Math 298
Fundamental Concepts in Computational and Applied Mathematics
Lecture 7: N-Body Methods

Juan Meza
School of Natural Sciences
University of California, Merced

N-Body methods

- Involve computation of interactions between N-bodies/particles
- Examples arise in molecular dynamics, gravitation, electrostatics, etc.
- Also useful for solution of boundary value problems, biharmonic equations, Poisson equation, etc.

N-Body Problem Description

Want to compute sums of the following form:

$$u(x) = \sum_{i=1}^{N} w_i K(x, y_i), \tag{1}$$

where

- $x_i, i = 1, \ldots, N$ are called the source points
- $y_j, j = 1, \ldots, N$ are called the target points
- w_i are source weights
- $K(x, y_i)$ is called the kernel, e.g. potential function

Remark: : A straightforward algorithm would appear to be $\mathcal{O}(N^2)$

N-Body problem of gravitation

The gravitational potential is given by

$$\Phi(x_j) = \sum_{\substack{i=1 \\ i \neq j}}^{N} \frac{m_i}{r_{ij}}$$

and the gravitational field E by:

$$E(x_j) = \sum_{\substack{i=1 \\ i \neq j}}^{N} m_i \frac{x_j - x_i}{r_{ij}^3}$$

Remark: Same equations applicable to electrostatics

N-Body Problem (Short Detour)

Recall one such problem from our previous class, i.e. the FFT:

$$u_j = \sum_{k=1}^{N} e^{2\pi i j k/N} w_k,$$

for $j = 1, \ldots, N$.

What is the complexity for such an algorithm?

Finite Rank/Degenerate Kernels

First consider a kernel which can be written as:

$$K(x, y) = \sum_{k=1}^{p} \phi_k(x)\psi_k(y).$$

These are called finite rank or degenerate kernels.
We can reduce our original problem Eq (1) to the following 2-step procedure.

1. Compute $A_k = \sum_{i=1}^{N} w_i \psi_k(y)$
2. Evaluate $u(x) = \sum_{k=1}^{p} A_k \phi_k(x)$

What was the complexity for such an algorithm?

Class Participation

Question:

Can you think of another example where you might be able to use this idea?

Answer:

Motivation

Like many other problems first take a look at the structure of the problem

- Forces can usually be broken down into "short-range" and "long-range"
- Can we take advantage of this to develop faster algorithms?

2 Key Ideas

1. Replace group of "distant" particles by one "pseudo-particle"
2. Decompose space into a hierarchy of areas that are suitably "distant"

Replacing group of particles: multipole expansion

Example: Electrostatic Potential due to a set of charges q_i located at x_i

Want

$$K(y-x) = \frac{1}{|y-x|} \approx \sum_{k=0}^{p} \phi_k(x)\psi_k(y)$$

which is given by

$$\frac{1}{|y-x|} = \frac{1}{|y|}\sum_{n=0}^{\infty} P_n(\cos\theta)\left(\frac{|x|}{|y|}\right)^n$$

where $P_n(\cos\theta)$ are the Legendre polynomials.

N.B. Series is convergent for $v = \frac{|x|}{|y|} < 1$

Hierarchy of domains

- Similar to our old friend divide and conquer
- Need to be careful about dividing space
- Too coarse a division and your approximation is not good enough
- Too fine a division leads you back to the original problem

Four key features of an FMM code

- A specified acceptable accuracy
- A hierarchical subdivision of space into panels or clusters of sources
- A far field expansion of the kernel in which the influence of source and evaluation of points separates
- (Optional) Conversion of far field expansions into local expansions

N.B. From Reference 2, Beatson and Greengard

Comparison of FFT and FMM

Property	FFT	FFM
Work	$5N \log N$	$N \log N$
Accuracy	exact	approximate
Domain	uniform spatial grid	any
Based on	Algebra	Analytics

Table: Comparison of FFT with FMM

References

- **A Fast Algorithm for Particle Simulations**, L. Greengard and V. Rokhlin, J. Comp. Phys. 73, 325-348, 1987
- **A Short Course on Fast Multipole Methods**, R. Beatson and L. Greengard
- **A Short Primer on the Fast Multipole Method**, Vikas Chandrakant Raykar, vikas@umiacs.umd.edu