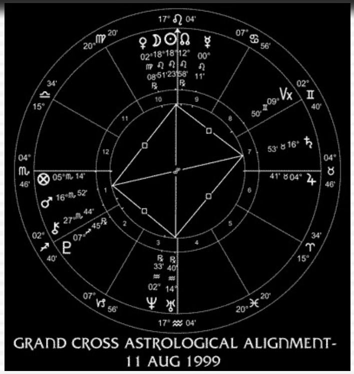
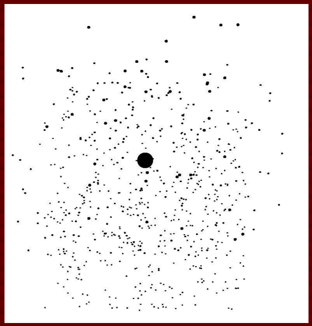
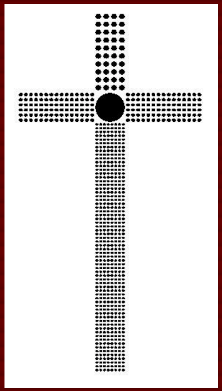
Nostradamus' Grand Cross



- N-body simulation





Presenter Name : **Li Dong, Hui Lin**
Title/Date : **May 7, 2015**

Rensselaer

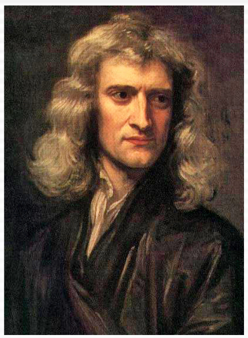
Rensselaer

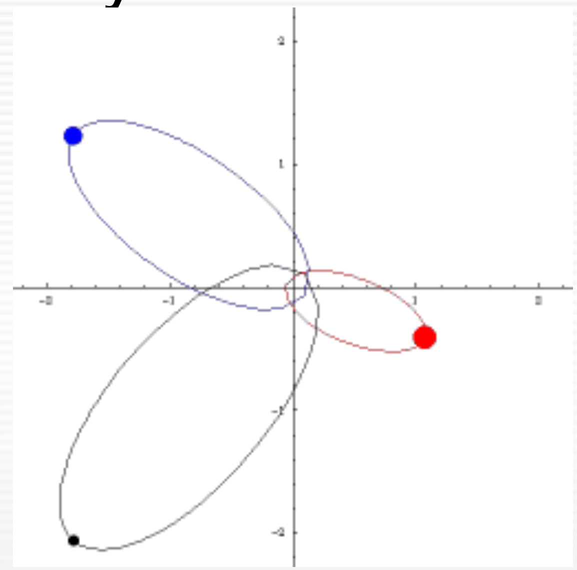
Outline

- Brief introduction to N-body problem
- Grand Cross evolution
  - Demo
  - How we did it?!
- Sequential implementation
  - Brutal force scheme vs. Barnes Hut algorithm
  - Implementation of Seq. Barnes Hut algorithm
- Parallelization with CUDA
  - Brutal force scheme vs. Barnes Hut algorithm
  - Speedup chart

