

# Fast Multipole Algorithm (FMA)

## Seminar „Top Ten Algorithms“

Daniel Vetter

22. May 2008

# Overview

- 1 Introduction
- 2 Potential Fields in 2D
- 3 The Fast Multipole Algorithm in 2D
- 4 Advanced Topics
  - Adaptive Algorithm
  - Boundary Conditions
  - FMA in 3D
- 5 Does it Work?

Most of the material is drawn from [Greengard, 1988].

# Introduction

## Problem

Evaluate the force upon  $N$  particles, where the force  $F$  has the form

$$F = F_{\text{near}} + F_{\text{far}} + F_{\text{external}}.$$

- $F_{\text{near}}$  e.g. chemical bonding forces.
- $F_{\text{external}}$  e.g. external known field (earth's magnetic field).

Both can be evaluated in  $\mathcal{O}(N)$ .

## The Real Problem

$$F_{\text{far}} = \nabla\Phi \text{ where } \nabla^2\Phi = 0,$$

the classical electrostatic and gravitational fields of the particles themselves.

- Naive approach:  $\mathcal{O}(N^2)$
- Clever divide-and-conquer approach:  $\mathcal{O}(N \log N)$ .
- The FMA manages  $\mathcal{O}(N)$ !

# Potential Fields in 2D

## Fundamental Field

$$\Phi_{x_0} = \log(\|x - x_0\|)$$

for a particale at  $x_0$  and  $\nabla^2\Phi = 0$ .

## Lemma

Let a point charge of intensity  $q$  be at  $z_0$ . Then for  $z$  with  $|z| > |z_0|$ :

$$\Phi_{z_0}(z) = q \log(z - z_0) = q \left( \log(z) - \sum_{k=1}^{\infty} \frac{1}{k} \left( \frac{z_0}{z} \right)^k \right).$$

## Theorem (Multipole Expansion)

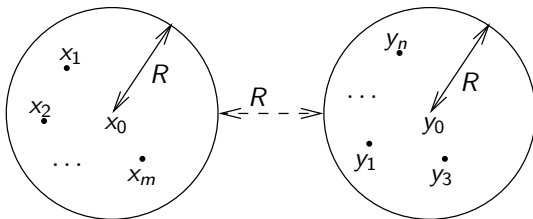
*$p$ -term Multipole Expansion (ME) of  $m$  particles with charge  $q_i$  at locations  $z_i$  around 0 is the Laurent series truncated after the  $p$ -th term:*

$$\Phi_{ME}(z) = a_0 \log(z) + \sum_{k=1}^p \frac{a_k}{z^k}$$

*with the error estimate*

$$|\Phi(z) - \Phi_{ME}(z)| \leq A \cdot c_1(c) \cdot \left(\frac{1}{c}\right)^p$$

*where  $c = \left|\frac{z}{R}\right|$ ,  $R = \max |z_i|$  and  $A = \sum_{i=1}^m |q_i|$ .*



Evaluating the field due to the particles  $x_1, \dots, x_m$  at the points  $y_1, \dots, y_n$  using multipole expansion takes only  $\mathcal{O}(mp) + \mathcal{O}(np)$  operations.

## Lemma (Conversion of Multipole Expansion into Local Expansion)

$(a_0, \dots, a_p)$  coefficients of a ME with center  $z_0$  and radius  $R$ .  
 If  $|z_0| \geq (c + 1)R$  with  $c > 1$ , then  $\Phi_{ME}$  converges in a circle of radius  $r$  around the origin. The local expansion (LE) is then the Taylor expansion truncated after the  $p$ -th term:

$$\Phi_{LE}(z) = \sum_{l=0}^p b_l \cdot z^l \text{ where } b_0 = a_0 \log(-z_0) + \sum_{k=1}^p \frac{a_k}{z_0^k} (-1)^k$$

$$\text{and } b_l = -\frac{a_0}{l \cdot z_0^l} + \frac{1}{z_0^l} \sum_{k=1}^p \frac{a_k}{z_0^k} \binom{l+k-1}{k-1} (-1)^k, \text{ for } l \geq 1.$$

$$\text{Error estimate: } |\Phi_{ME}(z) - \Phi_{LE}(z)| \leq A(c_1(c)p + c_2(c)) \left(\frac{1}{c}\right)^p.$$

## Lemma (Translation of Multipole Expansion)

$(a_0, \dots, a_p)$  coefficients of a ME around  $z_0$  and radius  $R$ .  
Then the ME around 0 (and new radius  $R + |z_0|$ ) with coefficients  
 $(b_0, \dots, b_p)$  computes as follows:

$$b_0 = a_0 \text{ and } b_l = -\frac{a_0 z_0^l}{l} + \sum_{k=1}^l a_k z_0^{l-k} \binom{l-1}{k-1}.$$

## Lemma (Translation of Local Expansion)

The same translation for LE (but this time the radius stays the same):

