18.336/6.335 Fast Methods for Partial Differential and Integral Equations Spring 2013

February 5, 2013

Instructors: Laurent Demanet (laurent@math.mit.edu), Jacob White (white@mit.edu)
TA: Richard Zhang (ryz@mit.edu)
Office hours: Wednesday 2-4pm, room 2-392.

1 Syllabus

1.1 Course grade

50% problem sets - lowest score dropped, deadline on Thursdays. 50% final project - maximum 5 pages conference paper style report and presentation.

1.2 Key dates

02/19 (Tue)	No class.
03/06 (Wed)	Project proposal due.
03/08 (Fri)	Add date.
04/03 (Wed)	Mid-term meeting, to discuss progress of final project.
05/14 (Tue), 05/16 (Thu)	Final project presentation.
05/16 (Thu)	Project report due.

1.3 Prerequisites:

- 18.085: knowledge of ODEs, basic PDEs, exposure to Laplace's equation, heat equation, Poisson's equation.
- Fourier transform.
- Matrices and linear algebra. Decompositions, subspaces.
- Basic numerical methods, ODE or PDE.
- Programming, done in language of choice. E.g. Matlab / Julia, Python or C/C++.

1.4 Class Topics

Transversal views of:

- Formulating problems. PDE, linear integral equations.
- Discretization and numerical methods. Finite differences, Galerkin methods, Nystrom methods etc.
- Fast algorithms.
- Apply $Ax = b$ fast, or invert A fast.

2 Motivation: A simple example PDE

Consider $u \triangleq$ potential, $f \triangleq$ charge density within the Poisson equation:

$$\Delta u(x) = f(x), \tag{1}$$

where Δ is the Laplacian operator:

$$\Delta \equiv \nabla^2 \equiv \nabla \cdot \nabla \equiv \sum_i \frac{\partial^2}{\partial x_i^2}.$$

Equation (1) has the integral form:

$$u(x) = \int G(x,y) f(y)\, \mathrm{d}y. \tag{2}$$

$G(x,y)$ is known as the Green's function. For example, it is known in the three-dimensional Poisson's equation to be:

$$G(x,y) = \frac{1}{4\pi \|x-y\|}$$

The Poisson equation is common to many interrelated fields. The $u \iff f$ pair could also be defined as temperature $\iff$ power density in the heat equation, or gravitational potential $\iff$ mass density in astrophysics. Regardless of the definitions, the objective for the standard forward problem is to obtain $u(x)$ for some known $f(x)$, either by solving (1) or by applying (2).

A simple approach to the above problem in one dimension is to discretize the solution domain (i.e. an interval) into N sub-intervals with uniform grid spacing h:

The PDE of (1) can be discretized using either finite differences or finite elements, both giving the same matrix equation for this very simple problem:

$$\Delta_h = \frac{-1}{h^2} \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & 2 & -1 \\ & & & -1 & 2 \end{bmatrix}$$

$$\Delta_h u_h = f_h$$

$$u_h = \Delta_h^{-1} f_h$$

The overarching goal in the design of numerical methods is to solve equations in $O(N)$ operations. Suppose Δ_h were dense; it would take $O(N^3)$ to find the inverse and thus solve the problem. However in this very simple problem, Δ_h is sparse and can be quickly solved using Gaussian elimination in $O(N)$. To see how this might be the case, consider the LU decomposition of the one-dimensional Δ_h:

$$\Delta_h^{(1D)} = \underbrace{\begin{bmatrix} \ddots & & & \\ \ddots & \ddots & & \\ & \ddots & \ddots & \\ & & \ddots & \ddots \end{bmatrix}}_{L} \underbrace{\begin{bmatrix} \ddots & \ddots & & \\ & \ddots & \ddots & \\ & & \ddots & \ddots \\ & & & \ddots \end{bmatrix}}_{U}$$

	Operations	Memory
1-D	$O(N)$	$O(N)$
2-D	$O(N^3)$	$O(N^2)$
3-D	$O(N^6)$	$O(N^3)$

Table 1: Complexities of Gaussian elimination for the discretized Poisson problem of various dimensionalities.

