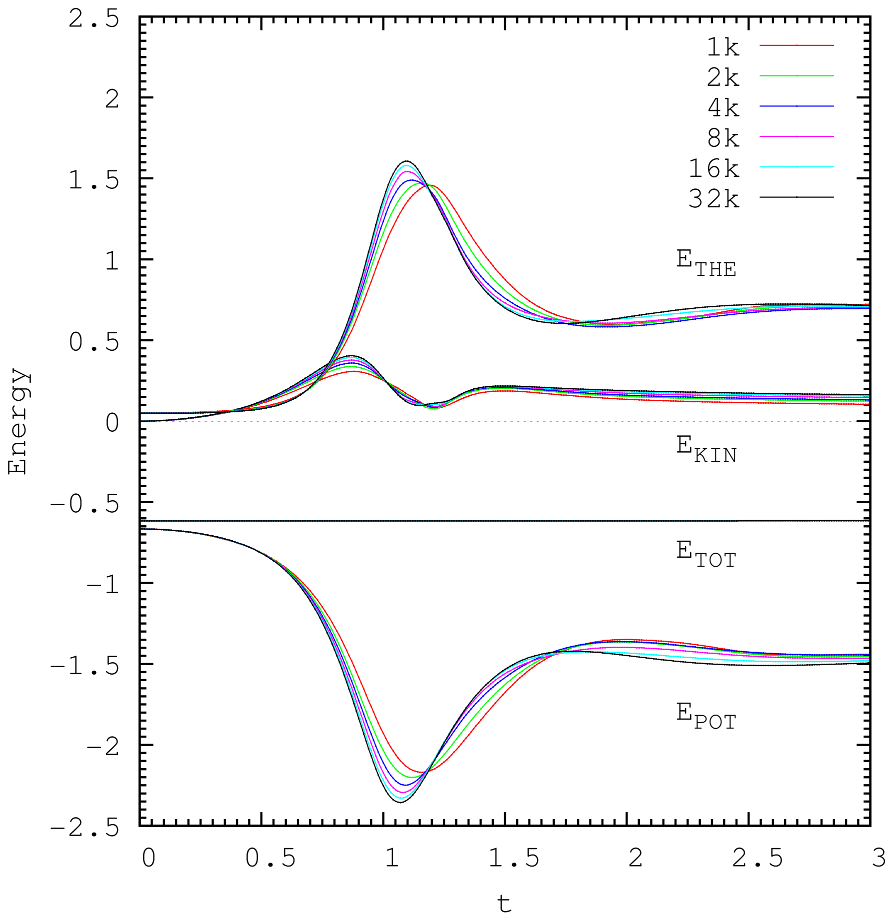
$$\rho = \frac{M}{2 \cdot \pi \cdot R^2} \cdot \frac{1}{r}$$

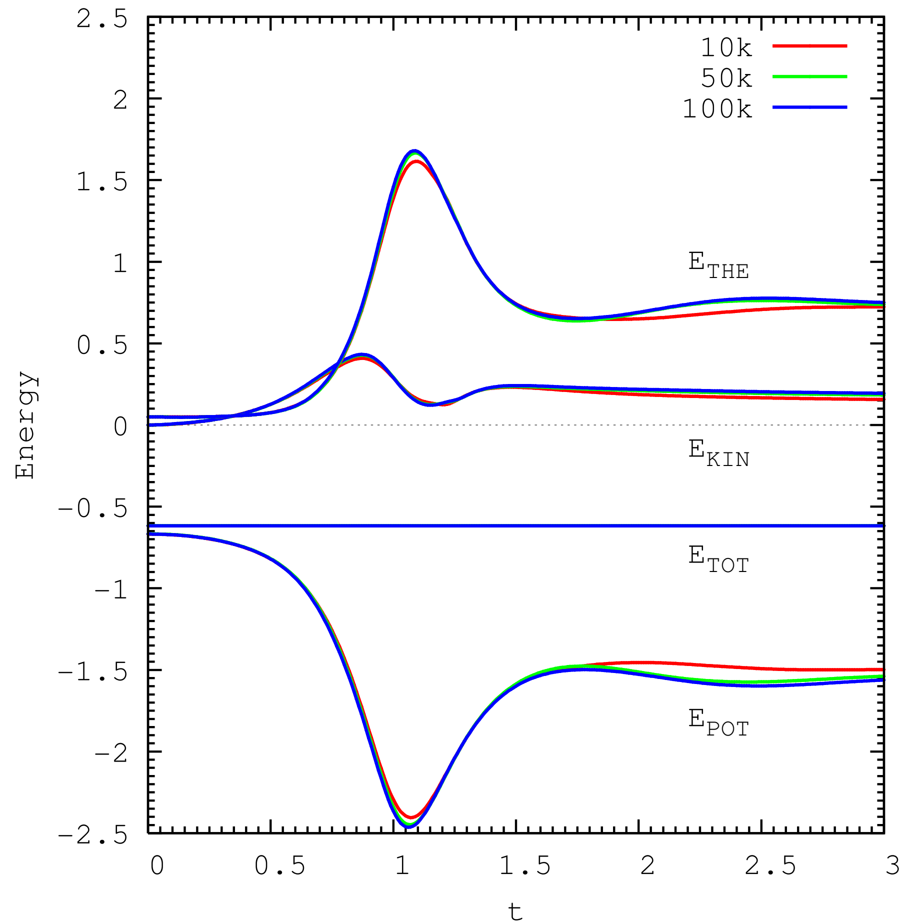
$$E_G = -\frac{2}{3} \cdot \frac{G \cdot M^2}{R}$$

$$G = M = R = 1$$

$$u = 0.05 \cdot \frac{G \cdot M}{R}$$

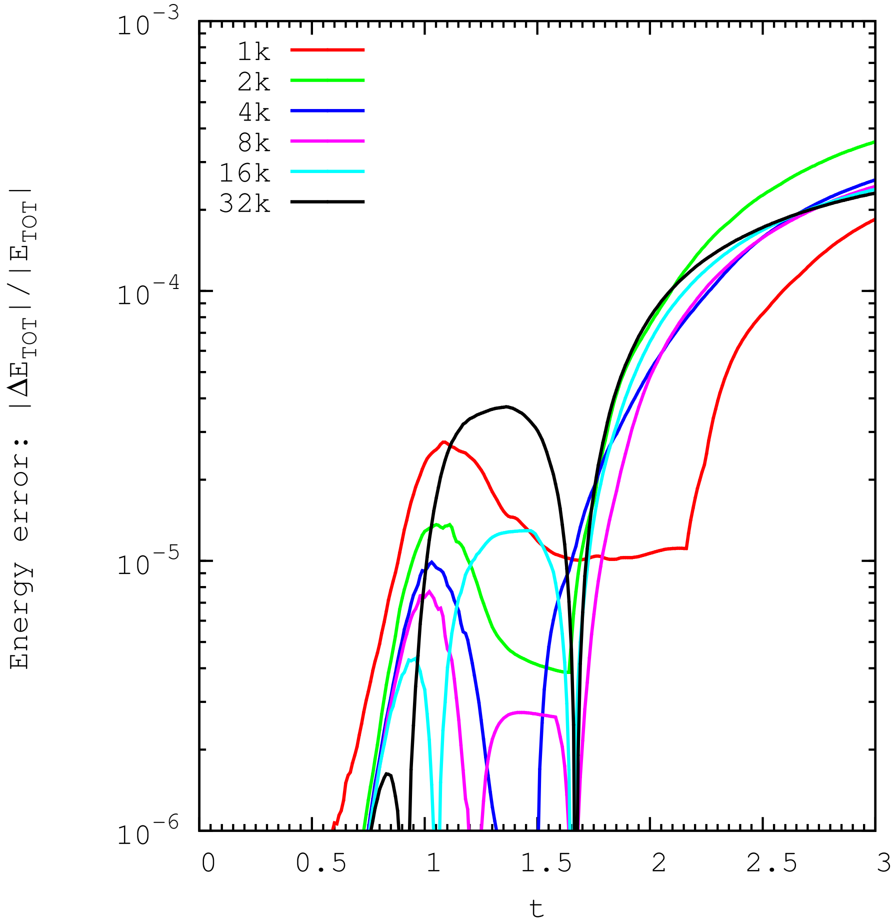
SPH - test

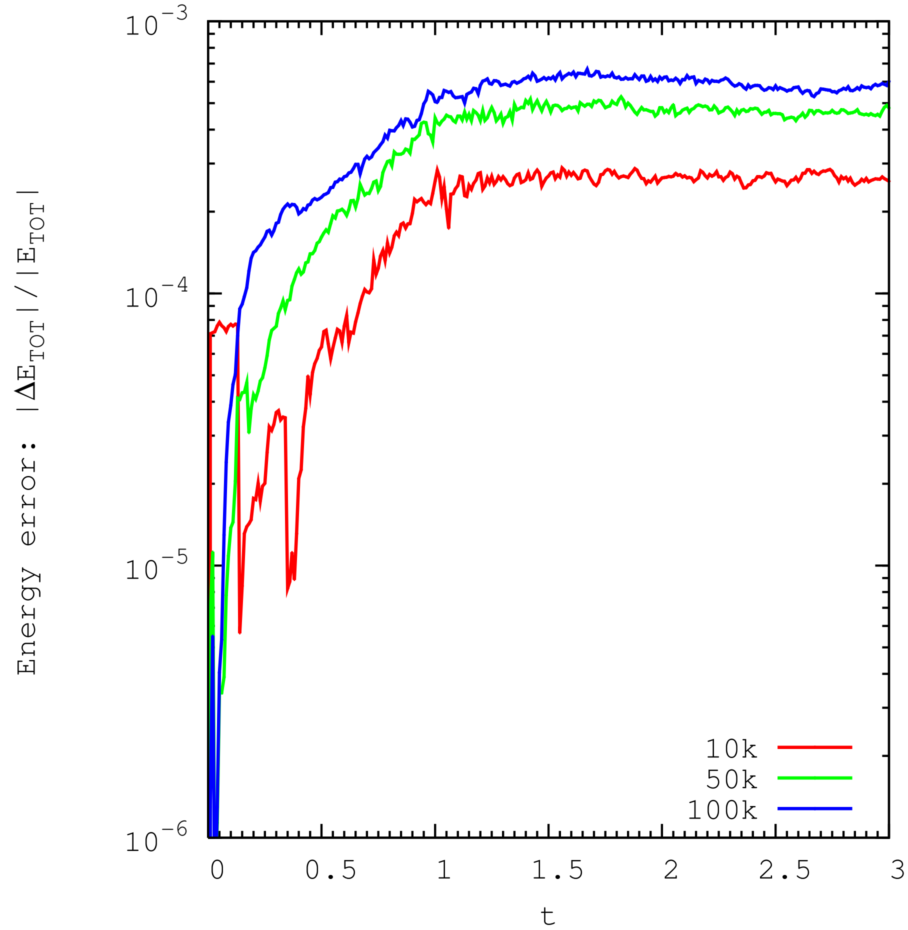




Berczik (N_{CPU}=1) **Nakasato (N_{CPU}=4)**

SPH - test



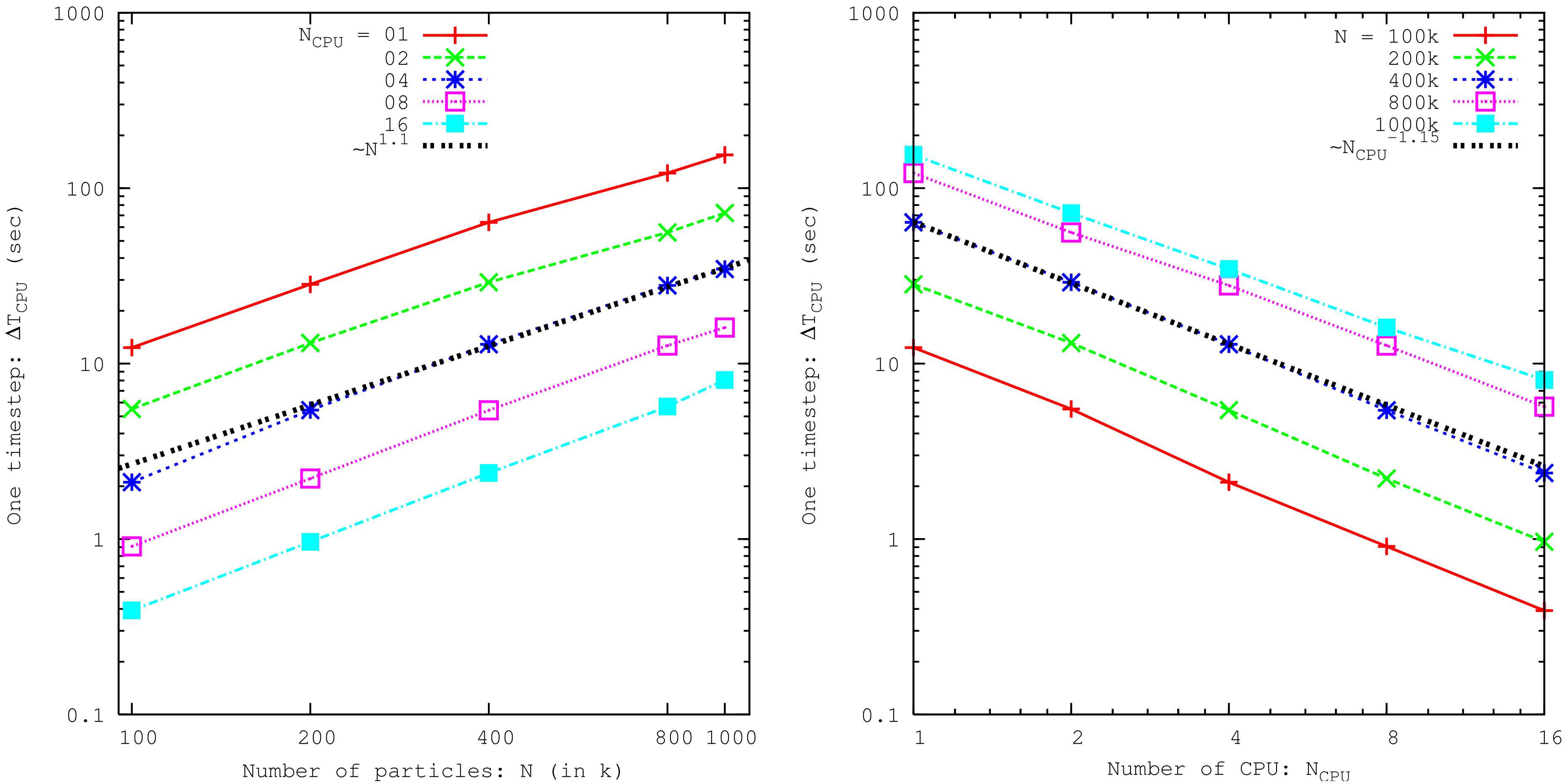


Berczik (N_{CPU}=1)

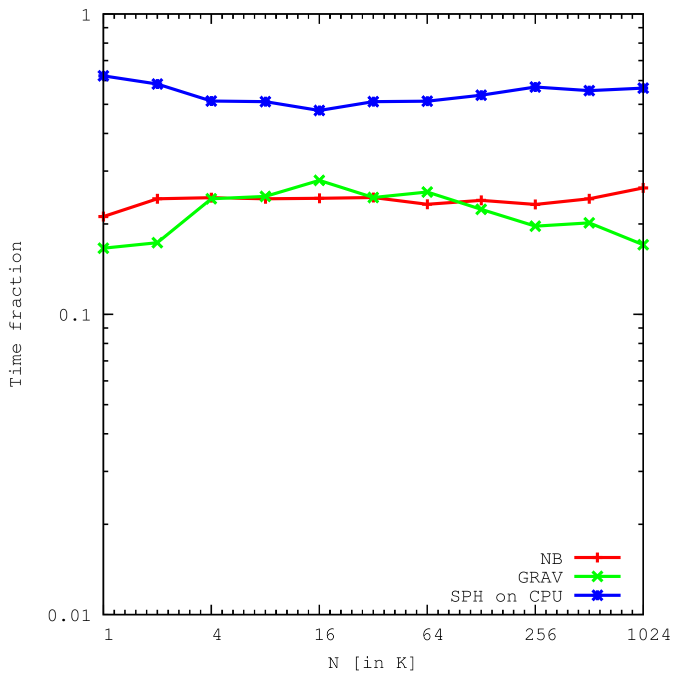
Nakasato (N_{CPU}=4)

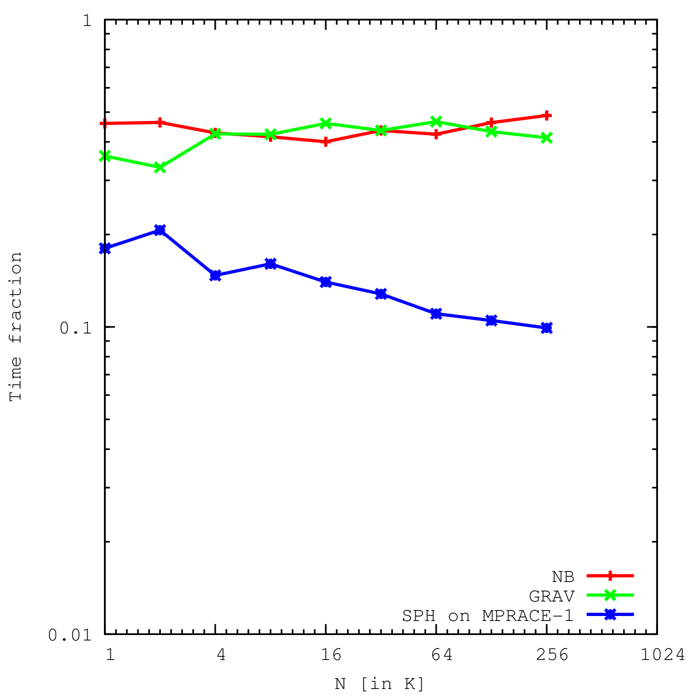
Scaling results

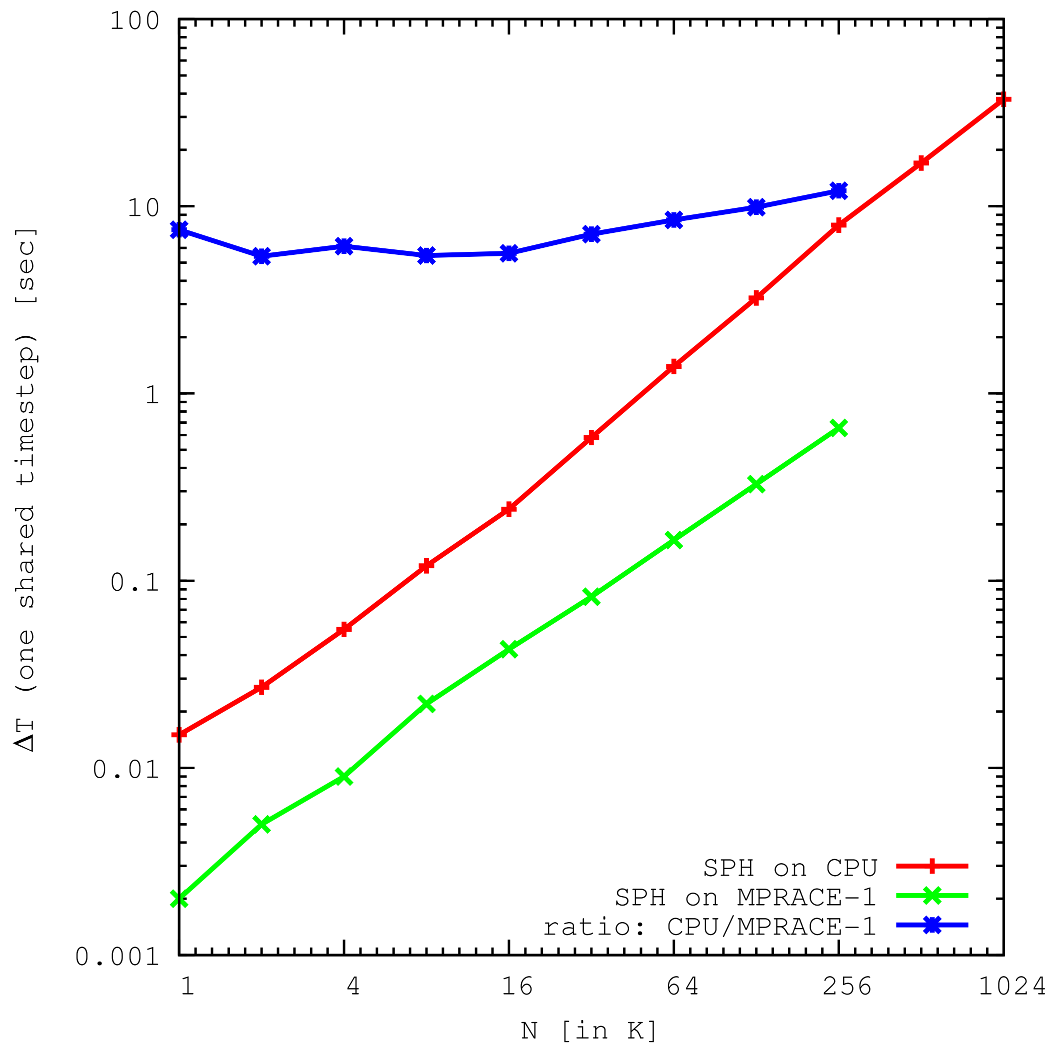
GRAPE + SPH code: One timestep integration