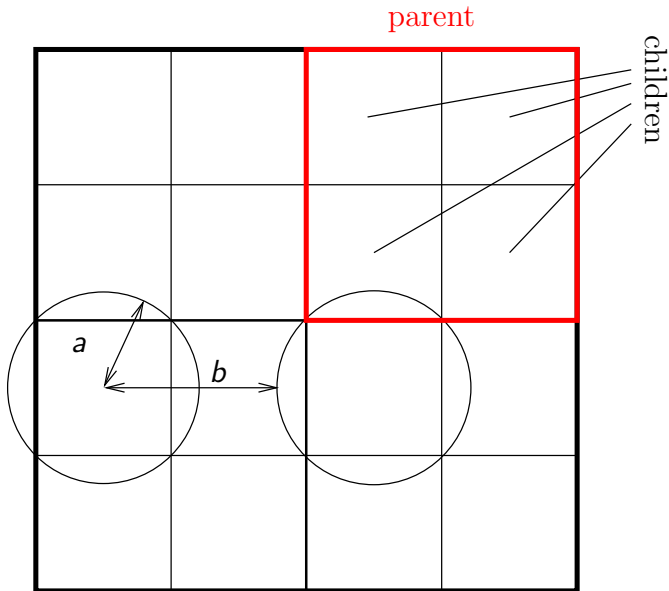
$$b_l = \sum_{k=l}^n a_k \binom{k}{l} (-z_0)^{k-l}.$$

# The Fast Multipole Algorithm in 2D

## Idea

- Recursively divide the computational domain  $C$ .
- Calculate ME's of ever larger areas from the bottom up.
- Evaluate  $\Phi$  using LE's from the top downwards.
- Directly evaluate local effects of neighbouring particles.

This section is based upon [Greengard and Rokhlin, 1987].



$$c = b/a = (4 - \sqrt{2})/\sqrt{2} \approx 1.828$$

# Naming Conventions

- $p = \lceil \log_c(\epsilon) \rceil$  number of terms of the ME and the LE where  $\epsilon$  is the desired precision.
- $n = \lceil \log_4(N) \rceil$  level of refinement of the finest mesh.
- $\Phi_{l,i}$   $p$ -term ME around the center of box  $i$  at mesh level  $l$ .
- $\Psi_{l,i}$   $p$ -term LE around the center of box  $i$  at level  $l$ .
- $\tilde{\Psi}_{l,i}$   $p$ -term LE around the center of box  $i$  at level  $l$ , describing the potential field outside of box  $i$ 's *parent's* box.

# Upward Phase

## Step 1

```
for  $ibox = 1, \dots, 4^n$  do  
    Calculate  $p$ -term ME  $\Phi_{n,ibox}$ .  
done
```

Complexity:  $\mathcal{O}(4^n \cdot p) = \mathcal{O}(Np)$ .

## Step 2

**for**  $l = n - 1, \dots, 0$  **do**

**for**  $ibox = 1, \dots, 4^l$  **do**

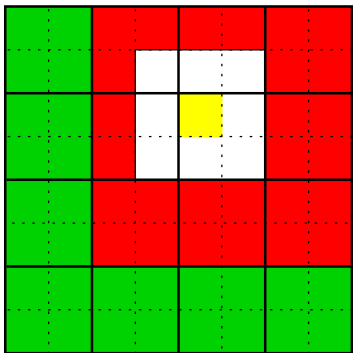
        Calculate  $p$ -term ME  $\Phi_{l,ibox}$  by shifting the center of each child's ME to the center of the current box and adding them up.

**done**

**done**

$$\text{Complexity: } \mathcal{O}\left(\sum_{l=0}^{n-1} 4^l \cdot p^2\right) = \mathcal{O}\left(\frac{1 - 4^{n-1}}{1 - 4} p^2\right) = \mathcal{O}(Np^2).$$

# Downward Phase



For the yellow box, the white boxes are its nearest neighbours. The green ones have already been taken into account for LE at one mesh level above. The red ones are on its *interaction list* and are the ones to be consider at this level for the LE.

## Step 3

**set**  $\tilde{\Psi}_{1,1} = \tilde{\Psi}_{1,2} = \tilde{\Psi}_{1,3} = \tilde{\Psi}_{1,4} = (0, 0, \dots, 0)$

**for**  $l = 1, \dots, n$  **do**

**for**  $ibox = 1, \dots, 4^l$  **do**

    Calculate  $\Psi_{l,ibox}$  by converting the ME  $\Phi_{l,j}$  for each box  $j$  in the *interaction list* of  $ibox$  to an LE around the center of  $ibox$ , adding these LE's and  $\tilde{\Psi}_{l,ibox}$  together.

**done**

**for**  $ibox = 1, \dots, 4^l$  **do**

    Calculate the LE  $\tilde{\Psi}_{l+1,j}$  for each children  $j$  of  $ibox$  by translating the LE to the children's box center.