Being bi-diagonal matrices, L and U can be inverted in $O(N)$ operations by back substitution. The full Δ_h matrix can therefore be inverted by inverting each of its components:

$$\begin{aligned} [LU]\, u_h &= f_h \\ U u_h &= L^{-1} f_h \\ u_h &= U^{-1} \left[L^{-1} f_h \right] \end{aligned}$$

Note that this is a special, and very fragile example where Gaussian elimination alone achieves $O(N)$ complexity. Performing the same routine on the two- and three-dimensional Δ_h would fare significantly worse (See Table 1). $x = A \backslash b$ is not enough!

2.1 The four major themes of fast methods

As described above, it is very uncommon for a problem to be solved in $O(N)$ complexity using Gaussian elimination. Fast algorithms are usually necessary for very large, complex, real world problems. A large collection of fast algorithms have been developed over the years; however, they generally share the following four major themes:

1. Translational invariance, or the statement "the physics are the same irrespective of where the interaction takes place." In our 1-D Poisson example, interactions between intervals 1 & 2 would be the same if they were relocated to intervals 5 & 6.

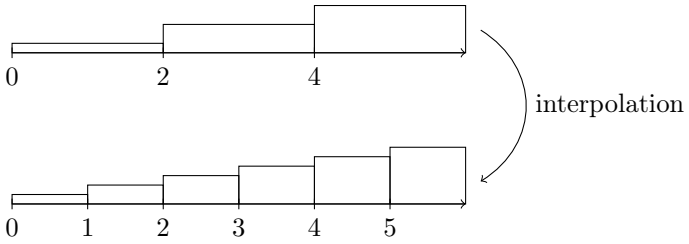
For the Poisson problem discussed above, this property is highlighted in the fact that the Δ_h matrix contains identical diagonal and off-diagonal lines. Equivalently, translation invariance is seen in the fact that the Green's function $G(x, y)$ is only dependent on the difference vector $(x - y)$.

Fast Fourier Transform (FFT) -based methods are the preferred fast methods for translationally invariant problems, giving $O(N \log N)$ complexity [1]. Our first few lectures is to discuss how to treat these simple systems using FFT methods. Some example applications that can be treated with FFT methods:

- Ewald summation, molecular dynamics.
- Volume integral equation, MRI.
- Krylov-subspace methods.
- Precorrected FFT methods.

2. Multiscale. A problem is considered to be multiscale when we can solve the problem first at a coarse scale, then leverage the information gathered to solve at a finer mesh scale. This intermediate step between the coarse and fine mesh scales is known as "interpolation".

[1] Despite being multiplied by a $logN$ factor, FFT-based methods are usually competitive even against $O(N)$ algorithms like the Fast Multipole Method.



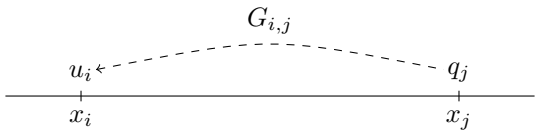
The multigrid algorithm is the dominant algorithm for multiscale problems. It is great for PDEs with lack of uniformity, for example spatially dependent weights:

$$\nabla \cdot \alpha(x) \nabla u(x) = f.$$

It is also very useful for complicated mesh grids or the solution domains. Multigrid usually gives an $O(N)$ algorithm, although the constant factor can be quite large. Typical applications for the multigrid algorithm are interior problems, such as:

- Fluid / heat flows
- Reservoir simulations
- Dielectric
- Immersed interfaces

3. Low-rank interactions. Consider the Poisson problem applied to $x \in [0, 1]$, discretized into N sub-intervals. Using an integral formulation, we can denote $G_{i,j}$ as the potential contribution at $x = x_i$, caused by a charge f_j placed at $x = x_j$.