SPH speedup with MPRACE



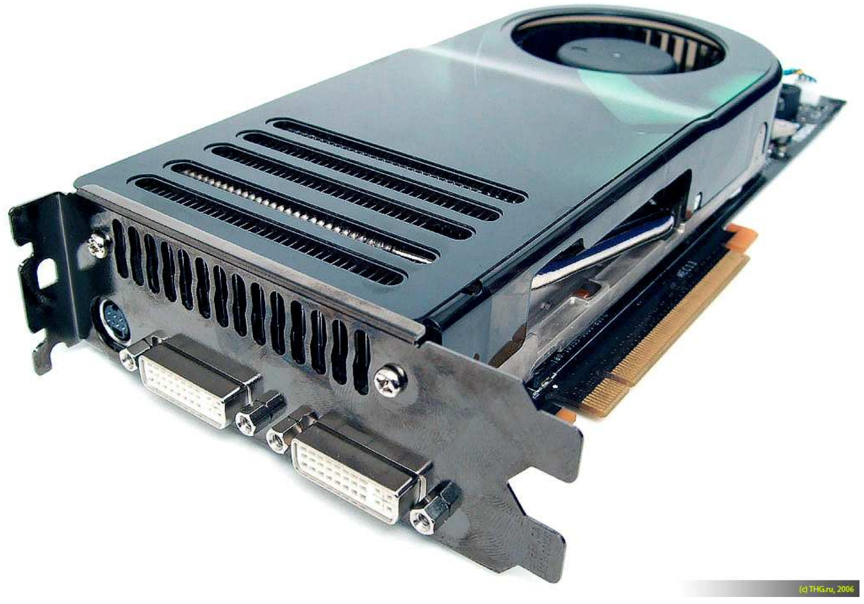




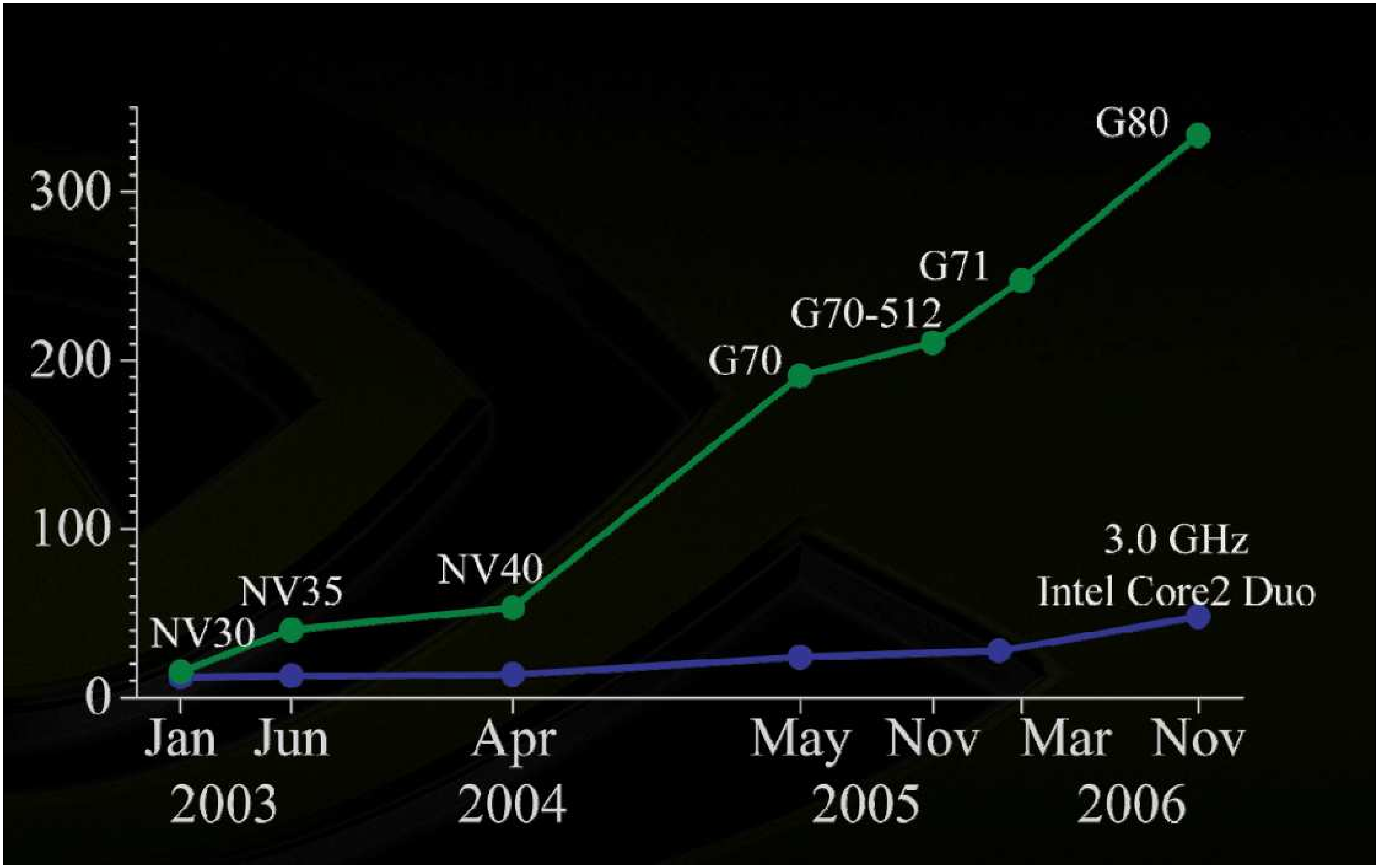
Hardware

2007…
GeForce 8800 GTX, 128 Stream Proc., 768 MB
GeForce 8800 GTS, 128 Stream Proc., 512 MB
GeForce 8800 GT, 112 Stream Proc., 512 MB

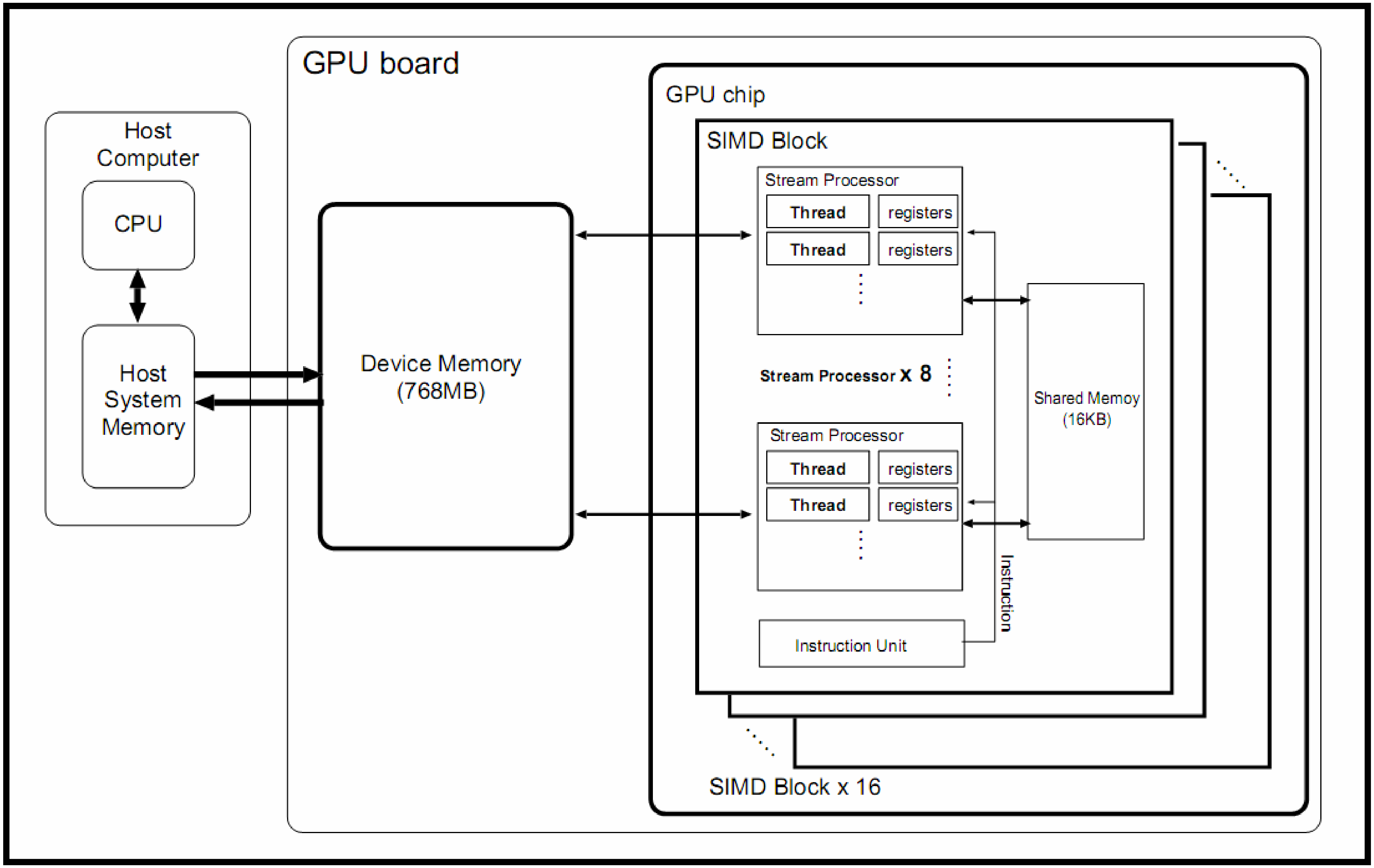
2008…
GeForce 9800 GTX, 128 Stream Proc., 512 MB
GeForce 9800 GX2, 256 Stream Proc., 1 GB
GeForce 9800 GT, 64 Stream Proc., 512 MB



CPU vs. GPU speedup timeline

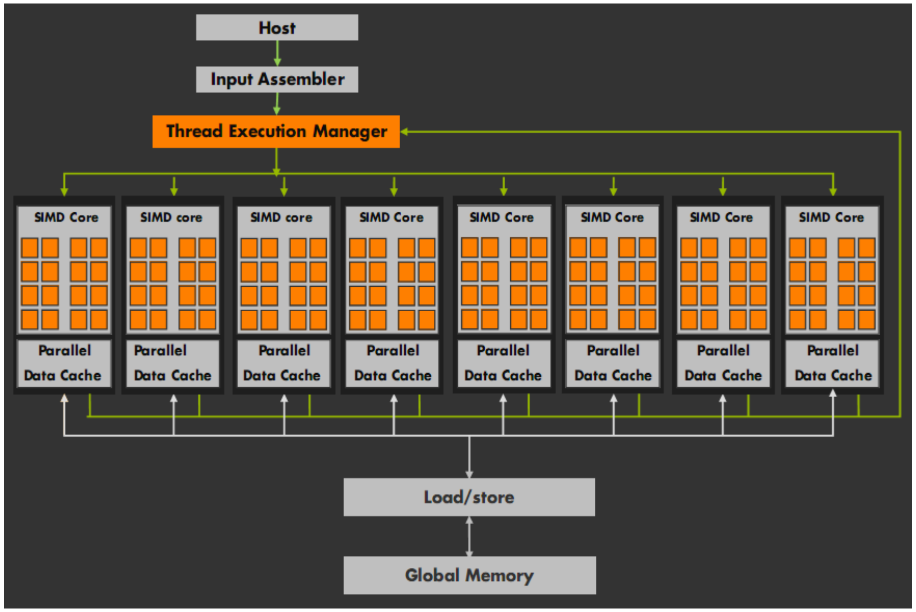


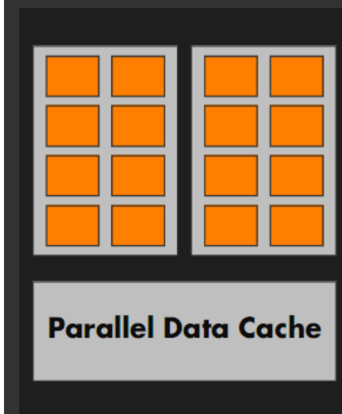
Hardware



GPU board
GPU chip
SIMD Block
Host
Computer
CPU
Host
System
Memory
Device Memory
(768MB)
Stream Processor
Thread
registers
Thread
registers
Stream Processor x 8
Stream Processor
Thread
registers
Thread
registers
Shared Memoy
(16KB)
Instruction
Instruction Unit
SIMD Block x 16

Hardware





Each core

- 8 functional units
- SIMD 16/32 "warp"
- 8-10 stage pipeline
- Thread scheduler
- 128-512 threads/core
- 16 KB shared memory

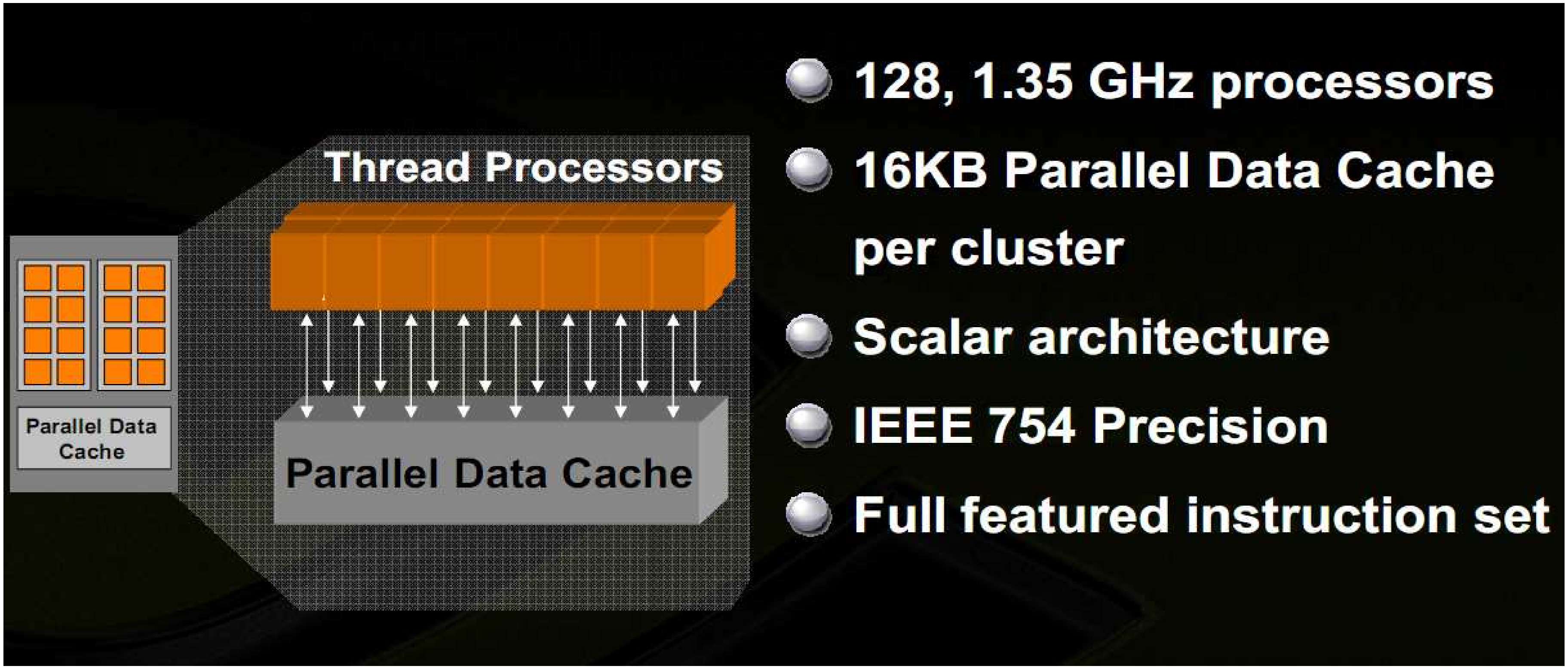
Total #threads/chip

16 * 512 = 8K

GeForce 8800 GTX:

575 MHz * 128 processors * 2 flop/inst * 2 inst/clock = 333 Gflops

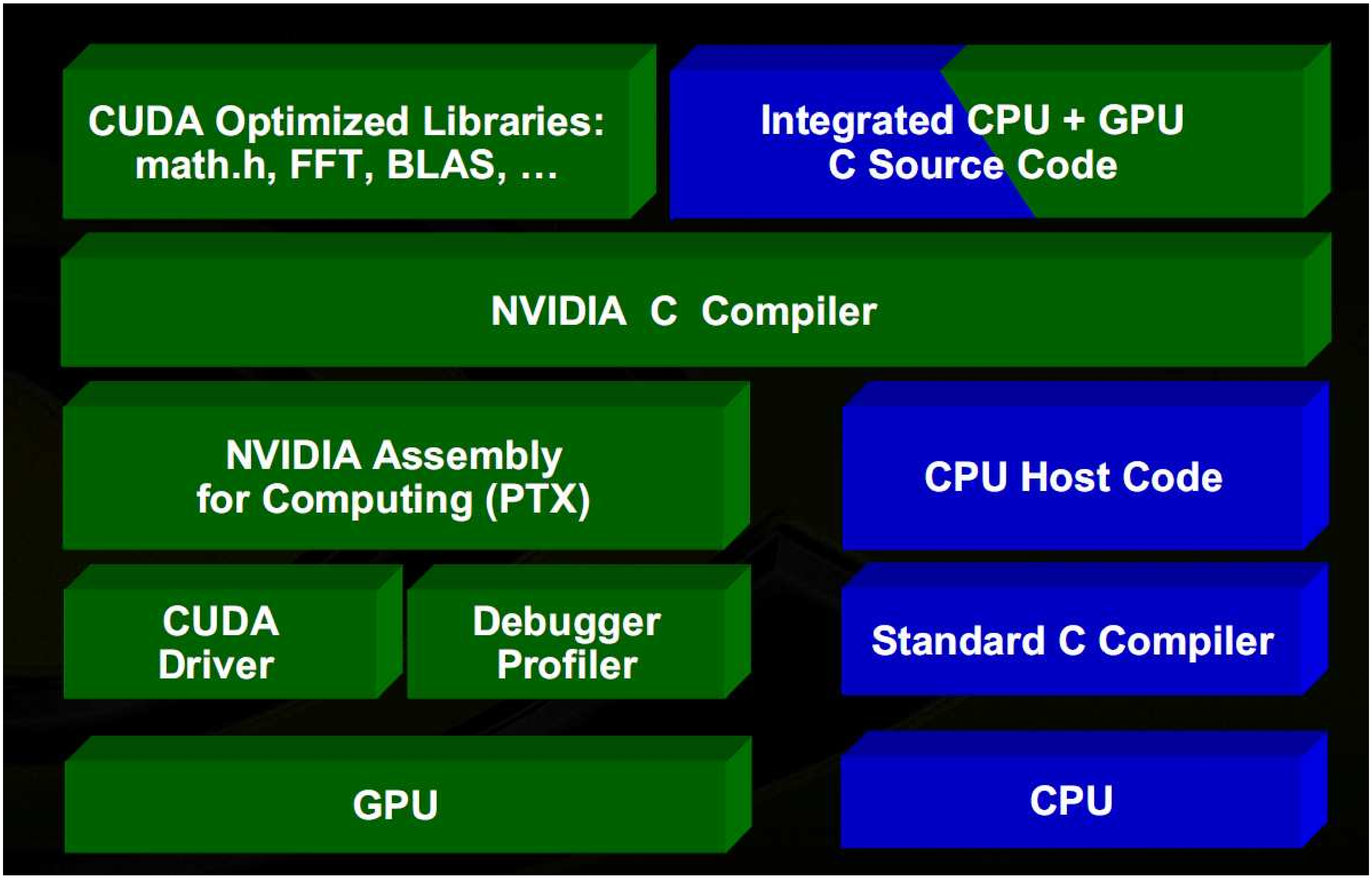
Hardware



GeForce 8800 GTX:

575 MHz * 128 processors * 2 flop/inst * 2 inst/clock = 333 Gflops

CUDA


CUDA Optimized Libraries:
math.h, FFT, BLAS, …
Integrated CPU + GPU
C Source Code
NVIDIA C Compiler
NVIDIA Assembly
for Computing (PTX)
CPU Host Code
CUDA
Driver
Debugger
Profiler
Standard C Compiler
GPU
CPU

Simple CUDA example

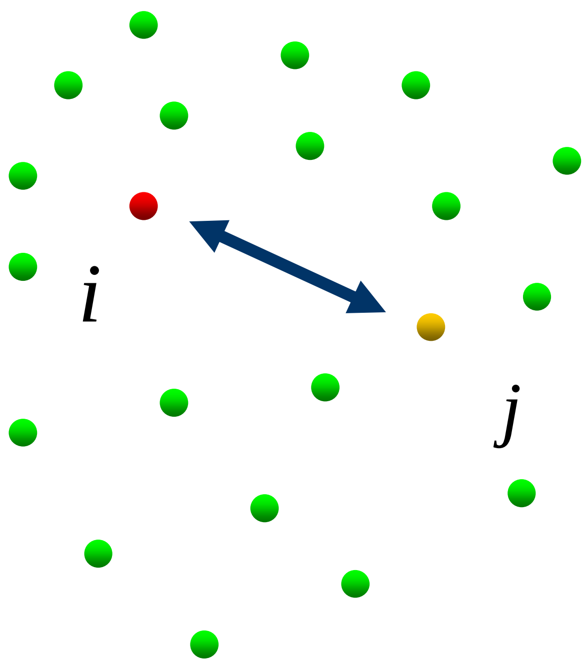
CPU C program

```
void addMatrix(float *a, float *b,
               float *c, int N)
{
  int i, j, index;
  for (i = 0; i < N; i++) {
    for (j = 0; j < N; j++) {
      index = i + j * N;
      c[index]=a[index] + b[index];
    }
  }
}

void main()
{
  .....
  addMatrix(a, b, c, N);
}
```

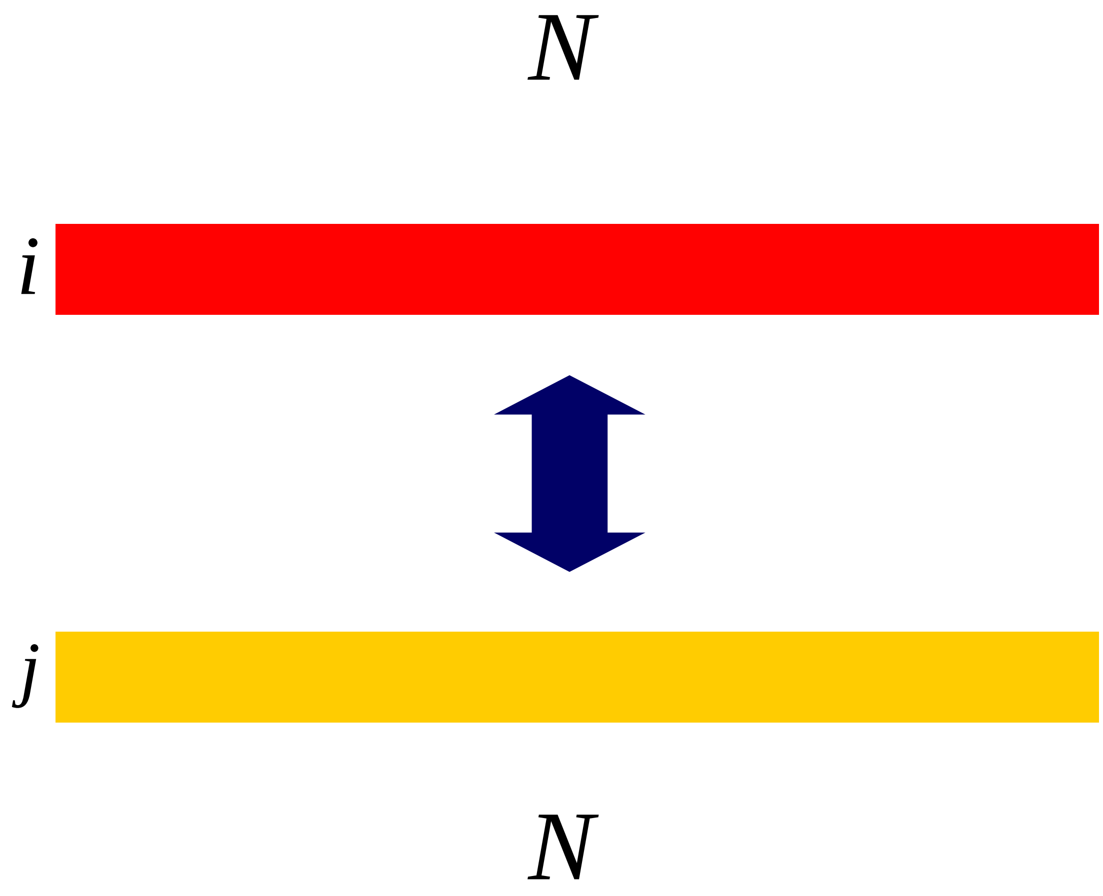
CUDA C program

```
__global__ void addMatrix(float *a,float *b,
                          float *c, int N)
{
  int i=blockIdx.x*blockDim.x+threadIdx.x;
  int j=blockIdx.y*blockDim.y+threadIdx.y;
  int index = i + j * N;
  if ( i < N && j < N)
    c[index]= a[index] + b[index];
}

void main()
{
  ..... // allocate & transfer data to GPU
  dim3 dimBlk (blocksize, blocksize);
  dim3 dimGrd (N/dimBlk.x, N/dimBlk.y);
  addMatrix<<<dimGrd,dimBlk>>>(a, b, c,N);
}
```

Basic idea of any N-body code



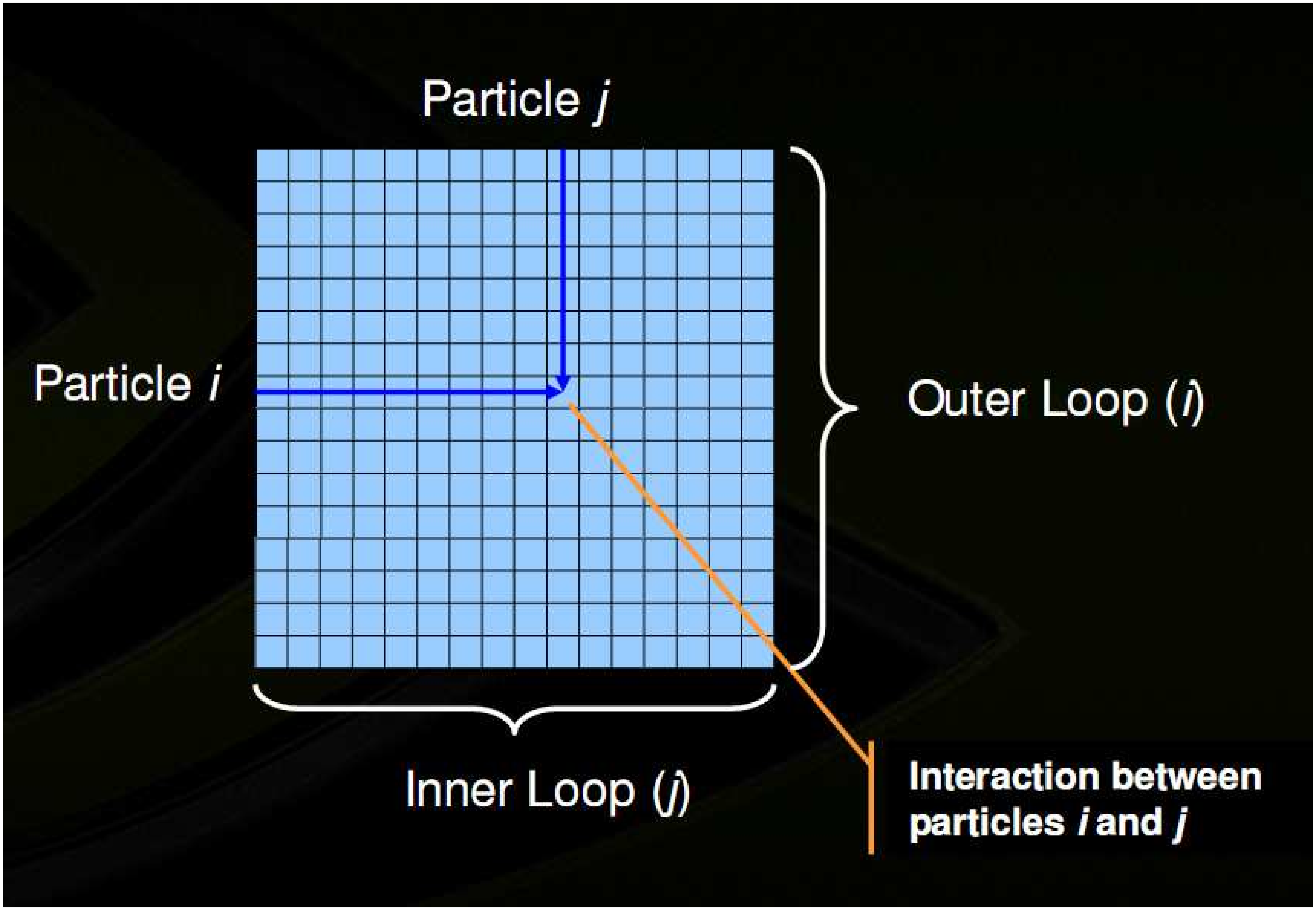
m_j



$\vec{r}_i$

$\vec{v}_i$

t_i

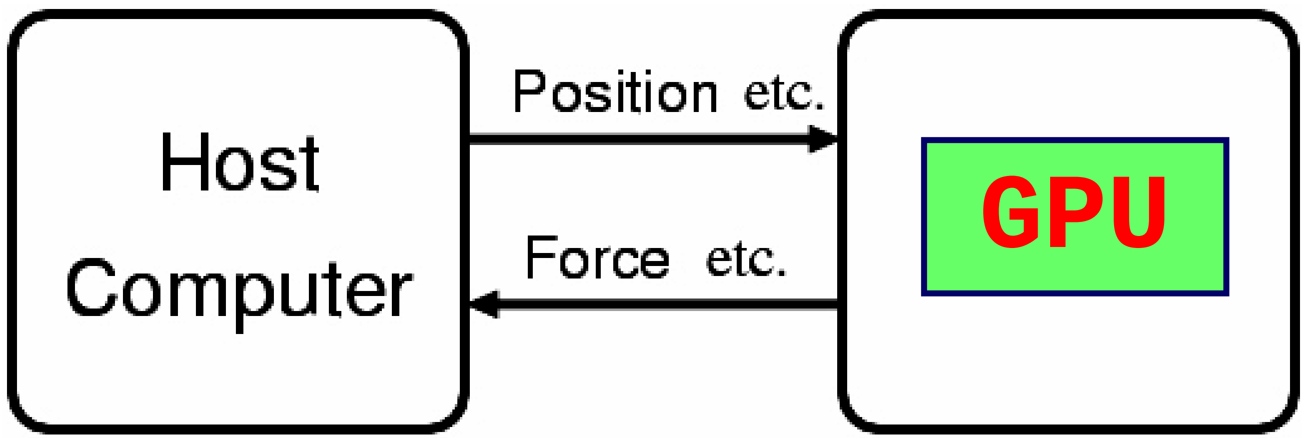

Particle *j*
Particle *i*
Outer Loop (*i*)
Inner Loop (*j*)
Interaction between particles *i* and *j*

```
__device__ float3
bodyBodyInteraction(float3 ai, float4 bi, float4 bj) {
    float3 r;
    r.x = bi.x - bj.x;                 // r_ij [3 FLOPS]
    r.y = bi.y - bj.y;
    r.z = bi.z - bj.z;

    // distSqr = dot(r_ij, r_ij) + EPS^2  [6 FLOPS]
    float distSqr = r.x * r.x + r.y * r.y + r.z * r.z;
    distSqr += softeningSquared;

    // invDistCube =1/distSqr^(3/2)       [4 FLOPS (2 mul, 1 sqrt, 1 inv)]
    float distSixth = distSqr * distSqr * distSqr;
    float invDistCube = 1.0f / sqrtf(distSixth);

    float s = bj.w * invDistCube;  // s = m_j * invDistCube [1 FLOP]

    ai.x += r.x * s;                   // a_i =  a_i + s * r_ij [6 FLOPS]
    ai.y += r.y * s;
    ai.z += r.z * s;
    return ai;
}
```

Total: 20 FLOPS

Basic idea of GRAPE/GPU N-body code

~N

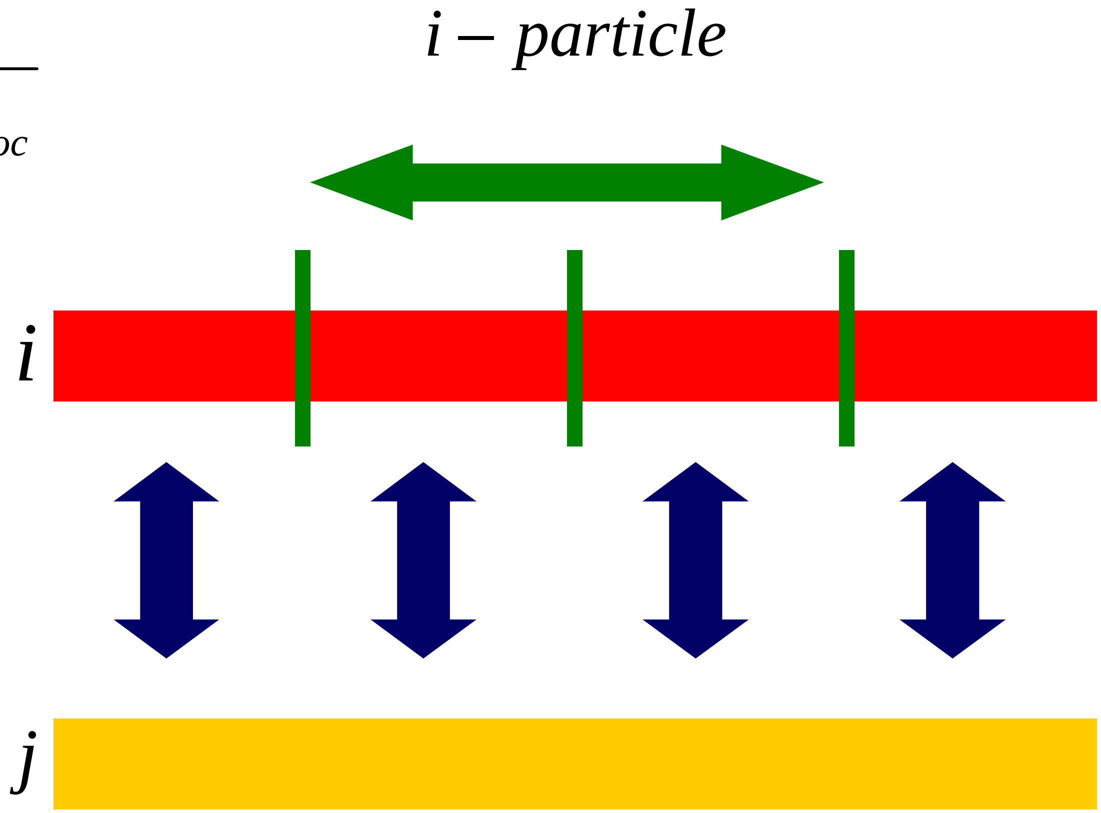


~N^2

$$\vec{a}_i = \sum_{j=1;\ j\neq i}^{N} \vec{f}_{ij} \qquad \vec{f}_{ij} = -\frac{G\cdot m_j}{(r_{ij}^2+\varepsilon^2)^{3/2}}\vec{r}_{ij}$$