**done**

**done**

$$\text{Complexity: } \mathcal{O} \left( \sum_{l=0}^n 4^l \cdot (27 \cdot p^2 + 4 \cdot p^2) \right) = \mathcal{O}(Np^2).$$

## Step 4

**for**  $ibox = 1, \dots, 4^n$  **do**

For every particle  $p_j$  located at  $z_j$  in the box  $ibox$

- evaluate  $\Psi_{n,ibox}(z_j)$ ,
- directly compute the interactions with all other particles within the box and its nearest neighbours and
- add far field and direct terms together.

**done**

Assume that there is an upper bound  $k_n$  for the number of particles per box in the finest mesh level.

$$\text{Complexity: } \mathcal{O}\left(N \cdot p + 4^n \cdot \frac{9}{2} k_n + N\right) = \mathcal{O}(Np).$$

# Complexity

Indeed:  $\mathcal{O}(Np^2)$ !

Note that storage complexity is only  $\mathcal{O}(Np)$  (because we need to store  $\mathcal{O}(N)$   $p$ -term ME's and LE's).

# Adaptive Algorithm

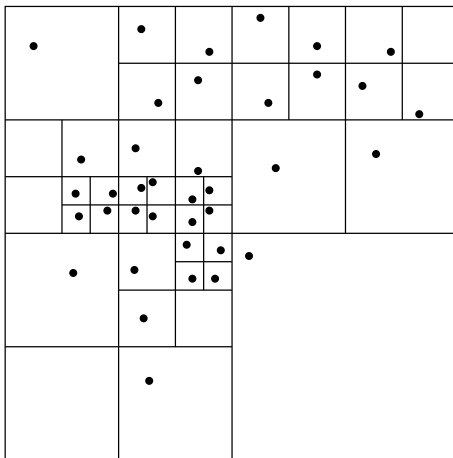
## Problem

What if the particles are not uniformly distributed, i.e. there is no upper bound  $k_n$ ?

## Solution

Fix  $k_n$ .

Recursively divide the boxes until there are at most  $k_n$  particles in a box.



For further details see [Carrier et al., 1988].

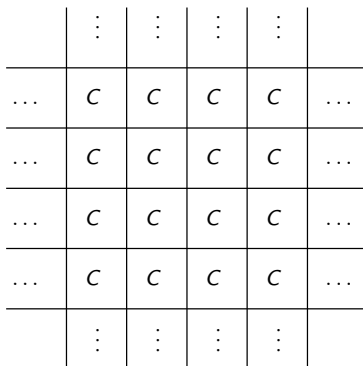
# Boundary Conditions

Two special examples exploiting the FMA:

- Periodic boundary conditions.
- Dirichlet boundary conditions.

# Periodic Boundary Conditions

Copy the computational domain in a mosaic structure over the whole of  $\mathbb{R}^2$ .



Application: Simulation of biochemical reactions in cell fluid.

# Dirichlet Boundary Condition

## Condition on $\phi$

$$\phi = 0 \text{ on } \partial C.$$

- Mirror  $C$  at the top border and invert charges, yielding  $\bar{C}$ .
- Proceed by successively mirroring at the borders of the original domain  $C$  until the whole of  $\mathbb{R}^2$  is filled.

# FMA in 3D

How to carry over the idea of the first lemma into 3D?

$$\Phi_{z_0}(z) = q \log(z - z_0) = q \left( \log(z) - \sum_{k=1}^{\infty} \frac{1}{k} \left( \frac{z_0}{z} \right)^k \right).$$

Use polar coordinates:  $Q, P \in \mathbb{R}^3$ ,  $r = \|Q\|$  and  $\rho = \|P\|$ .

$$\Phi_Q(P) = \frac{1}{r'} \text{ where } r'^2 = r^2 + \rho^2 - 2r\rho \cos \gamma,$$

with  $\gamma = \angle P O Q$ .

Want to expand  $\Phi_Q$  into a series expansion centered at 0.

### Expansion of a Simple Field




$$\frac{1}{r'} = \sum_{n=0}^{\infty} \frac{\rho^n}{r^{n+1}} P_n(u) \text{ where } u = \cos\gamma.$$

$P_n$  denotes the so called Legendre Polynom of degree  $n$ .

# Does it Work?

YES!

# Referenzen

-  Carrier, J., Greengard, L., and Rokhlin, V. (1988).  
A Fast Adaptive Multipole Algorithm for Particle Simulations.  
*SIAM Journal on Scientific and Statistical Computing*, 9:669.
-  Greengard, L. (1988).  
*The Rapid Evaluation of Potential Fields in Particle Systems*.  
Mit Press.
-  Greengard, L. and Rokhlin, V. (1987).  
A fast algorithm for particle simulations.  
*Journal of Computational Physics*, 73(2):325–348.