Fundamental Concepts in Computational and Applied Mathematics

Juan Meza
School of Natural Sciences
University of California, Merced

Fall 2014

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

Suppose we are given a set of source points $x_i, i = 1, \ldots, N$ and we want to compute sums of the following form:

$$u(x_i) = \sum_{j=1}^{N} w_j K(x, y_j), \tag{1}$$

where

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

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

Example: 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:

$$u_j = \sum_{k=1}^{N} e^{2\pi ijk/N} \cdot w_k, \qquad j = 1, \ldots, N.$$

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 is the complexity for such an algorithm?

Discussion

Question

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

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 can be written as

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

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 [1]

- 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

[1] Beatson and Greengard[1]

Comparison of FFT and FMM

Table: Comparison of FFT with FMM

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

References I

R. Beatson and L. Greengard
A Short Course on Fast Multipole Methods,
Cambridge University Press, 2009.

L. Greengard and V. Rokhlin.
A Fast Algorithm for Particle Simulations,
J. Comp. Phys. 73, 325-348, 1987.

Vikas Chandrakant Raykar
A short primer on the fast multipole method,
http://www.umiacs.umd.edu/labs/cvl/pirl/vikas/publications/FMM_tutorial.pdf