Brief introduction to N-body problem

- Two-body -> Three-body -> N-body





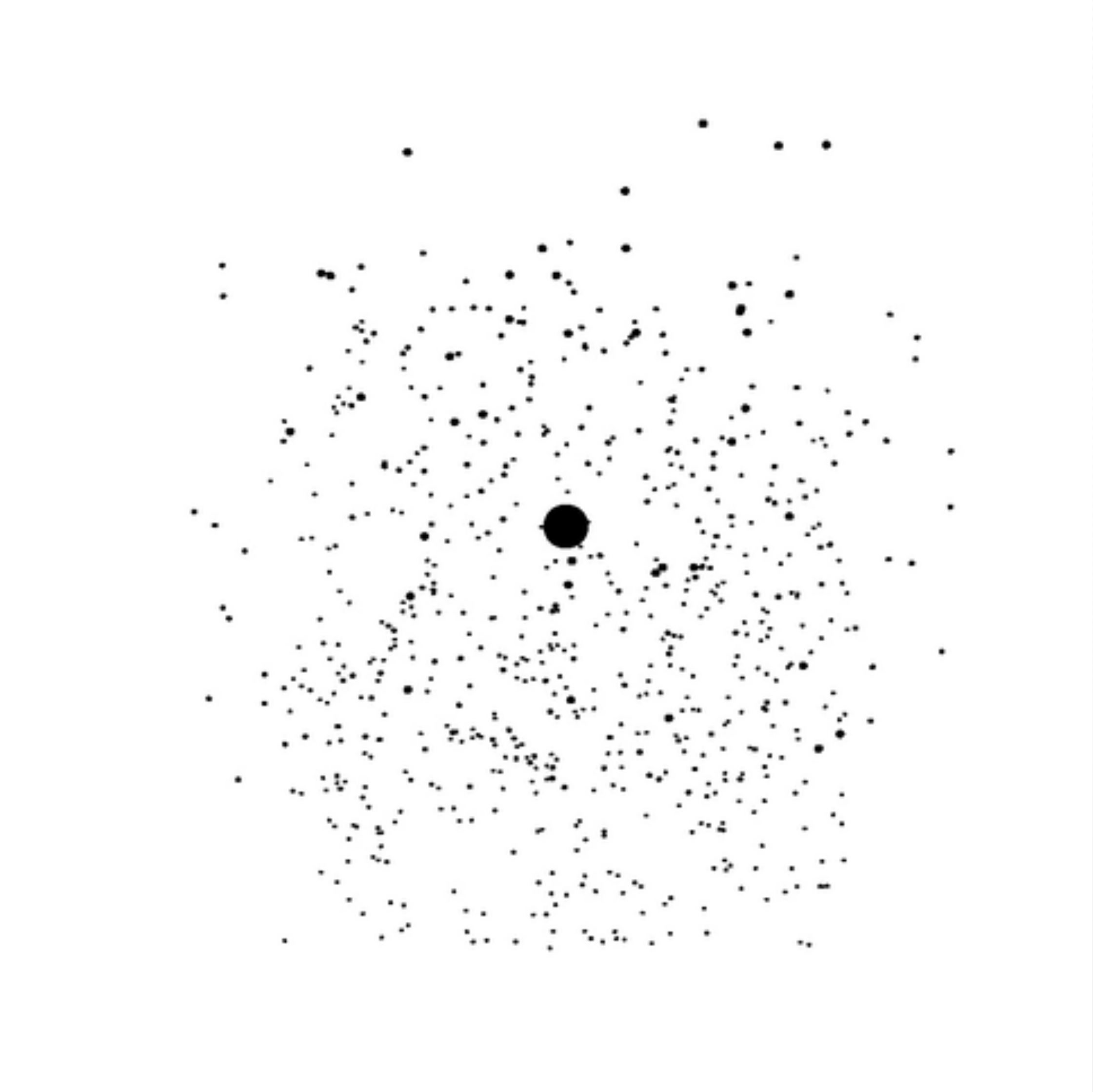
- Origin of the chaotic theory
  - Chaos: When the present determines the future, but the approximate present does not approximately determine the future.

Ref: Three body: http://www.3body.net/wp-content/uploads/2013/06/3body_problem_figure2.gif