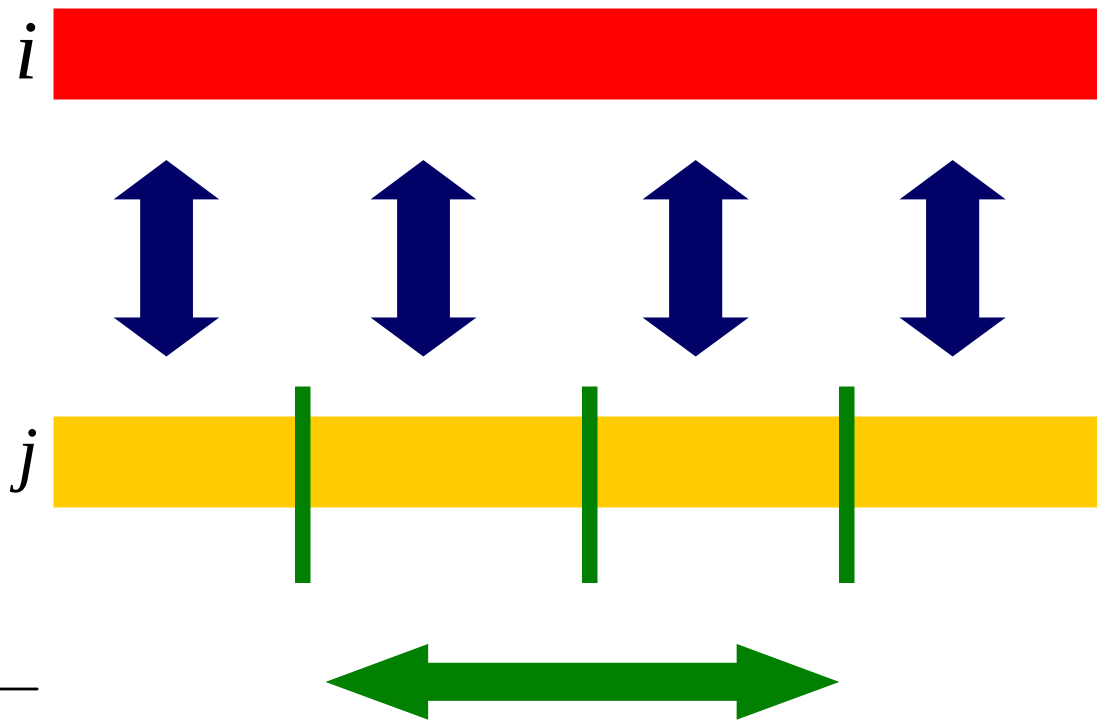
Basic idea of any parallel N-body code

$$N_{loc} = \frac{N}{N_{proc}}$$



Basic idea of any parallel N-body code

$j-particle$

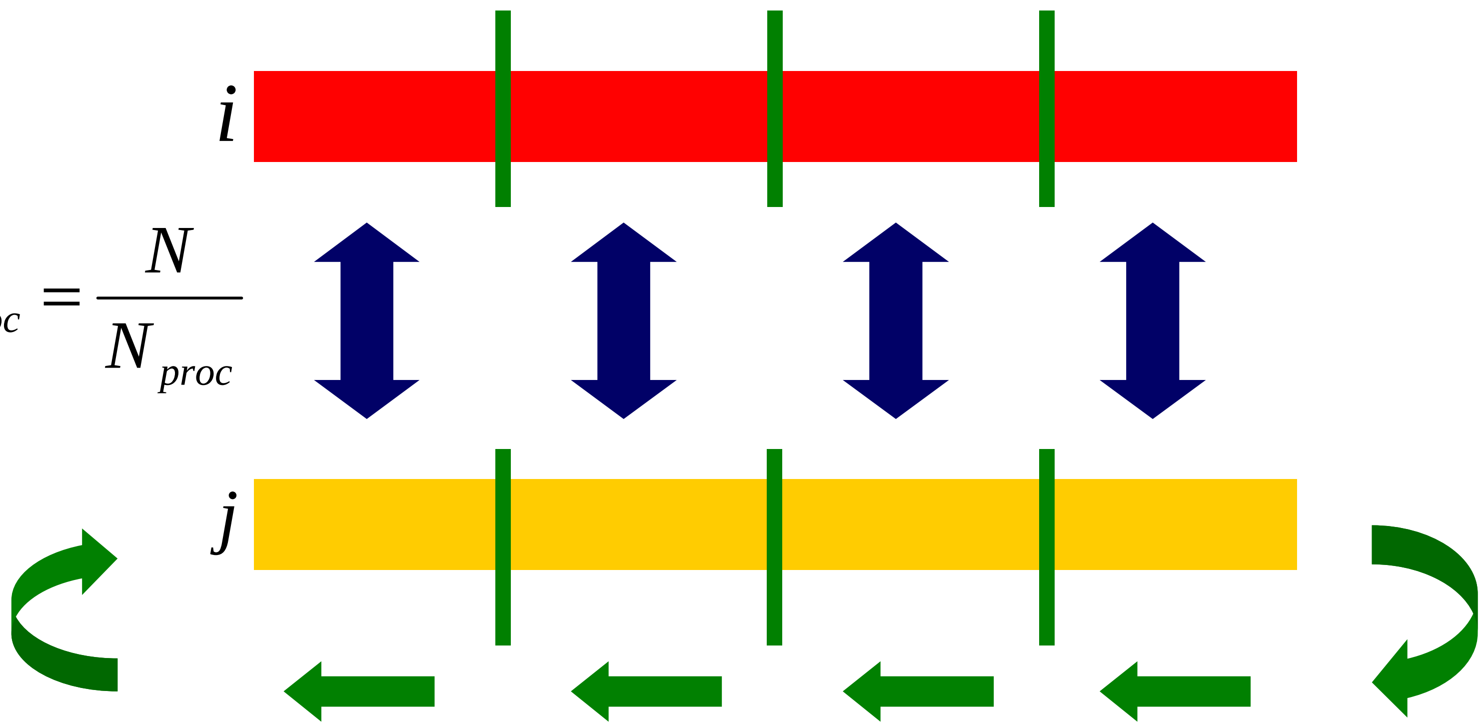


$$N_{loc} = \frac{N}{N_{proc}}$$

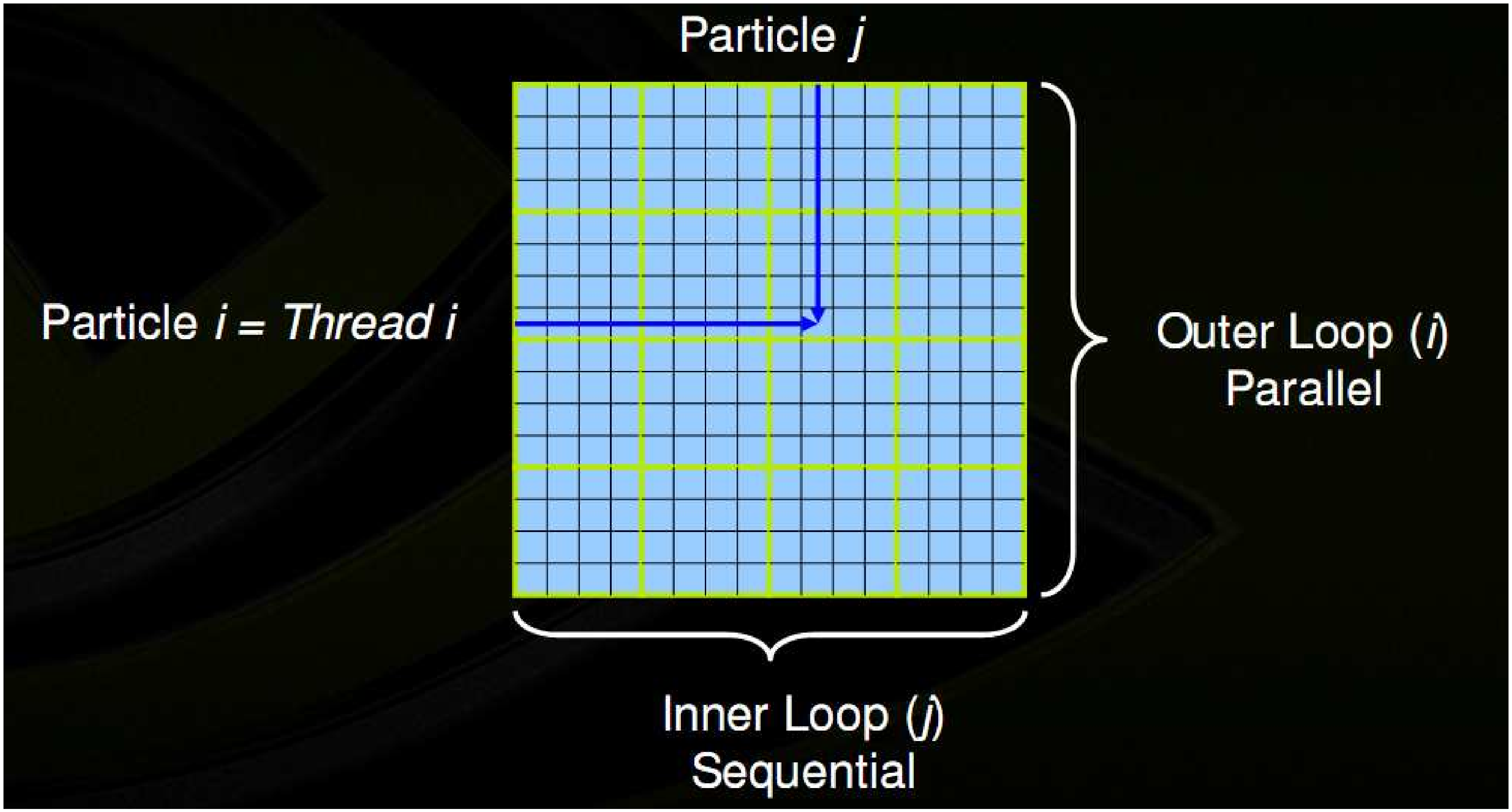
Basic idea of any parallel N-body code

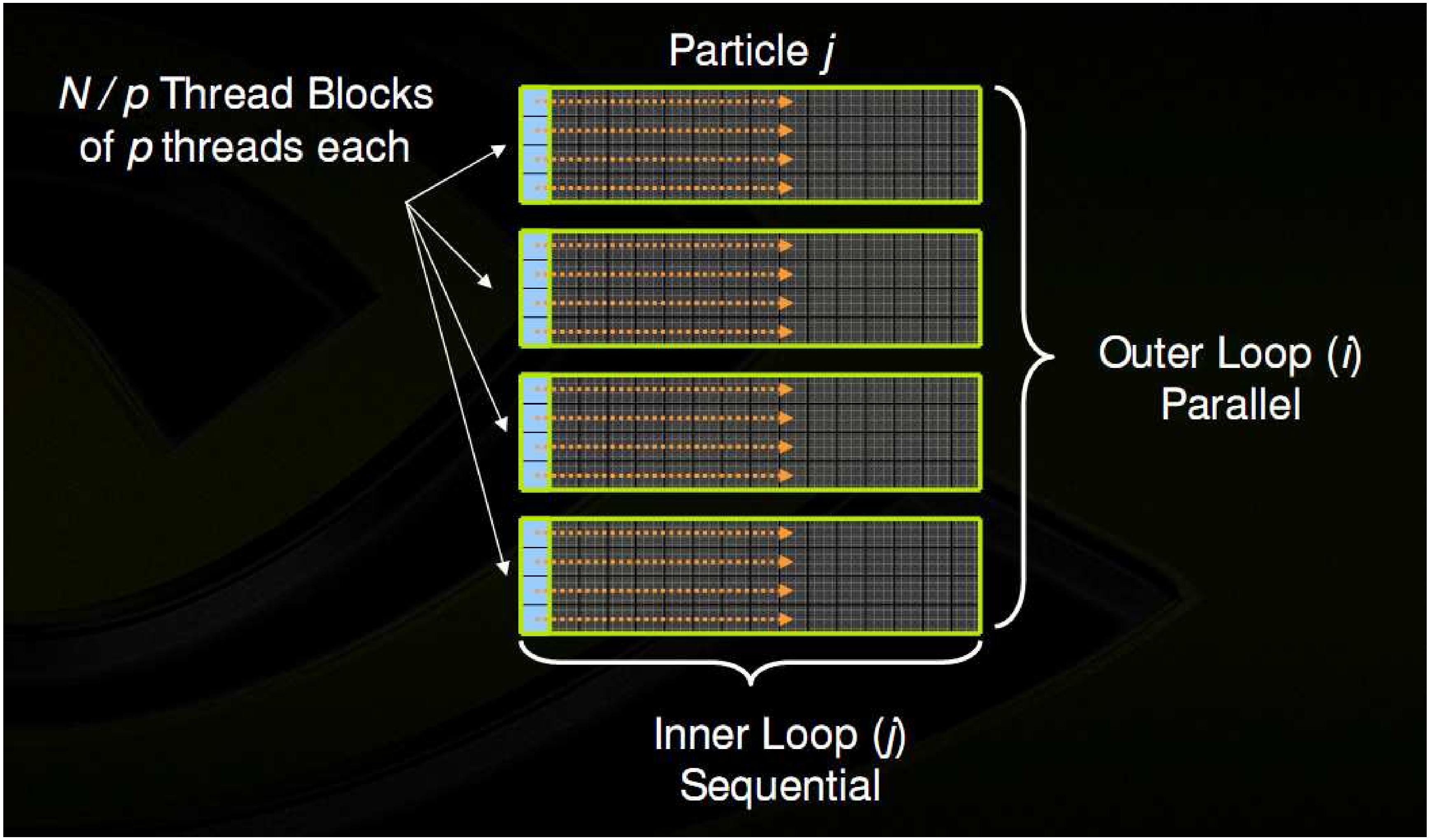
$$i, j - particle$$

$$N_{loc} = \frac{N}{N_{proc}}$$



Some communication scheme...

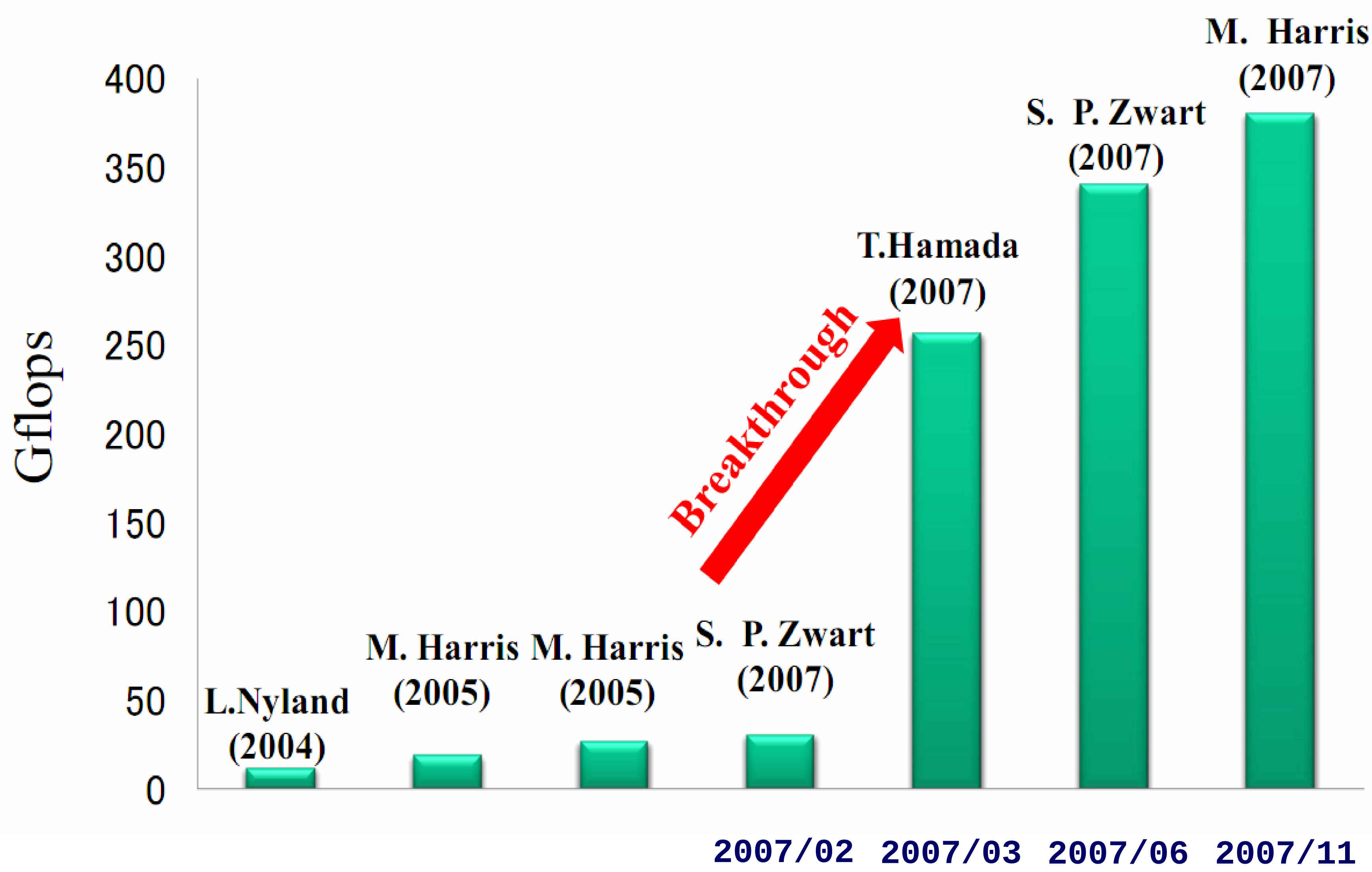

Particle *j*
Particle *i* = *Thread i*
Outer Loop (*i*)
Parallel
Inner Loop (*j*)
Sequential


Particle *j*
N / *p* Thread Blocks
of *p* threads each
Outer Loop (*i*)
Parallel
Inner Loop (*j*)
Sequential

```
forall bodies i in parallel {
  accel = 0;
  pos = position[i]
  foreach tile q {
    forall threads p in thread block in parallel {
      shared[p] = position[q*tile_size + p]
    }
    synchronize threads in block
    foreach body j in tile q {
      accel +=
        computeAcceleration(pos, position[j])
    }
    synchronize threads in block
  }
}
```