If a charge is placed at each of the $n+1$ nodes in the interval, then u_i is the total sum of the contribution by each charge:

$$u_i = \sum G_{i,j} f_j.$$

When this equation is written for all $n + 1$ charges, the following matrix equation is created:

$$\begin{bmatrix} u_0 \\ u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} = \begin{bmatrix} G_{0,0} & G_{0,1} & G_{0,2} & \cdots & G_{0,n} \\ G_{1,0} & G_{1,1} & & & \vdots \\ G_{2,0} & & \ddots & & \vdots \\ \vdots & & & \ddots & \vdots \\ G_{n,0} & \cdots & \cdots & \cdots & G_{n,n} \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ \vdots \\ q_n \end{bmatrix}$$

$$u_h = G_h f_h$$

The matrix G_h characterizes the potential interaction, and is known as the *Green's function matrix*. For the Poisson problem, it can be readily shown that $G_h = \Delta_h^{-1}$. As every charge affects every node within the system, G_h is a dense matrix, and the formation of G_h and evaluation of u_h given f_h are both of $\mathcal{O}(N^2)$ complexity.

The interaction that governs G is said to be *low-rank* if far off-diagonal sub-matrices of G_h, for example:

$$G_{far} = \begin{bmatrix} G_{0,n-2} & G_{0,n-1} & G_{0,n} \\ G_{1,n-2} & G_{1,n-1} & G_{1,n} \\ G_{2,n-2} & G_{2,n-1} & G_{2,n} \end{bmatrix}$$

are of low-rank. Equivalently, this means that off-diagonal matrices like G_{far} can be decomposed to very few singular values using SVD. Physically a low-rank interaction is one where the dimensionality of interaction is low. In the electrostatic, potential flow or gravitational potential problem, the interactions are low-rank because many far-away interactions can be lumped together and approximated as a single interaction without any significant loss of accuracy.

The Fast Multipole Method (FMM), also known as H-matrices and partitioned low-rank methods, is the dominant method in problems where low-rank interactions are considered. To use the FMM, the problem should be formulated as integral equations, where the Green's function is known. They are most efficient when there is a lack of uniformity in the excitation f, e.g. where most of the problem domain is empty free-space. Boundary integral equations are often solved using low-rank methods. A single-layer potential (SLP) integral is a surface integral of the following form:

$$u(x) = \int_S f(x)G(x,y)\,\mathrm{d}S_y.$$

Similarly, a double-layer potential (DLP) integral is written:

$$u(x) = \int_S f(x)\frac{\partial}{\partial n_y}G(x,y)\,\mathrm{d}S_y$$

where $\partial/\partial n_y$ is a partial derivative in the normal direction relative to $\mathrm{d}S_y$. In most cases, the nearby- and self-interactions involve integrals of singularities, and require special treatment. Applications for low-rank methods include:

- Potential flows
- Electrostatics
- Biomolecules
- MEMS

4. High-frequency problems. For example, consider the Helmholtz equation:

$$\Delta u + k^2 u = f$$

where $k = \omega/c$ is the wave number. Equivalently, the Green's function is:

$$G(x,y) = \frac{e^{jk\|x-y\|}}{\|x-y\|}$$

When $G(x,y)$ is plotted, one observes a decaying oscillation with a period of $\lambda = 2\pi/f$. This is a defining characteristic of high-frequency problems, and the interaction is no longer low-rank anywhere.

Butterfly algorithms have been developed to solve high-frequency problems. First, using butterfly steps, the high-rank high-frequency interaction is transformed into two unrecognizable low-rank interactions. Then, the low-rank interactions can be freely treated at $O(N)$ using the same philosophy as the Fast Multipole Method. Typical problems are:

- Acoustic, elastic and EM scattering.
- Radar imaging.

2.2 Summary - a comparison of themes

	Apply Int. Eqn.	Solve PDE	Solve Int. Eqn.
Uniform	FFT	FFT	FFT
Non-Uniform	FFT++	Multigrid	h-matrices
Very non-uniform	FMM	h-matrices	?
High-frequency non-uniform	Butterfly	?	?