Newton Portrait: http://www.newton.cam.ac.uk/art/portrait.html

Grand Cross evolution

- Demo



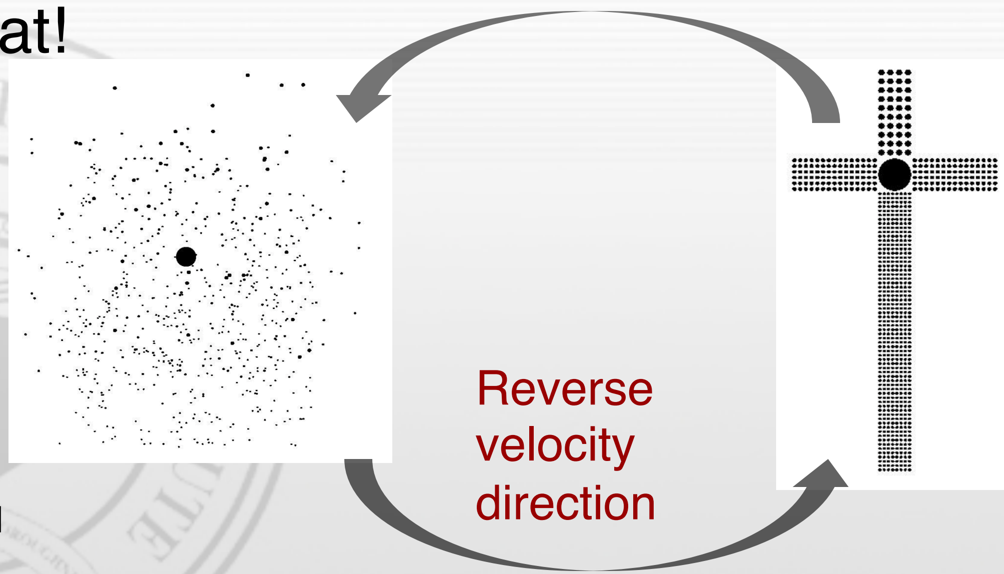
Grand Cross evolution *cont.*

- How we did it?!
  - Solve a constrained minimization problem

$$\pi = |\boldsymbol{u} - \boldsymbol{u}_{cross}|_{min} + R(\alpha)$$

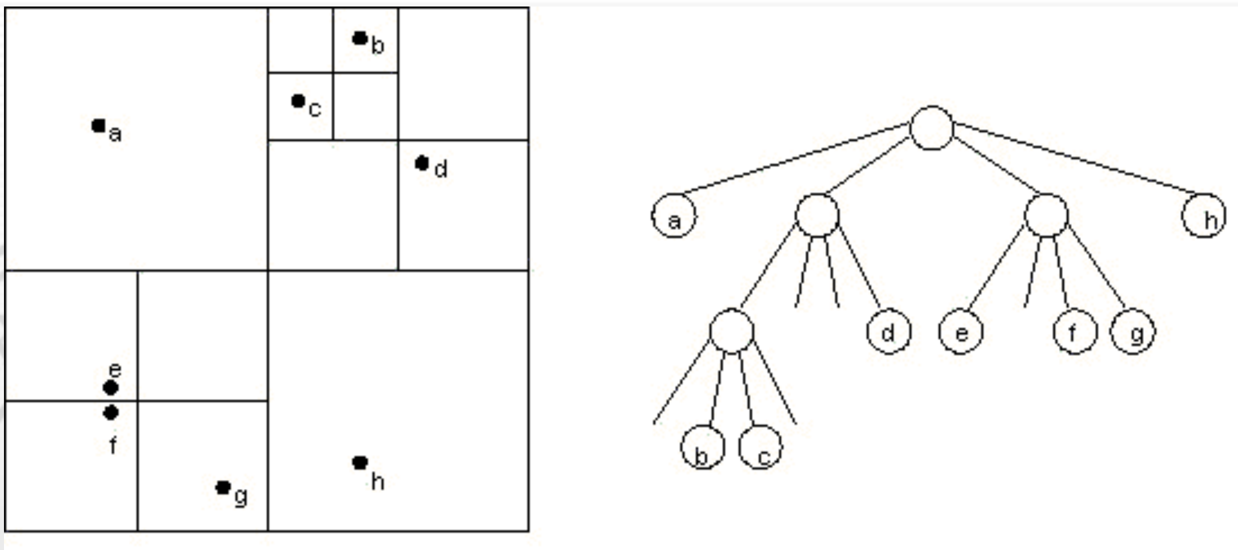
$$\text{subject to} \quad \boldsymbol{u}_j = (\boldsymbol{v}_j^t + \sum_{i=1, i\neq j}^{N} G\frac{m_i}{r_{ij}^2}\Delta t)\Delta t, \quad j = 1...N$$

  - Cheat!



Sequential implementation

- Brutal force scheme
  - Almost trivial, $O(N^2)$
- Barnes Hut algorithm
  - Quad tree (2D) / Oct tree (3D)



  - O(N log N)

Ref: Quad tree: http://www.cs.princeton.edu/courses/archive/fall03/cs126/assignments/barnes-hut.html

Barnes Hut algo: Barnes, Josh, and Piet Hut. "A hierarchical O (N log N) force-calculation algorithm." Nature,324, (1986): 446-449.

Sequential implementation *cont.*

- Implementation of Seq. Barnes Hut algorithm
  - For simplification: set a fixed bounding box, planets fly beyond the boundary are considered escaped and removed from the tree;
  - Accuracy control: *Theta = s / d = 0.025*, where *s* is the width of the region represented by the internal node, and *d* is the distance between the body and the node's center of mass;
  - No sorting nodes, softening factor = 0.01 for close planets, nodeMax = 16, fixed time step dt=0.001.
  - Pseudo code