GPU N-body speedup timeline

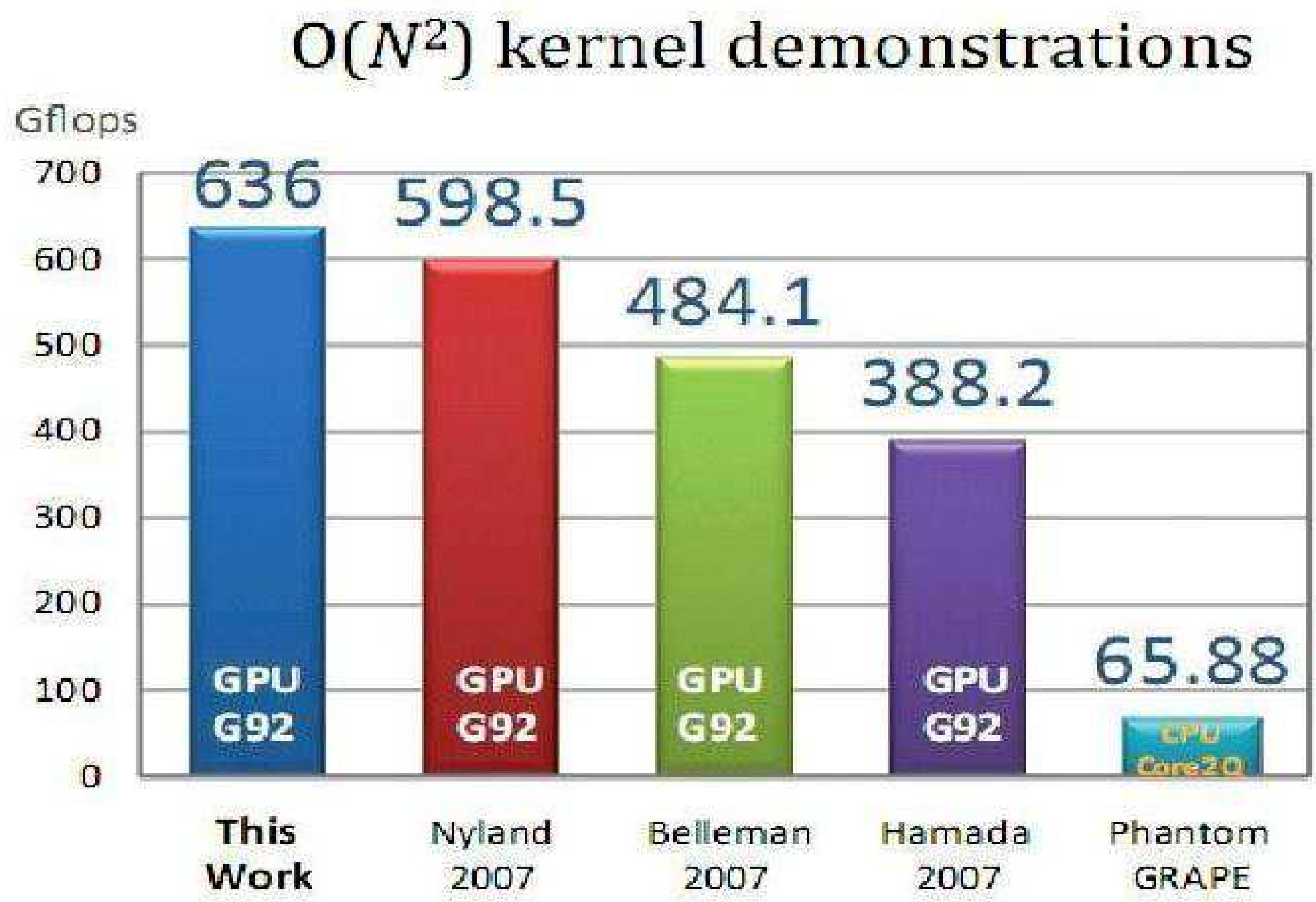
All on a same GPU: 8800 GTX (G80)



GPU N-body gravity

Hamada et al. 2008: Direct GPU code

GRAPE6a

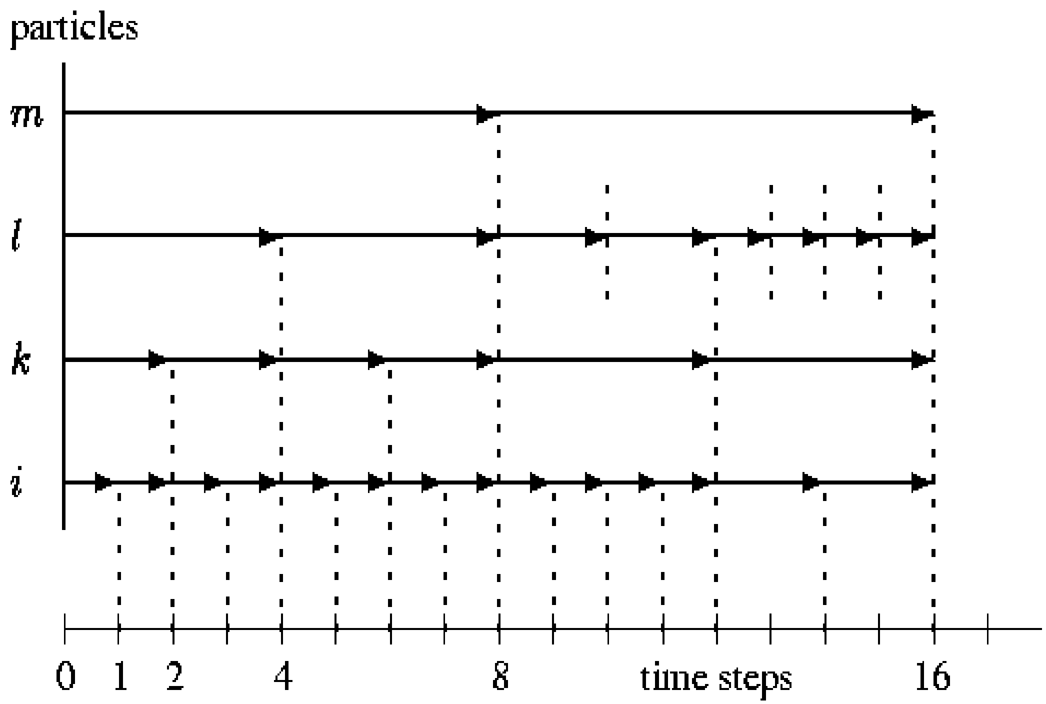


Results for 8800 GTS 512MB (G92)

Our own GRAPE/GPU N-body code

Harfst et al, NewA, 12, 357 (2007) [astro-ph/0608125]

Hierarchical Individual Block Time Steps



$$\Delta t = \sqrt{\eta \frac{|\vec{a}||\vec{a}^{(2)}| + |\dot{\vec{a}}|^2}{|\dot{\vec{a}}||\vec{a}^{(3)}| + |\vec{a}^{(2)}|^2}}\ .$$

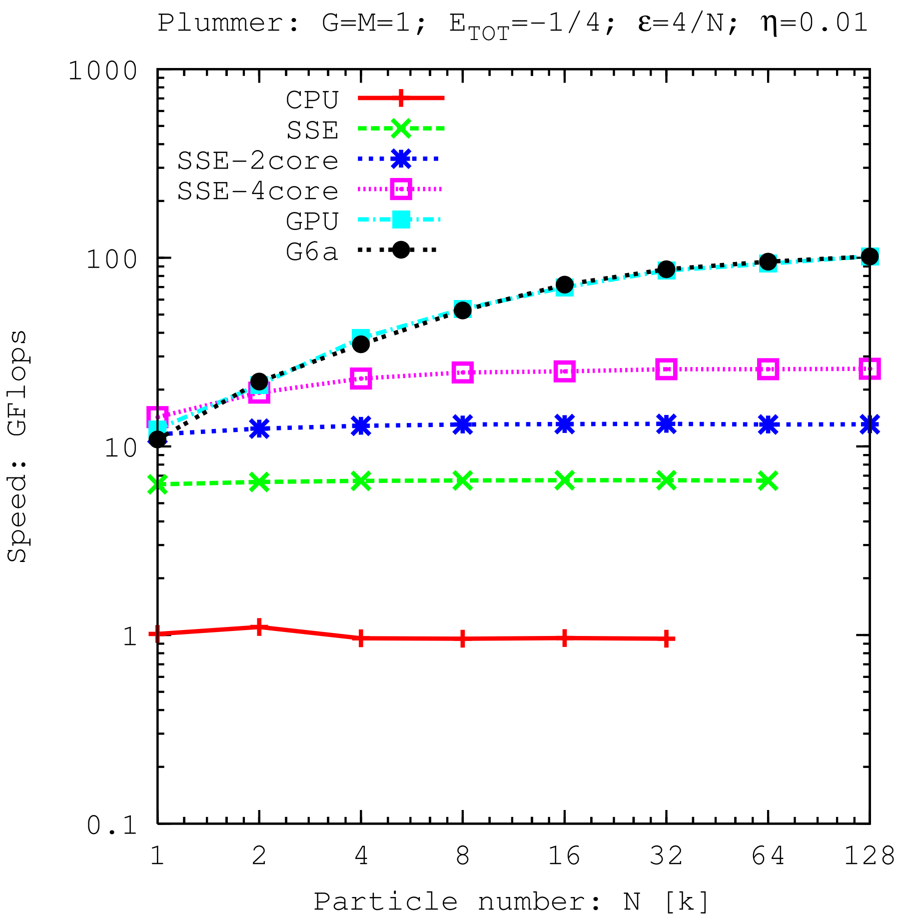
4th order Hermite scheme

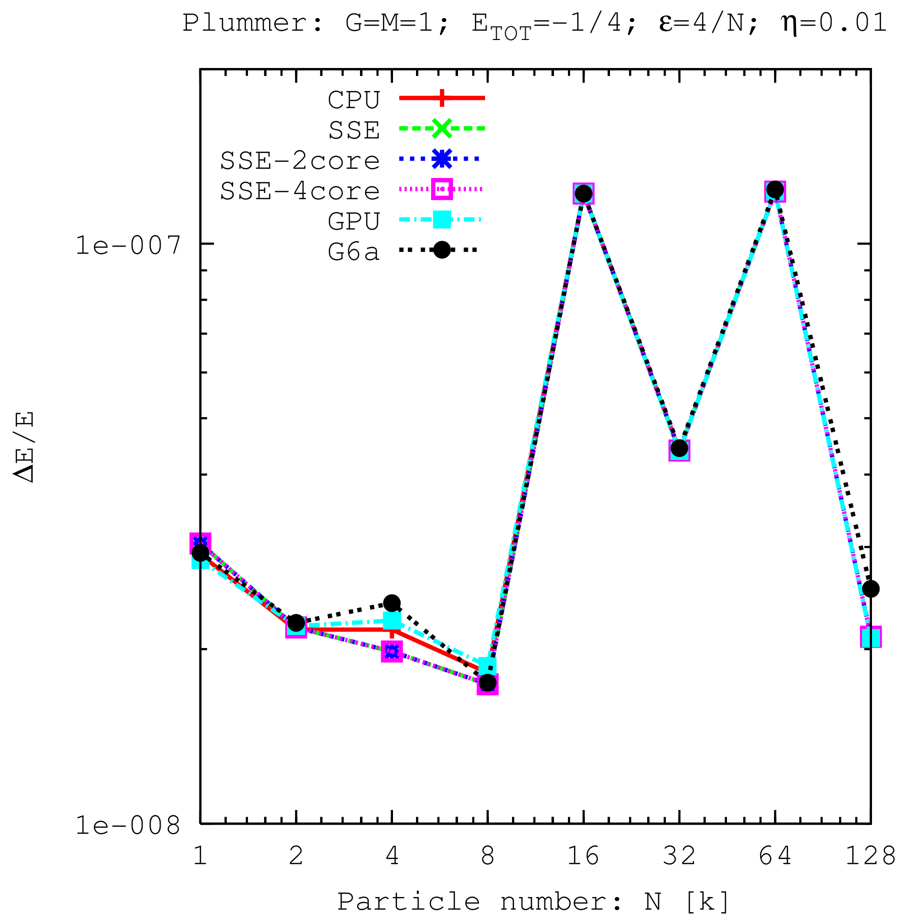
$$\frac{d^2\vec{r}_i}{dt^2} = \vec{a}_i$$

