N-body Interaction Simulations I

Learning Physics from the Sky Down

- Discovering phenomenology from simulations
- Using fundamentals to build models
- Coding algorithms to build simulations from models

Norman Chonacky
Department of Applied Physics
Yale University -- New Haven CT USA
Dave Joiner
Department of Science, Technology, and Math Education
Kean University -- Union NJ USA

Introduction to this Session (I)

Demonstration (GalaxSee):
a simulated N-body gravitating system.

- What can be learned?
  - Extreme example of Newton's law of gravitation
  - Basic physics, interplay between gravitational binding and angular momentum
  - Slightly more advanced, interplay between kinetic energy and gravitational binding (Virial Theorem)
  - Scaling of N-body style (action at a distance) problems
  - Unit systems and normalization

Does gravity need an N-body model?

- Why is an electromagnetic system different?
  - Action at a distance is different when charges cancel
  - Complicated electromagnetic systems (i.e. macromolecules solvated in polar solvents) may still have significant action at a distance for key problems.

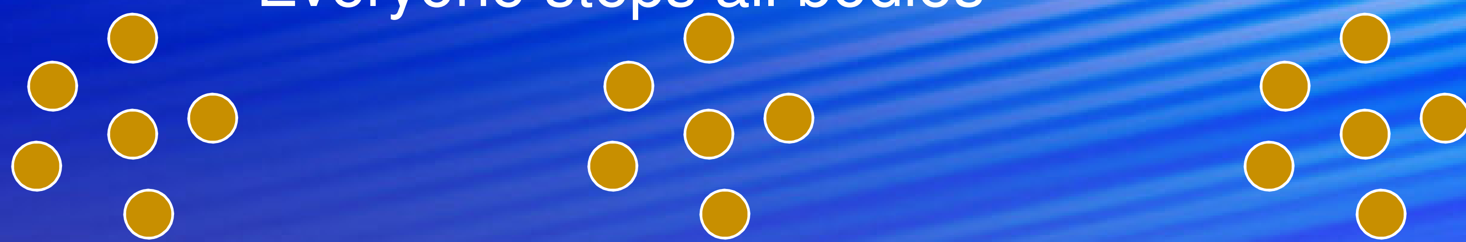
Direct Force Calculation

Algorithm

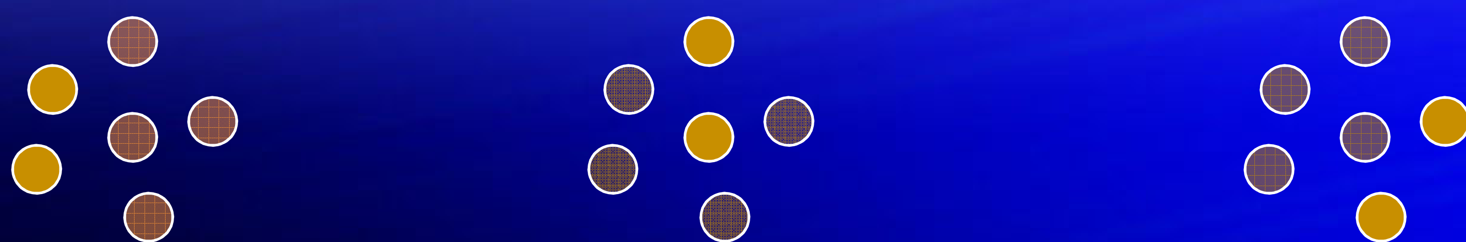
- Given its velocity, what is displacement for each object in this domain?
- Given their new positions, **WHAT IS THE RESULTANT FORCE** on each from all the other objects?
- What is the change in velocity of each object due to the net force on it?

Parallelization

Everyone steps all bodies



But forces are calculated across processes



Parallelization in the Direct Force Algorithm

The steps needed for each calculation:

- Single Instruction – Multiple Data
  - Each process integrates every particle
  - Each process calculates some of the accelerations
  - Each process shares acceleration information

The point of diminishing returns:

- The more you split up the problem, the less work each processor does, thus the ratio of concurrent work to redundant work and communication reduces.

Amdahl's Law and optimal efficiency:

- General Law
  - Best case time =
    A CONCURRENT/P + B REDUNDANT
  - Speedup approaches a limit
- N-body is worse:
  - time = $(a \cdot N^2 / P) + (b \cdot N \cdot P) + (c \cdot N) + (d \cdot P)$
  - Speedup falls off as 1/P for large P (time increases linearly)
  - Large N, less communication can increase the value of P before speedup falls off.

Gathering Performance Data

Logging onto a cluster:

Running GalaxSee on a cluster:

Gathering run time data:

- See handout

Prelude to N-Body II

Where is the most computation-intensive part of the calculations in GalaxSee?

- Force calculation, not the kinematic calculation.

Scalability issues:

- We must reduce the number of steps in this part, as dependent upon N - the number of objects.