    Initialize tree;

    for 1 : nStep {calculate center of mass for each node;

    calculate interactive attractions among nodes;

    update positions;

    migrate nodes if necessary; }

Ref: http://charm.cs.uiuc.edu/cs498lvk/projects/bhatele/code/sequential/

Rensselaer

Parallel implementation

- Brutal force scheme
  - moved force calculations to the kernel
  - The kernel code computes forces between a body and itself to eliminate an if statement
- Barnes Hut algorithm
  - Parallelism mainly exists in the following for-each loops[1]:

```
For each body x:
    search for the node set N1 in the quad tree that act on the body
    for each node y in N1:
        calculate the force on x from y
```

  - Grouping the bodies by spatial distance before the force calculation greatly improved the performance

Ref: Martin Burtscher, Keshav Pingali."An Efficient CUDA Implementation of the Tree-Based Barnes Hut n-Body Algorithm." Nature, 324, (1986): 446-449.

Performance Evaluation

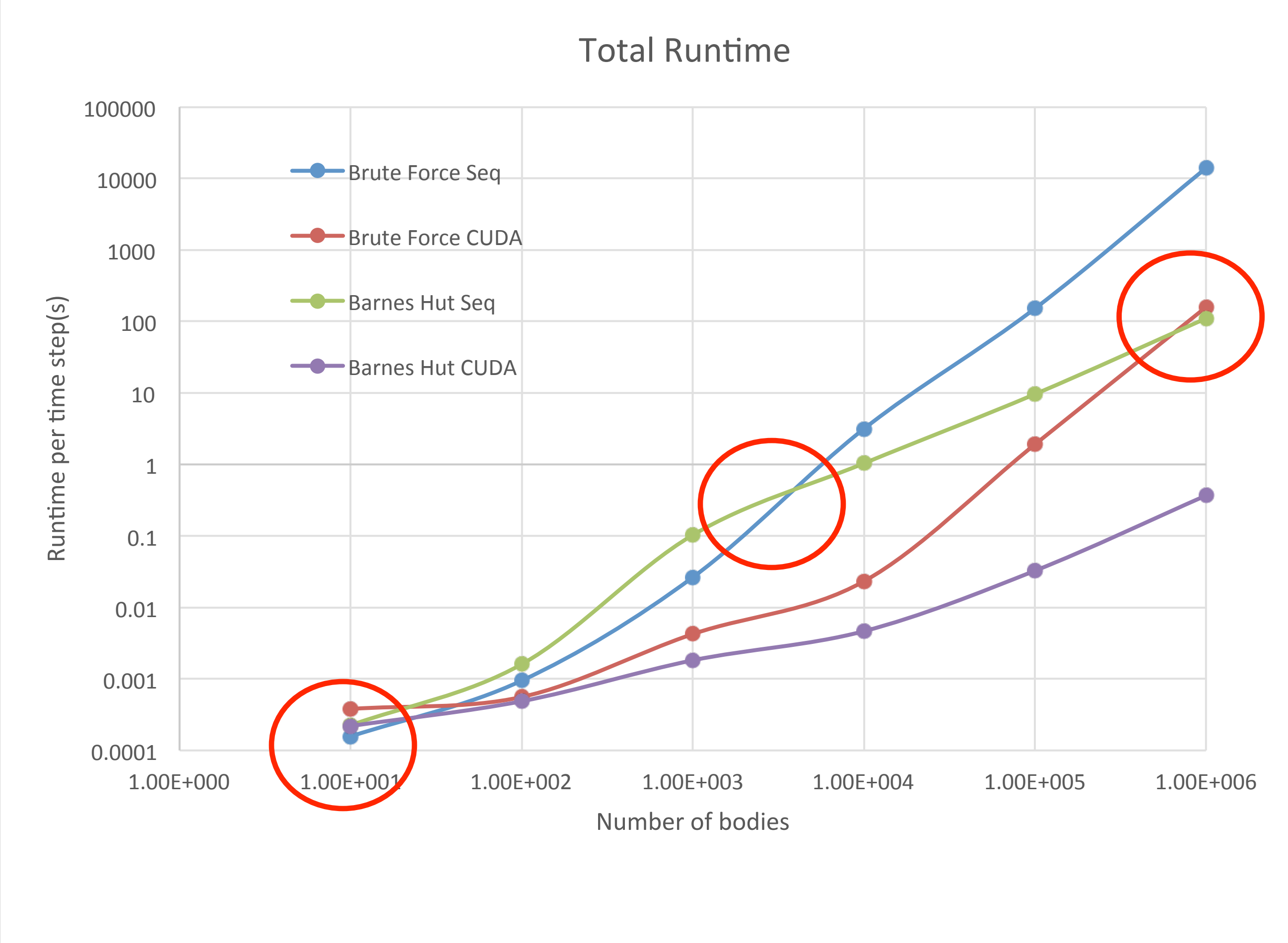
- Systems
  - Intel(R) Xeon(R) CPU E5-2687W 0 @ 3.10GHz
  - Tesla K20Xm GPU
- Compilers
  - Sequential Brute Force (gcc 4.9.2 -O3)
  - Sequential Barnes Hut (gcc 4.9.2 -O2)
  - CUDA Brute Force (nvcc 7.0 -O3 -arch=sm_20)
  - CUDA Barnes Hut (nvcc 7.0 -O3 -arch=sm_20)
- Inputs
  - 10 to 10^6 bodies
  - Best runtime of experiments, excluding I/O

Ref: Barnes Hut algo: Barnes, Josh, and Piet Hut. "A hierarchical O (N log N) force-calculation algorithm." Nature,324, (1986): 446-449.

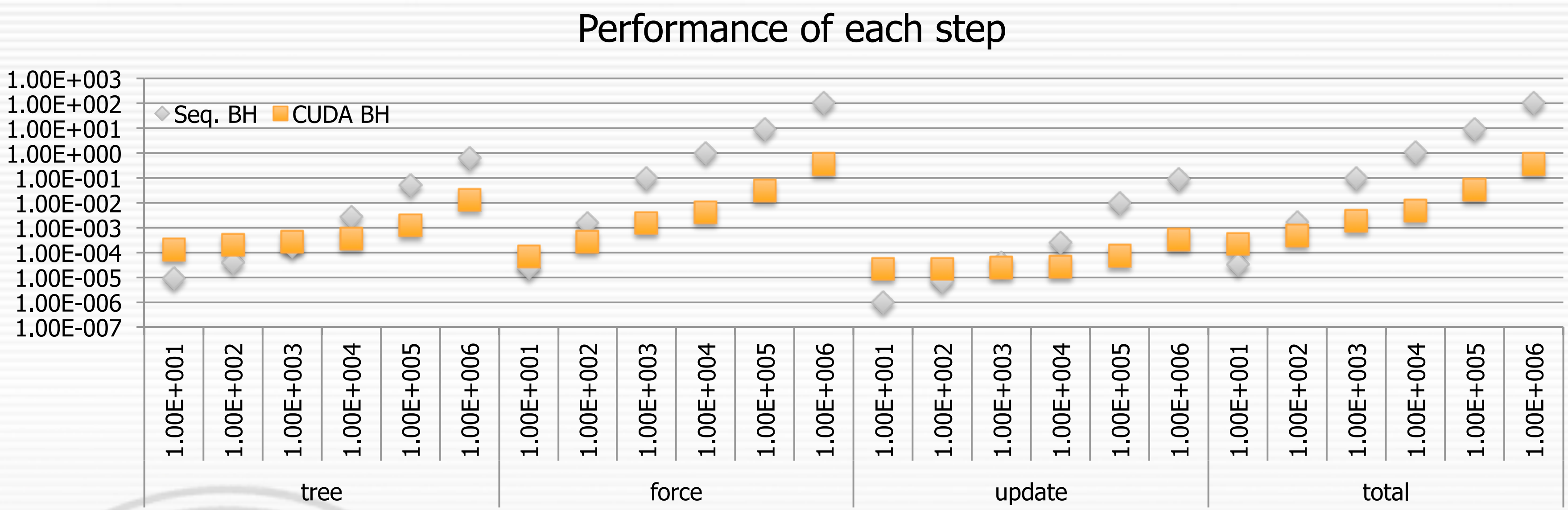
Rensselaer

Performance Evaluation *cont.*

- The benefit is lower with small N since the amount of parallelism is lower
- The cost of building tree is too much while N is small
- The O(N logN) BH makes its benefit over the O(n^2) algorithm increases rapidly with larger N.



Performance Evaluation *cont.*



- Force calculation > Building tree ≫ Updating new positions
- The Spatial Grouping function greatly improved the performance in calculating forces part

www.rpi.edu

Conclusions

- Cross shape exact recovery
- Implement entire Brute Force and Barnes Hut algorithm on CPU and GPU
  - Number of bodies matters
  - Building tree cost
  - Spatial grouping greatly improves the performance

Ref: Barnes Hut algo: Barnes, Josh, and Piet Hut. "A hierarchical O (N log N) force-calculation algorithm." Nature,324, (1986): 446-449.