ftp://ftp.ari.uni-heidelberg.de/pub/staff/berczik/phi-GRAPE/

GPU results

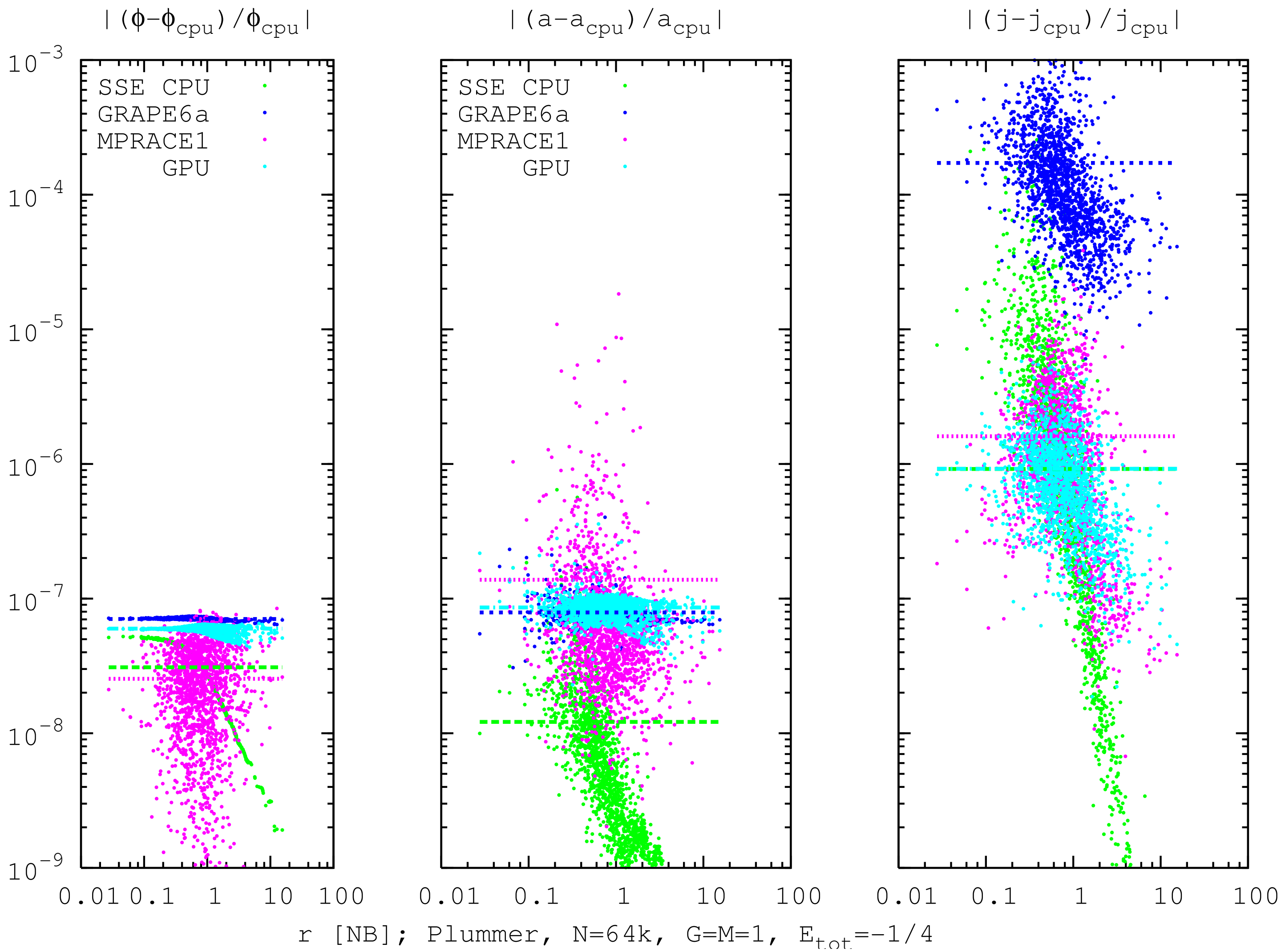
Nitadori, Berczik et al. 2007.11



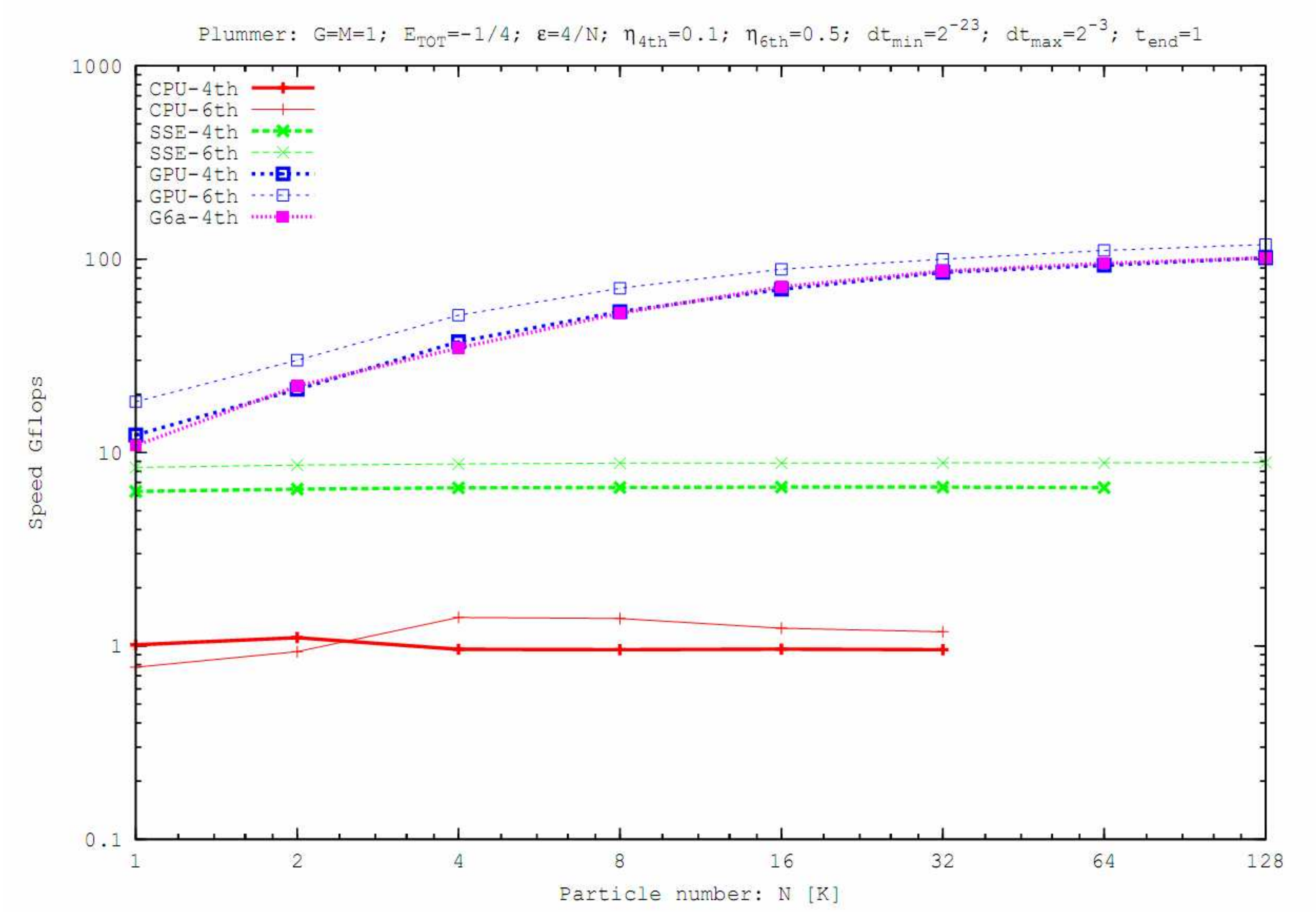


GPU results

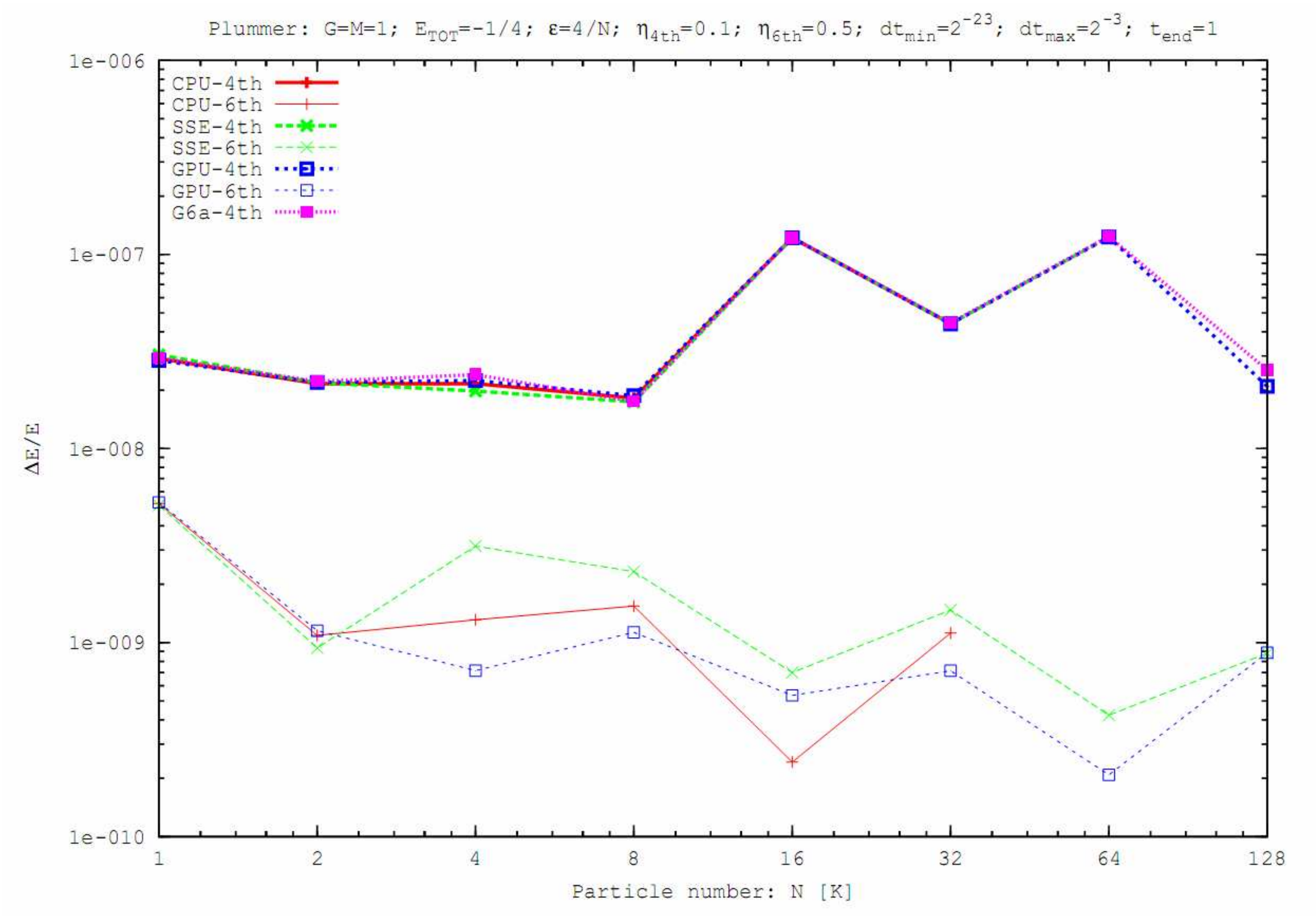
Nitadori, Berczik et al. 2007.11



GPU 4th vs. 6th order results:

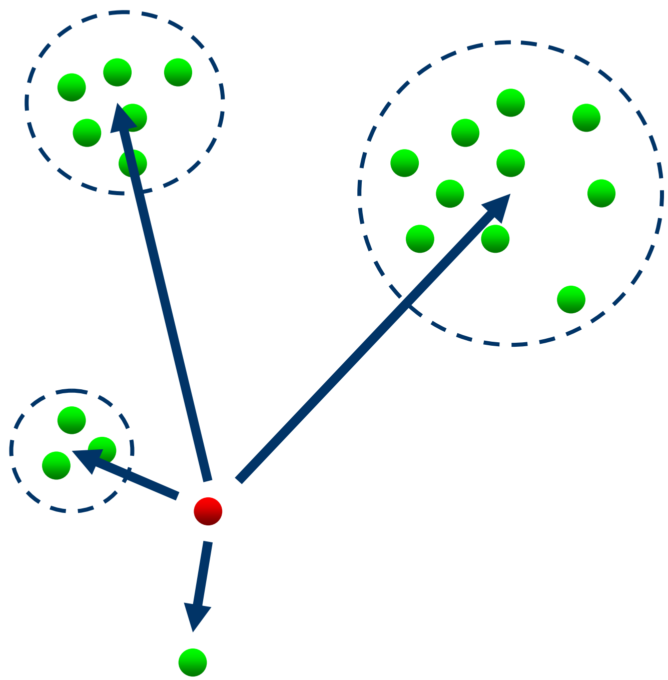


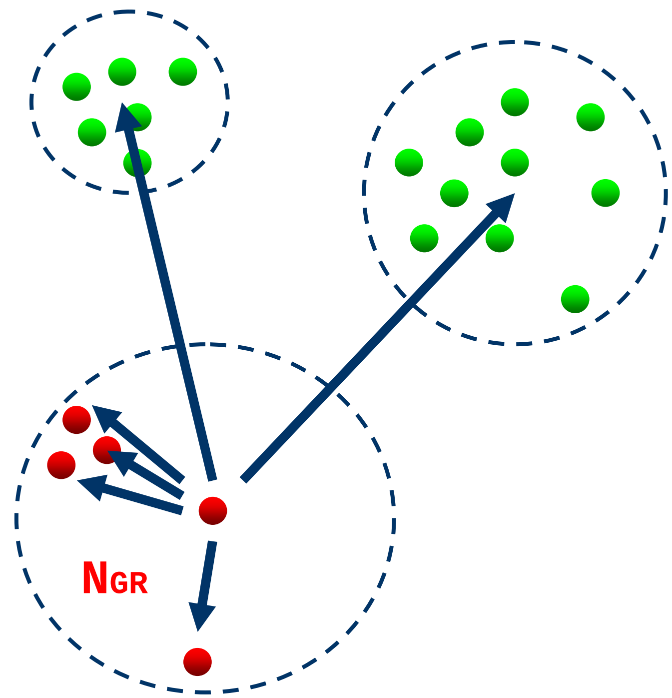
GPU 4^{th} vs. 6^{th} order results:



Parallel TREE GPU gravity

Jun Makino: TREE+GRAPE/GPU code





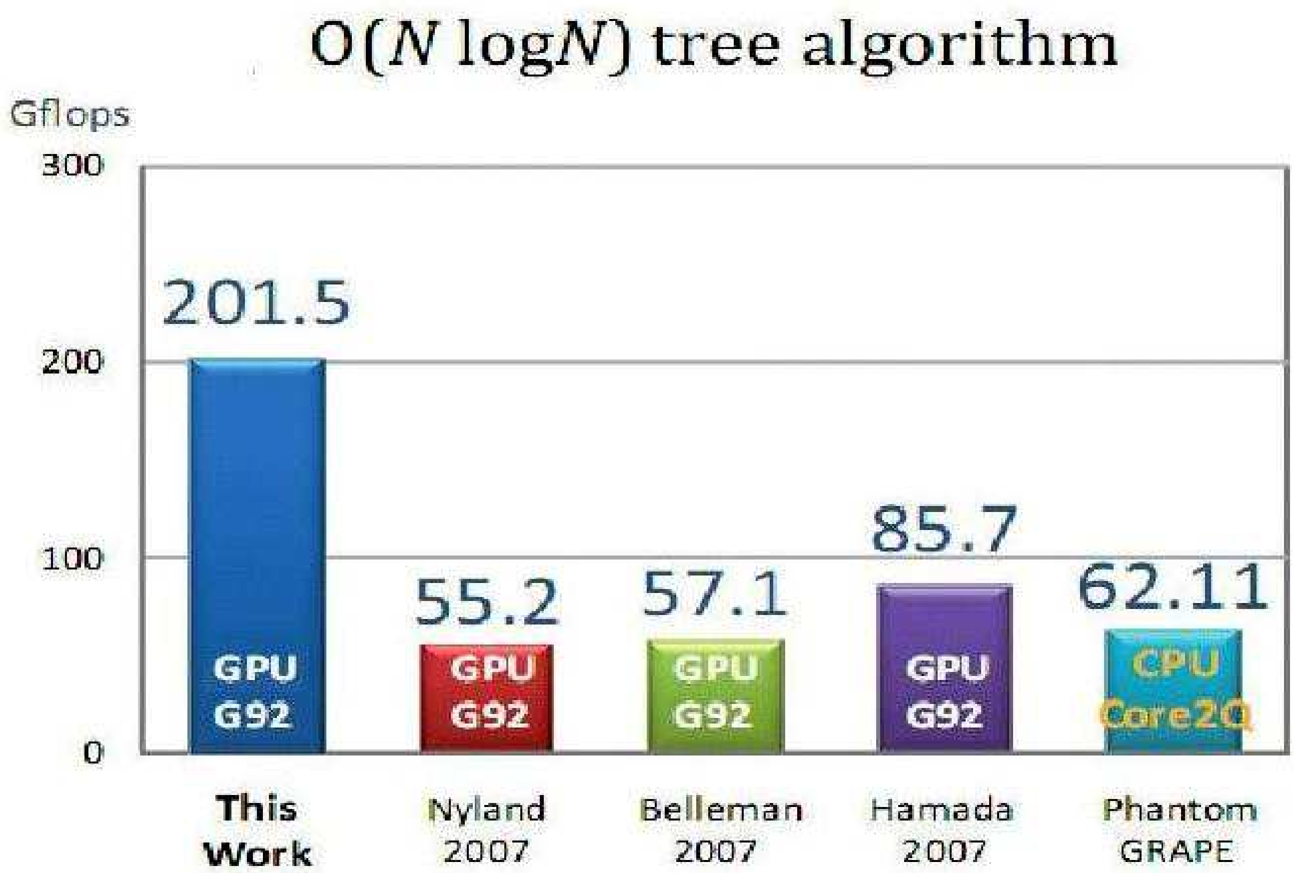
Makino, PASJ, 43, 621 (1991)

Inter. list on host ~N
Inter. list length -> short…

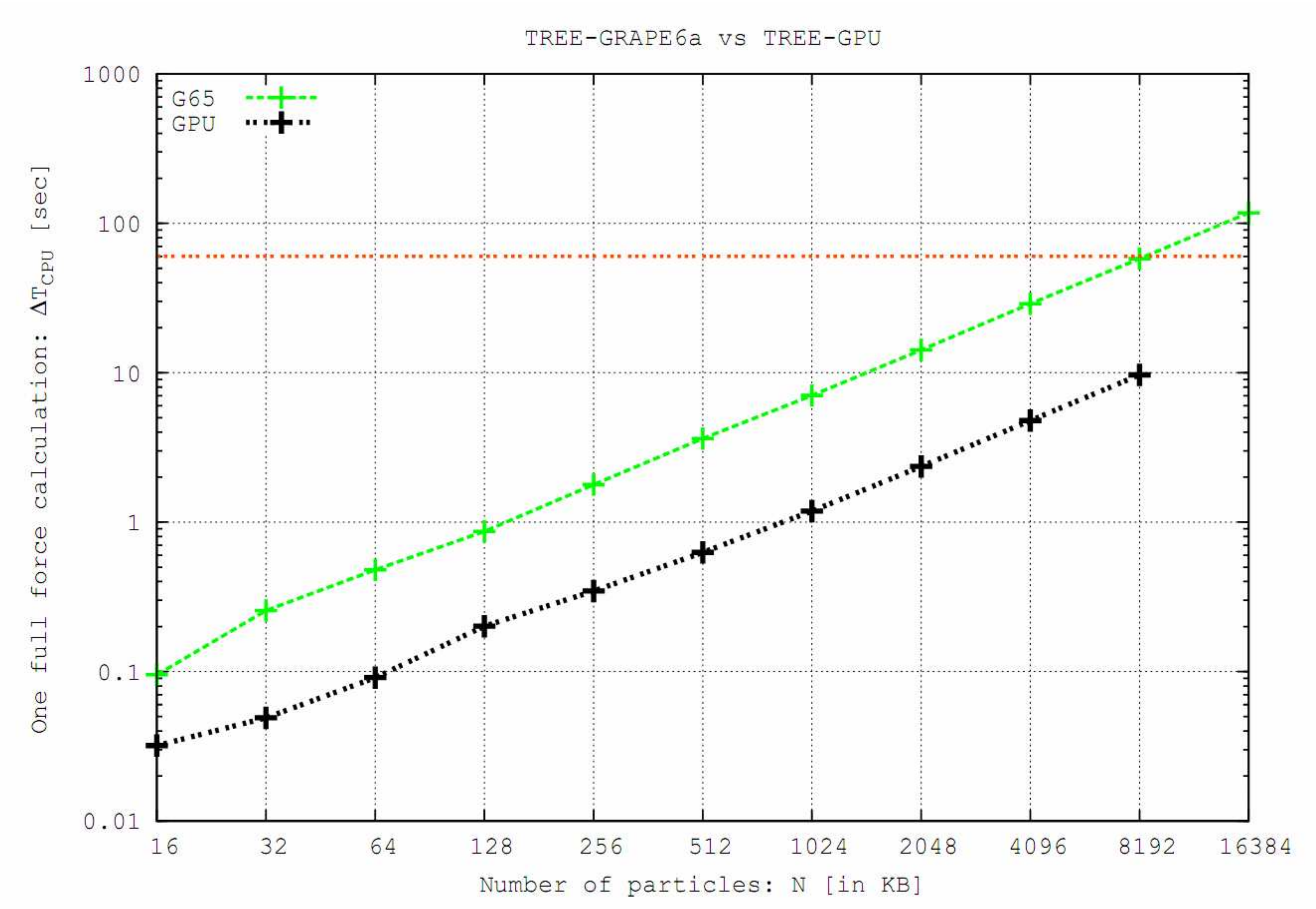
Makino, PASJ, 56, 521 (2004)
Fukushige, Makino & Kawai, PASJ, 57, 1009 (2005)
One interaction list is shared among NGR particles!
Inter. list on host ~N/NGR
Inter. list length -> larger…

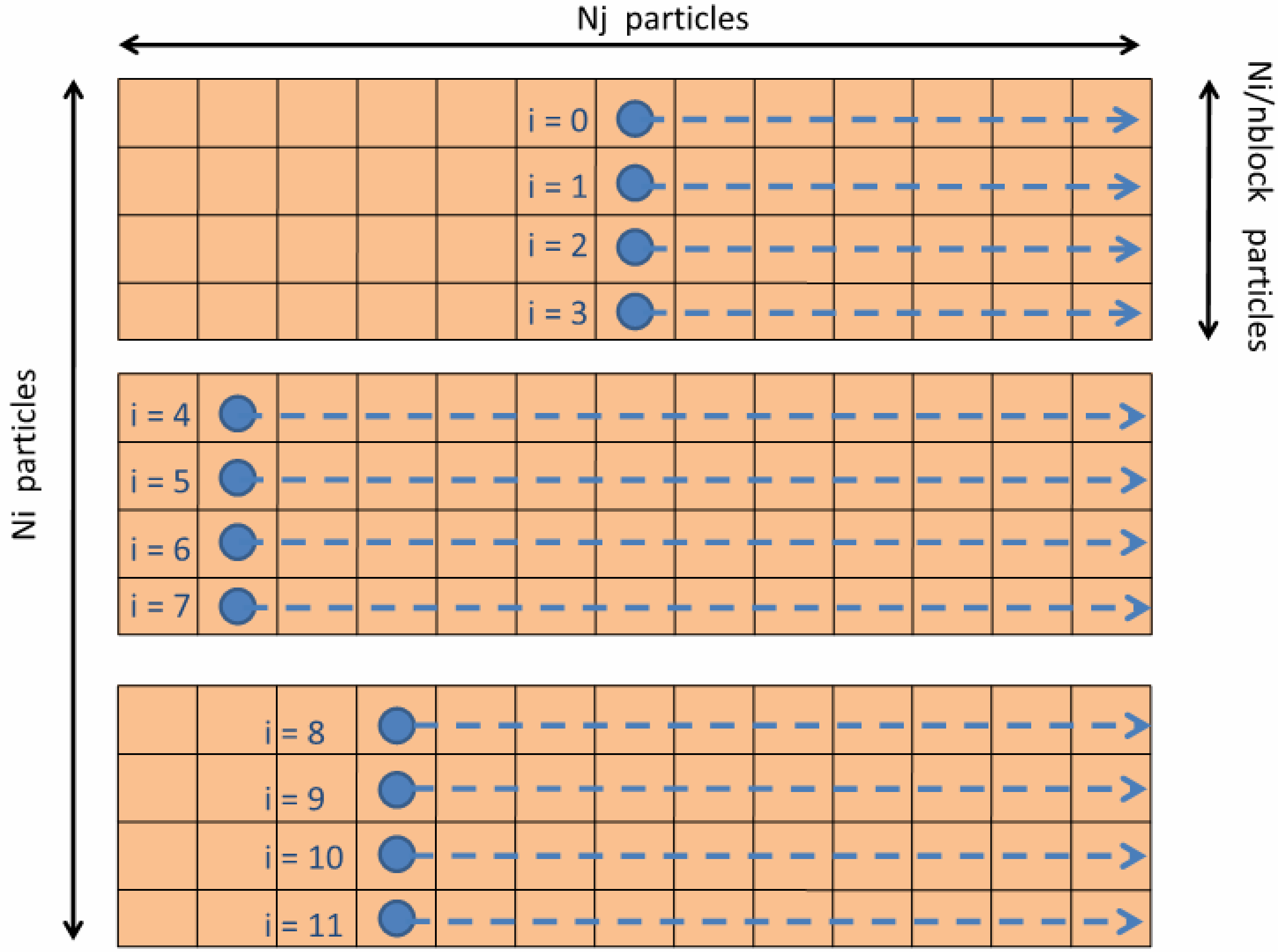
Parallel TREE GPU gravity

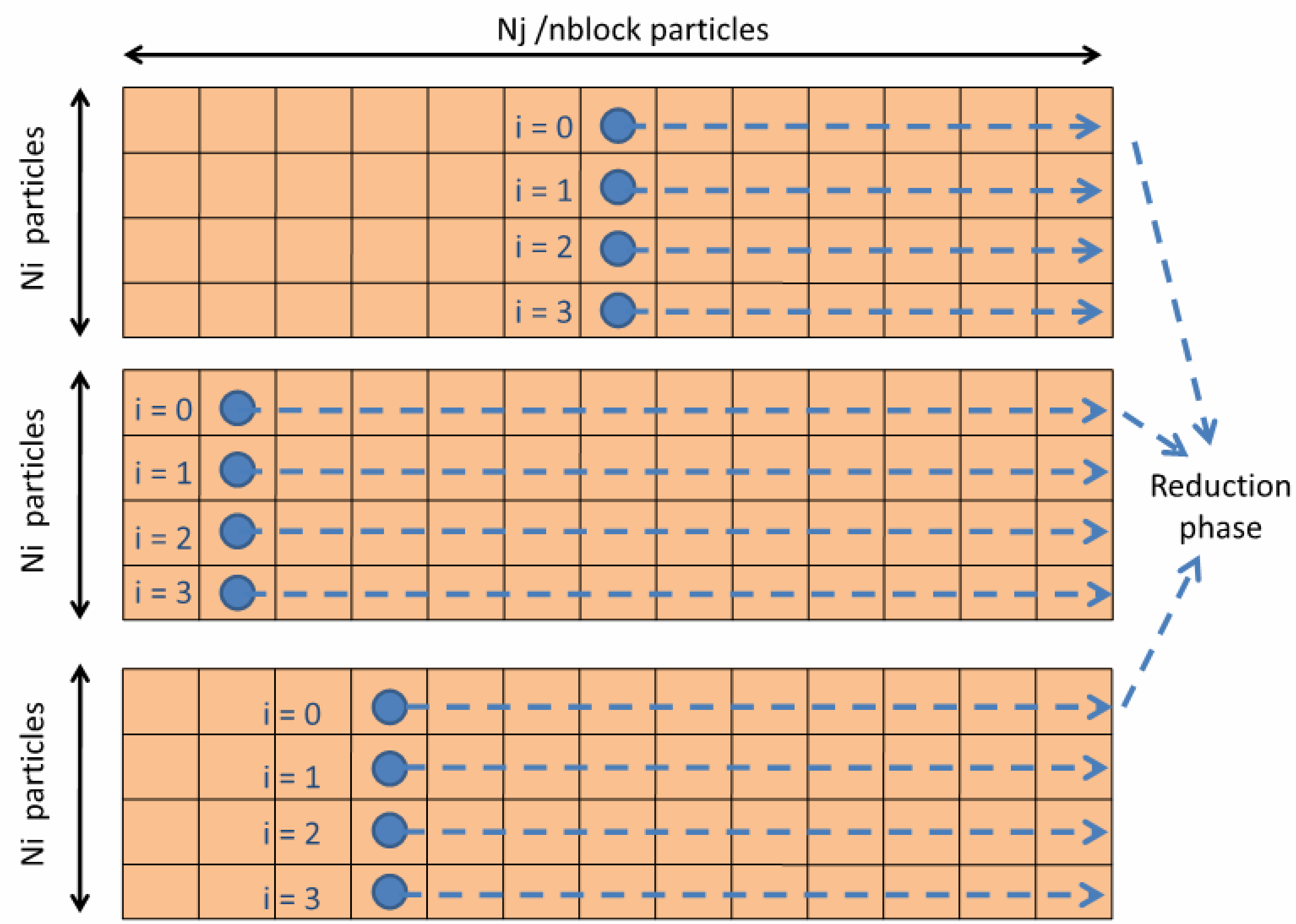
Hamada et al. 2008: TREE+GRAPE/GPU code



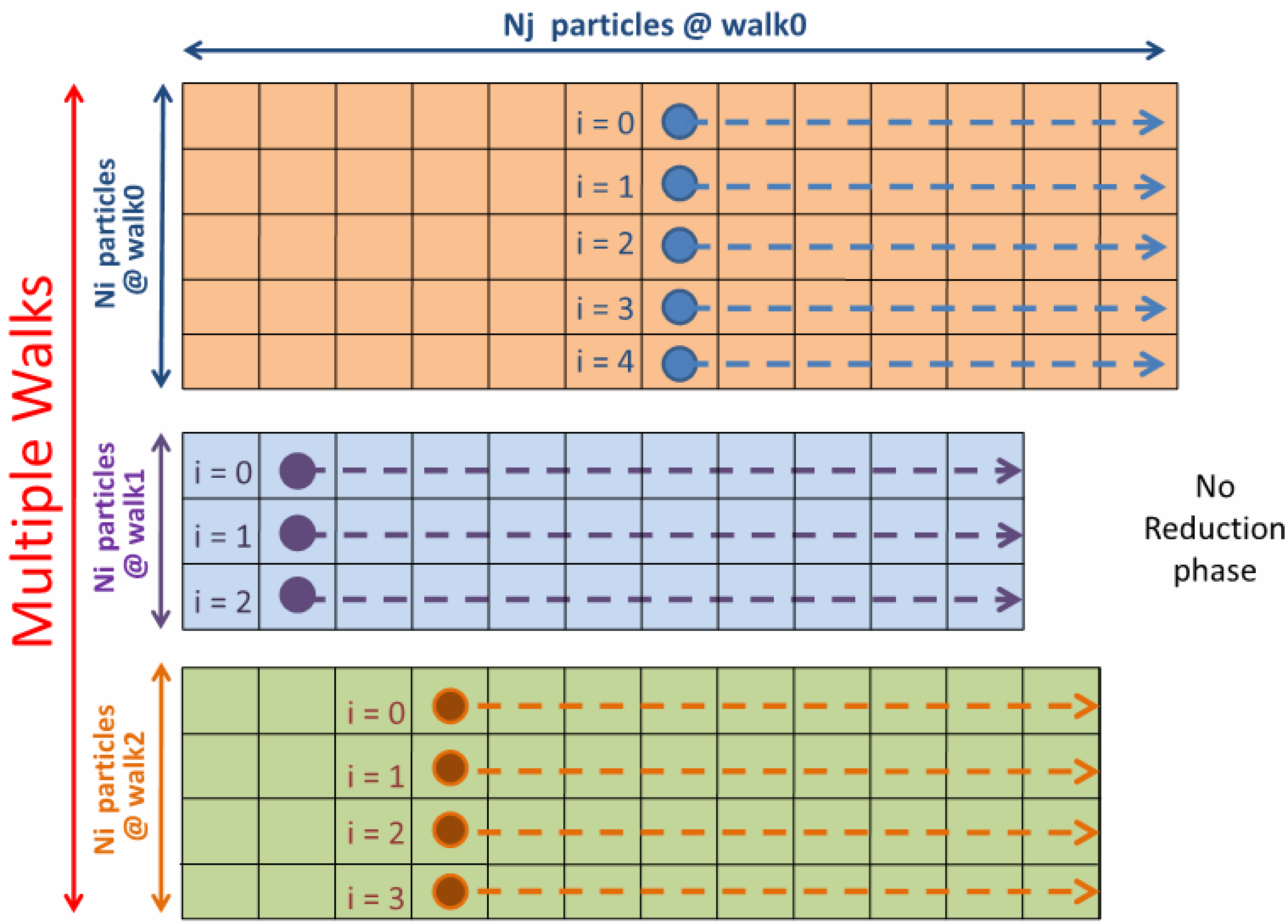
Parallel TREE GPU gravity

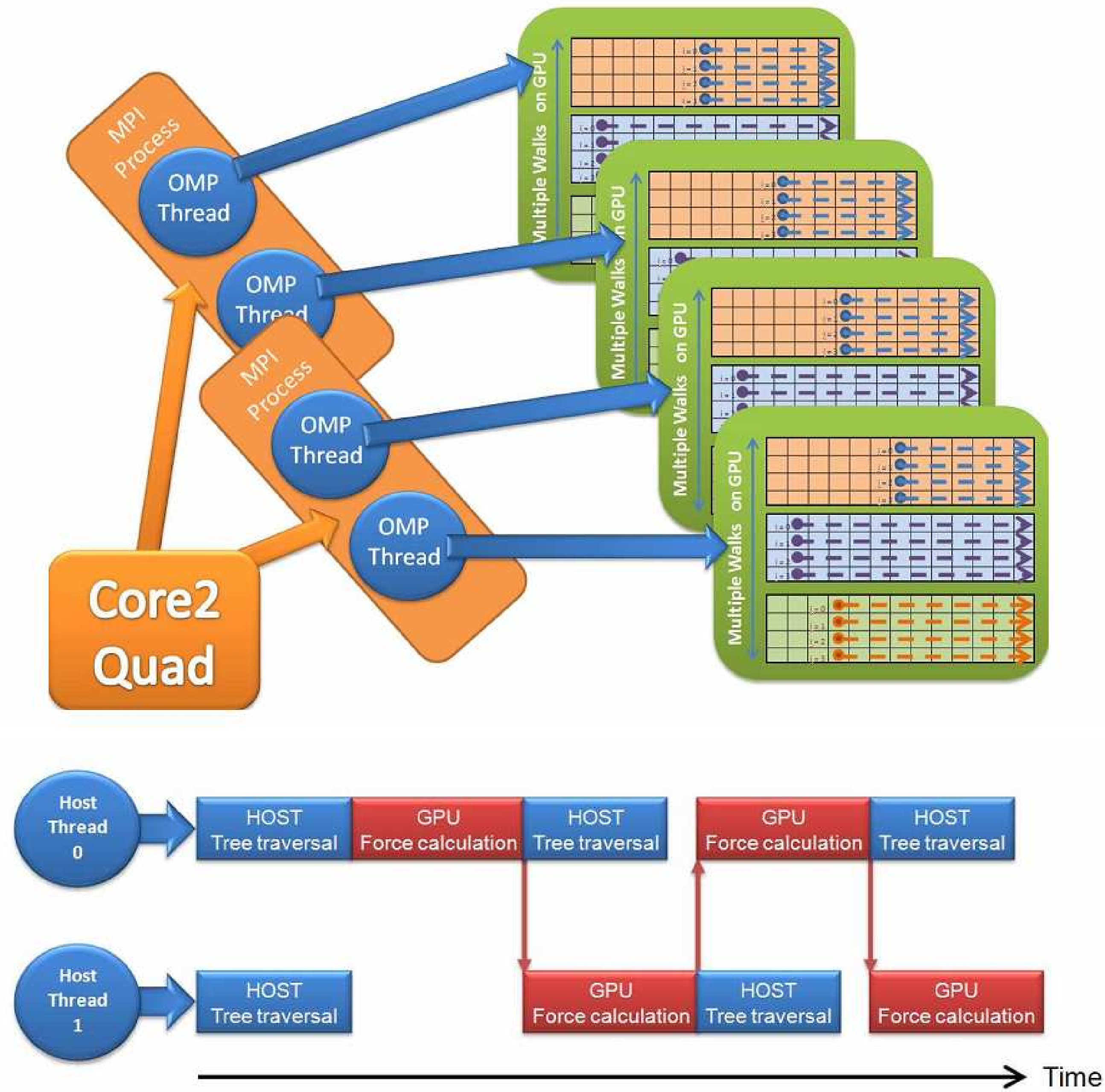



Nj particles
Ni/nblock particles
Ni particles
i = 0
i = 1
i = 2
i = 3
i = 4
i = 5
i = 6
i = 7
i = 8
i = 9
i = 10
i = 11



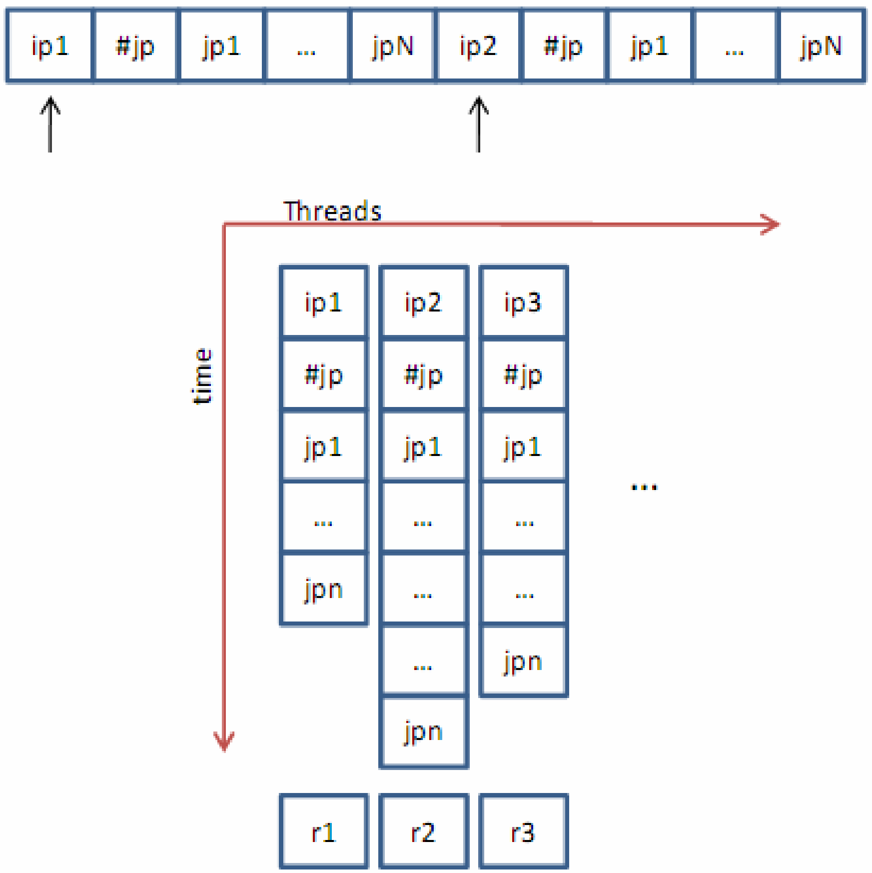
Nj /nblock particles
Ni particles
i = 0
i = 1
i = 2
i = 3
Ni particles
i = 0
i = 1
i = 2
i = 3
Ni particles
i = 0
i = 1
i = 2
i = 3
Reduction phase

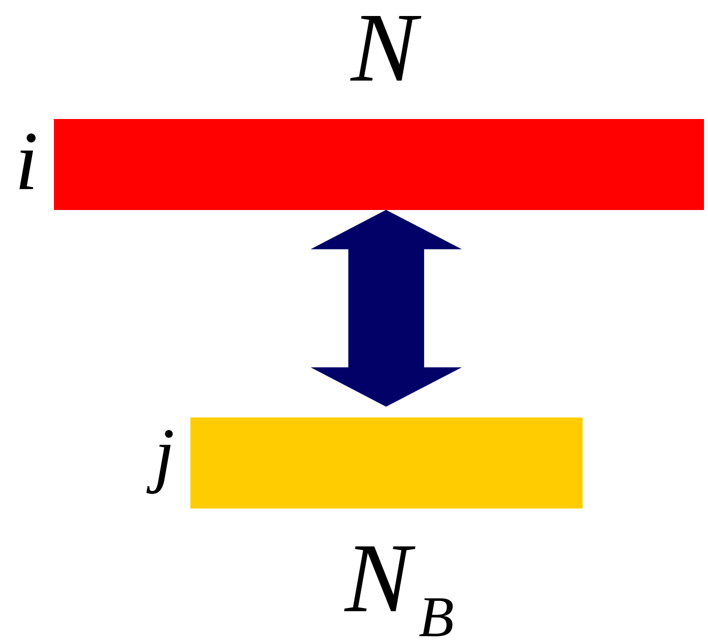

Nj particles @ walk0
Multiple Walks
Ni particles @ walk0
i = 0
i = 1
i = 2
i = 3
i = 4
Ni particles @ walk1
i = 0
i = 1
i = 2
No Reduction phase
Ni particles @ walk2
i = 0
i = 1
i = 2
i = 3

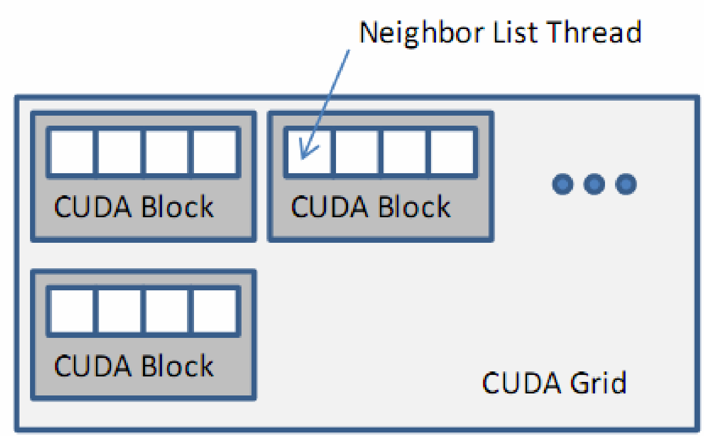


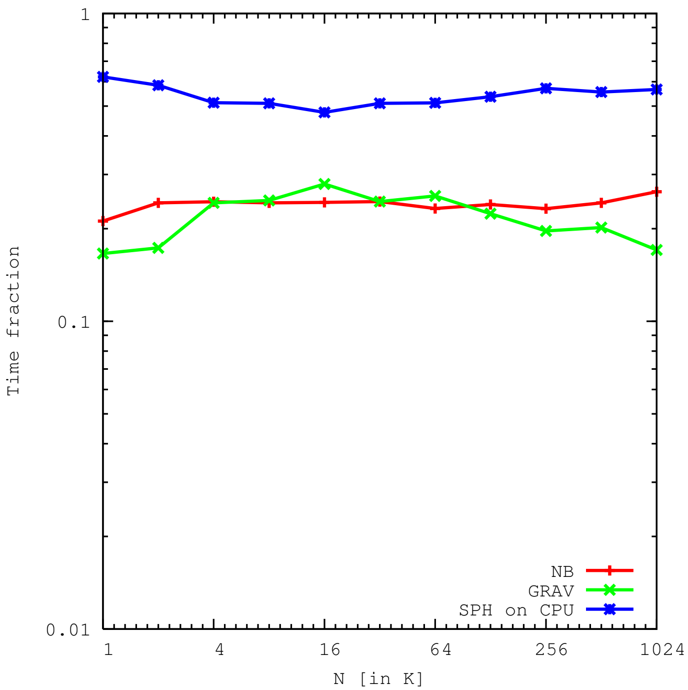
MPI Process
OMP Thread
OMP Thread
MPI Process
OMP Thread
OMP Thread
Core2 Quad
Multiple Walks on GPU
Host Thread 0
HOST Tree traversal
GPU Force calculation
HOST Tree traversal
GPU Force calculation
HOST Tree traversal
Host Thread 1
HOST Tree traversal
GPU Force calculation
HOST Tree traversal
GPU Force calculation
Time

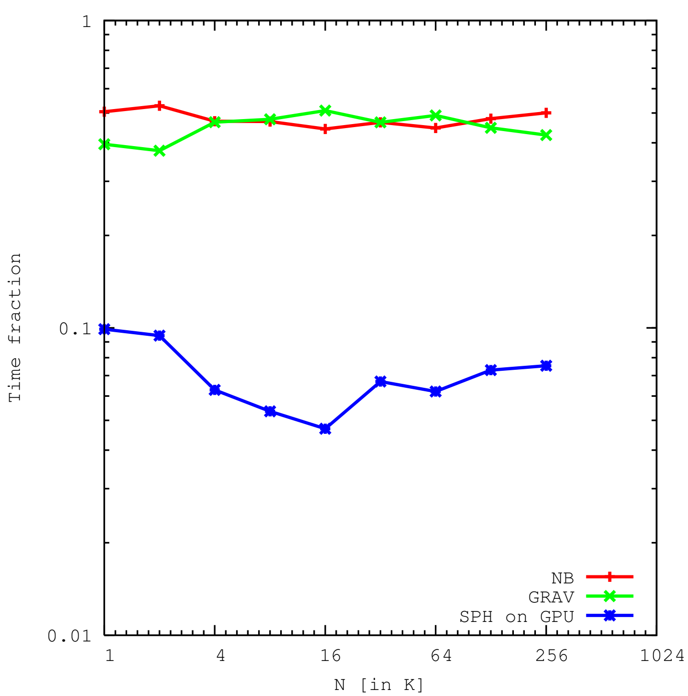
Simple GPU SPH code



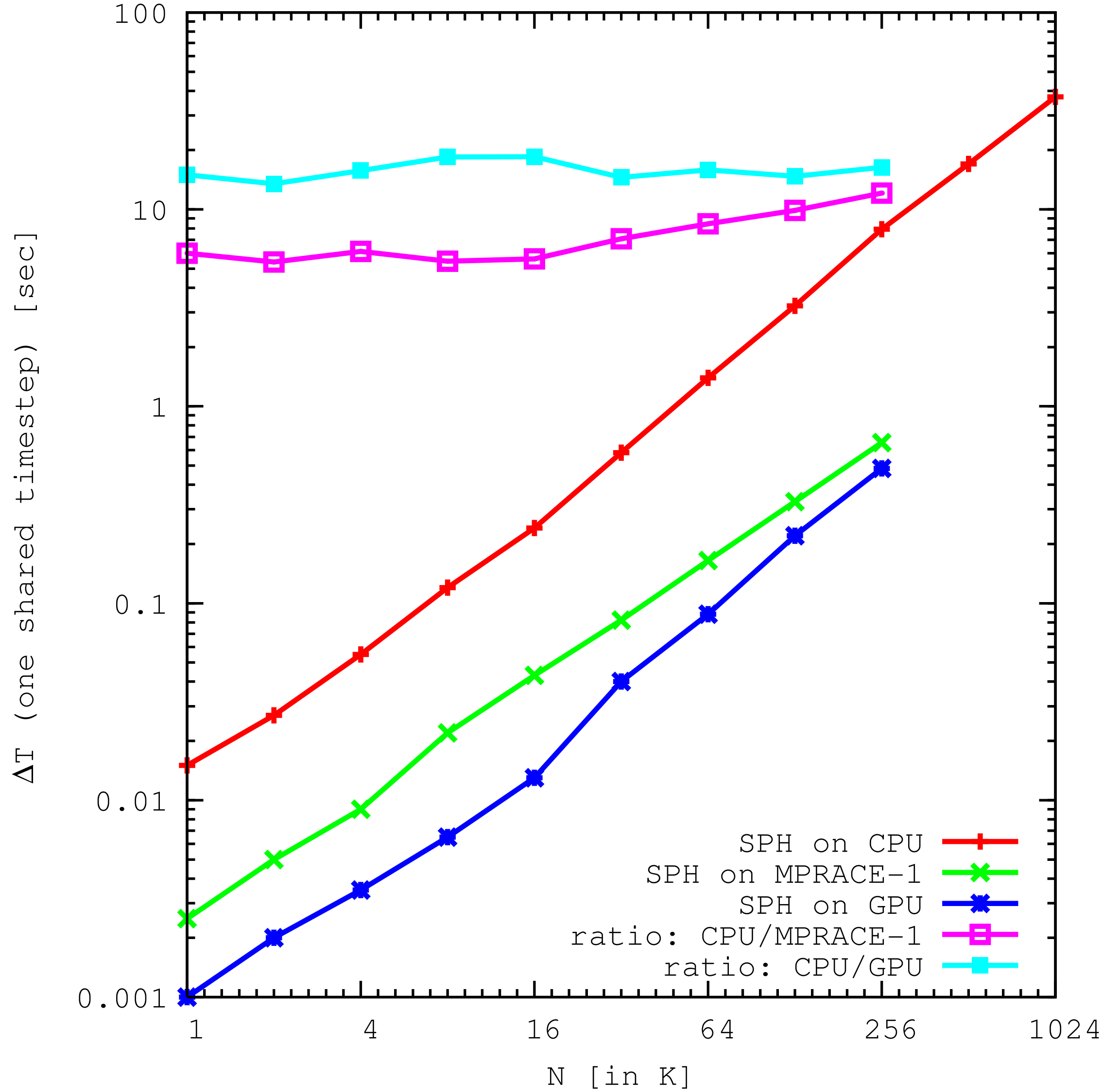








SPH speedup with GPU



SPH astrophysical results:

TREE-GRAPE + MPRACE (on 4 nodes)

M = 2000 $M_\odot$ R = 3 pc   fully -> H_2

Isothermal evolution.

Initial density distr. ~1/r

T = 20 K (c_sound = 0.3 km/sec)

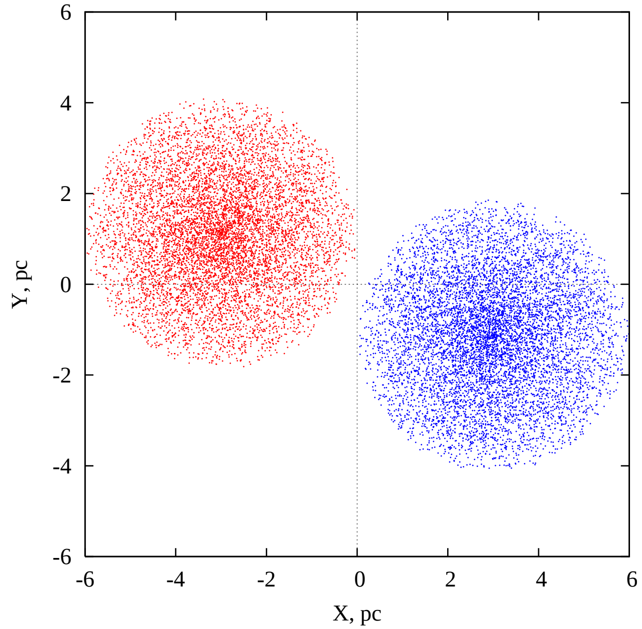
V_merge = 5 km/sec

Calculation time 3*t_ff = 6 Myr

Resolution is h_min = 1e-4 pc

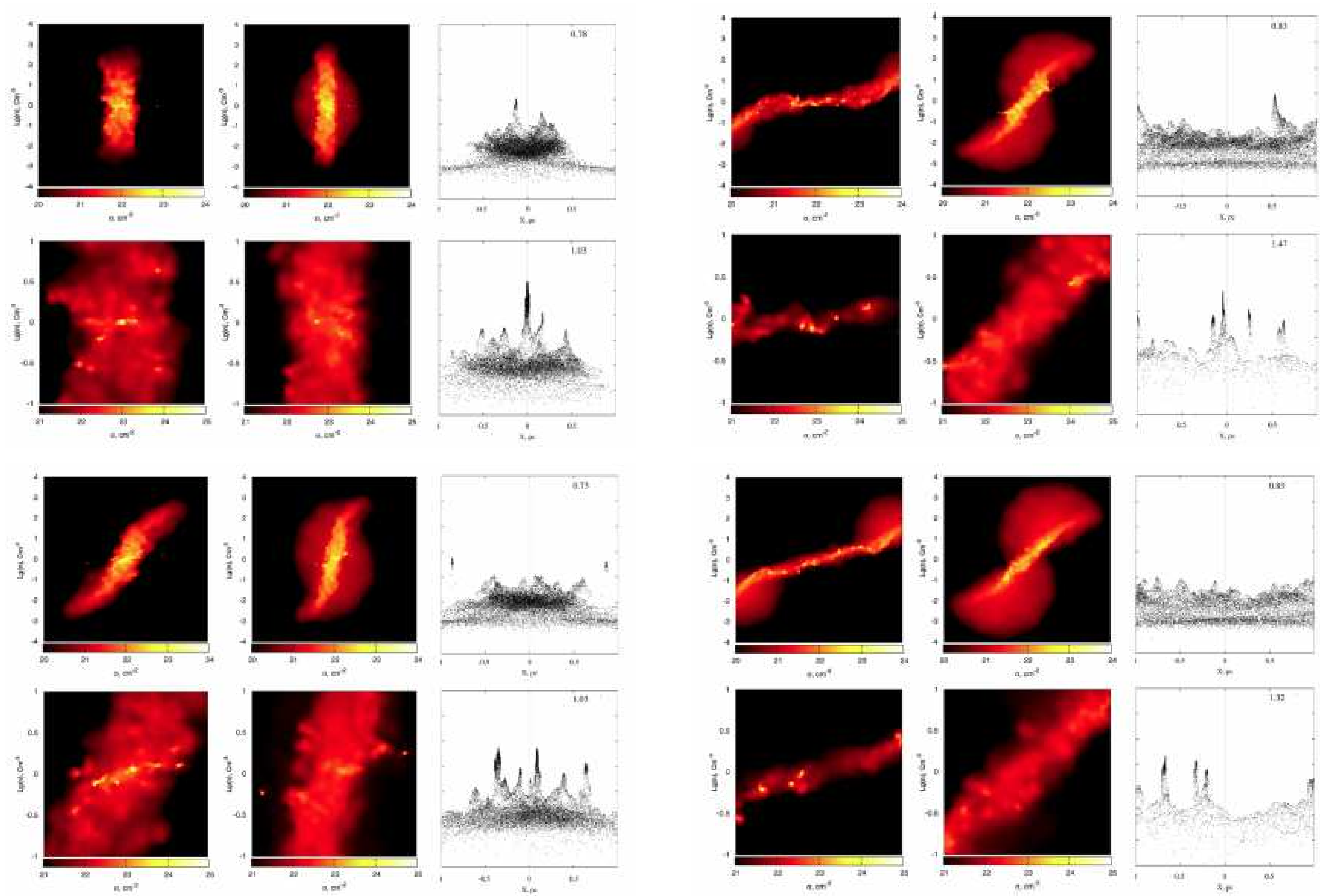
SPH MPRACE/CPU speedup ~10

Total GRAPE+MPRACE/CPU speedup ~15

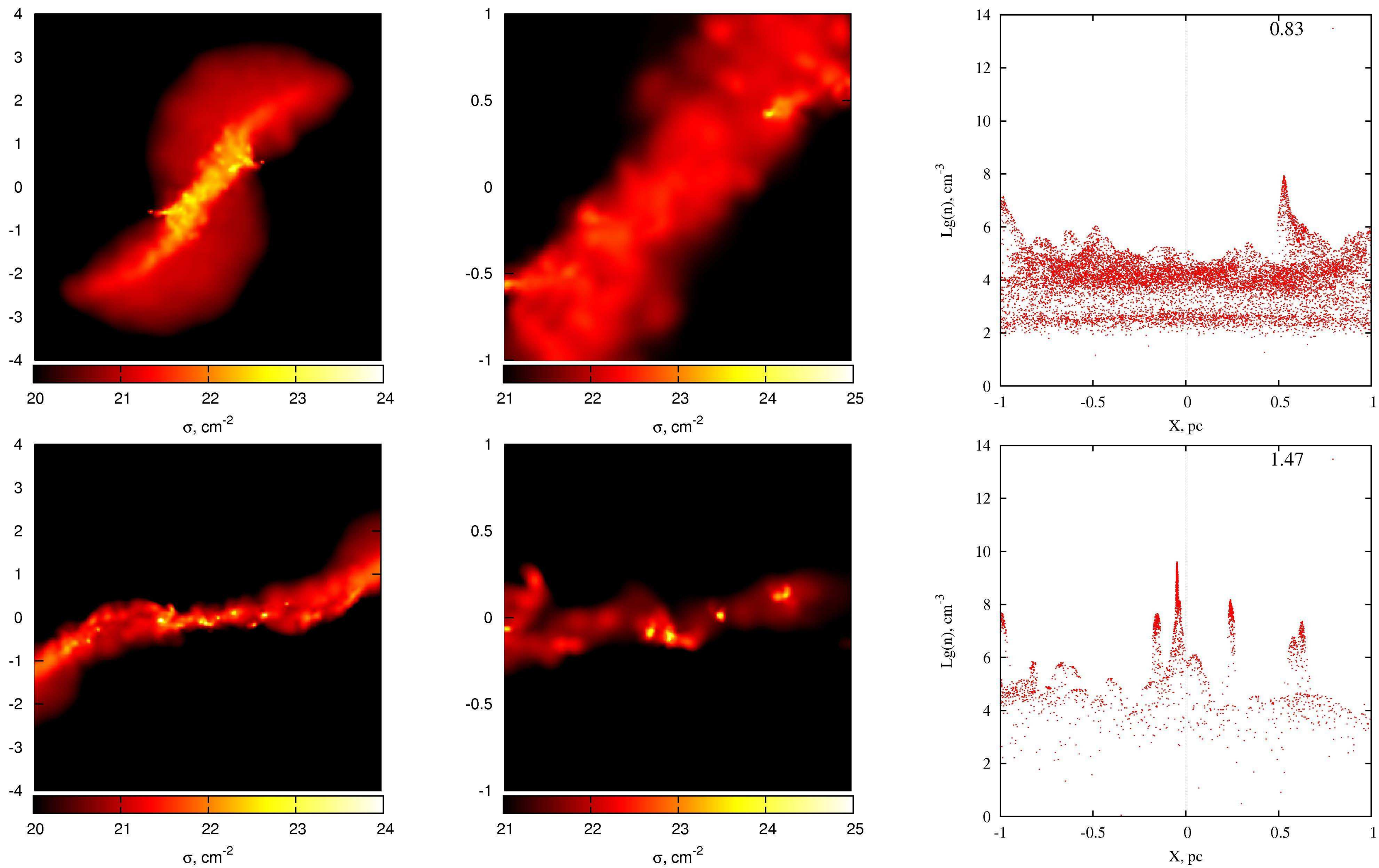


	N	DT_CPU
	N = 2x4k	DT_CPU = 52 min
	2x8k	1.74 hours
	2x16k	3.5 hours
	2x32k	6.9 hours
	2x64k	14 hours
*	2x128k	28 hours
	2x256k	55 hours
	2x512k	111 hours

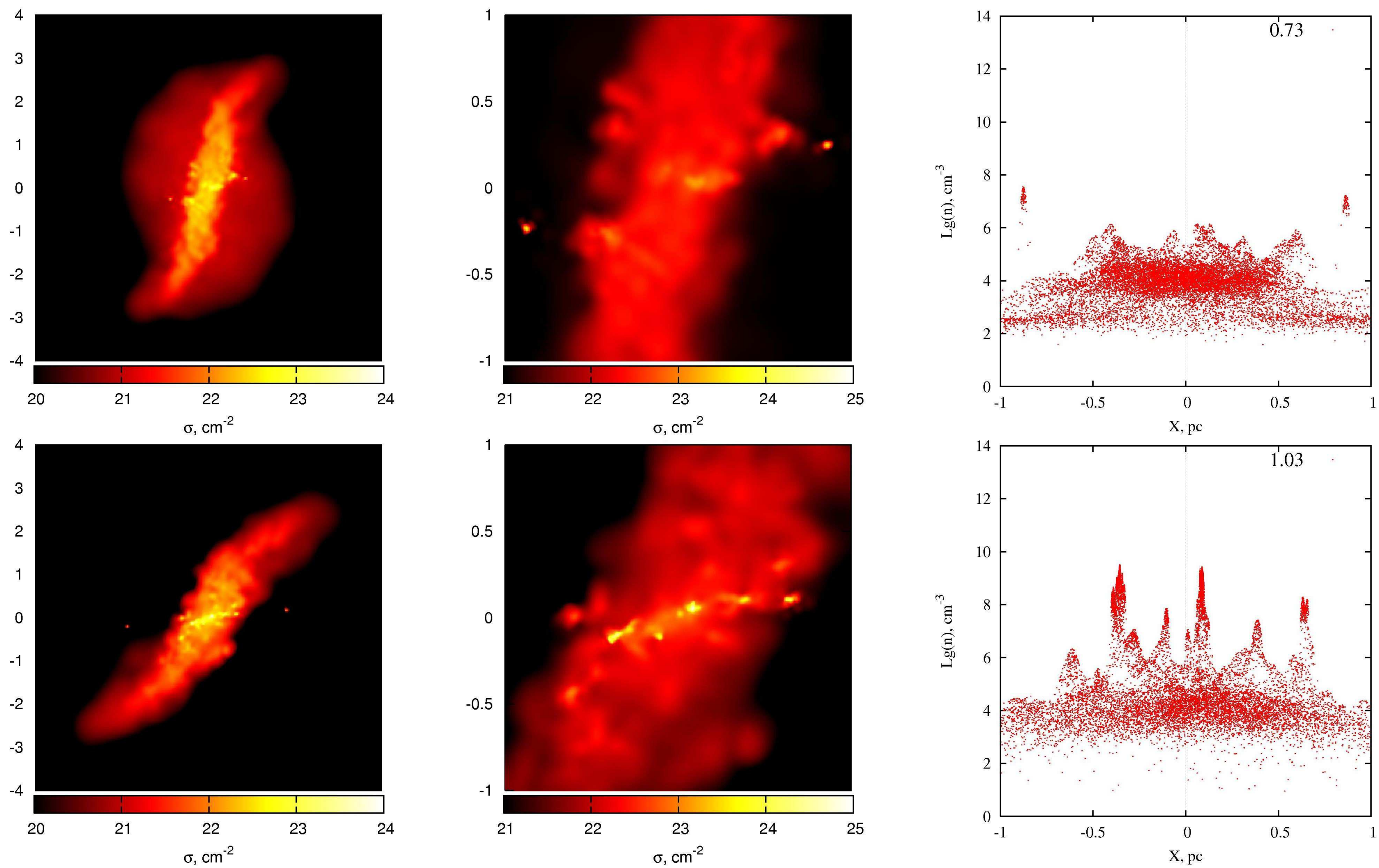
SPH astrophysical results:



SPH astrophysical results:



SPH astrophysical results:



SPH astrophysical results: