# The Fast Mulitpole Algorithm in the Differential Algebra Framework to Calculate 3D Self-field between Charged Particles

He Zhang, Martin Berz

Beam Theory and Dynamic Group, Physics and Astronomy Department,
Michigan State University

May 15th 2012, FNAL

# Outline

- Introduction of the fast multipole method (FMM)
- Details of the differential algebra (DA) based fast multipole method
  - Single level fast multipole algorithm for uniform charge distribution
  - Multiple level fast multipole algorithm for any charge distribution
  - parallel version
- Simulation examples

# I. INTRODUCTION

# Introduction

Most algorithm in beam community follows into two categories:

- Particle Particle Interaction (PPI): MAPRO2, SC3DELP, TOPKARK, SCHERM, Improved SCHERM,
- Particle in Cell (PIC): SCHEFF, PICNIC, GPT, IMPACT Z, WARP

We want to bring a new algorithm into the beam community:

- Fast Multipole Method (FMM), L.Greengard and V.Rokhlin, 1987

# Strategy of FMM

- Include the charged region with a box, then cut the box into small boxes.
- For each box (and the charges inside), the whole region can be divided into the near region and the far region to the box.
- For each box (charges inside), the contribution from the boxes (charges) in its near region is calculated directly.
- For each box, its far region is where we can play tricks and gain efficiency!

# Strategy of FMM

- For any box, its field in its far region can be expressed by a multipole expansion. (box-particle relation, $O(N \log N)$.)
  Idea of Tree Code or Barnes-Hut Algorithm
- Multipole expansion in source box can be converted into local expansion in observer box.(box-box relation, $O(N)$.)
  Idea of FMM
- For each box, the field contributed from the far region boxes (charges) can be calculated from the local expansion.
- For each box, the total field is the summation of the near region part and the far region part.

## II.1 SINGLE LEVEL FMM IN DA FRAMEWORK

# Single Level FMM

Two operations in COSY:

- Automatic Taylor expansion of a function

$$f(x+\boxed{\delta x}) = f(x) + f^{'}(x)\delta x + \frac{1}{2!}f^{''}(x)\delta x^2 + \frac{1}{3!}f^{'''}(x)\delta x^3 + ...$$

In COSY,

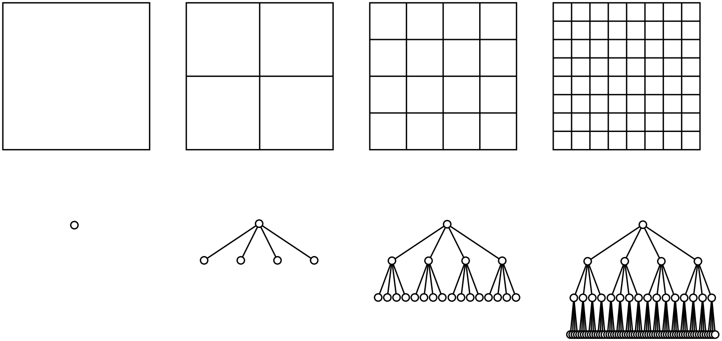
$$f(x+\boxed{da(1)}) = f(x) + f^{'}(x)da(1) + \frac{1}{2!}f^{''}(x)da(1)^2 + \frac{1}{3!}f^{'''}(x)da(1)^3 + ...$$

- Composition of two maps

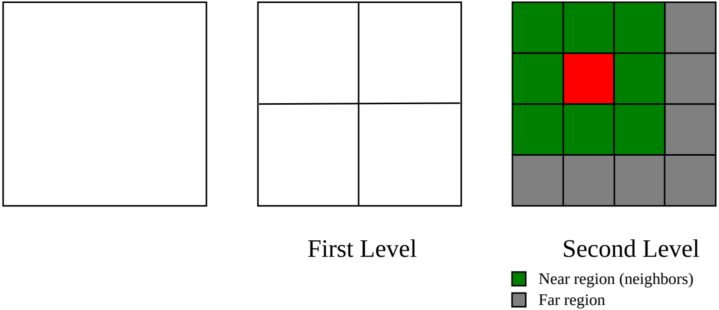
$$G(x) = G(F) \circ F(x), \text{ or } G(x) = G(F(x))$$

In COSY, it can can be done by the command POLVAL.

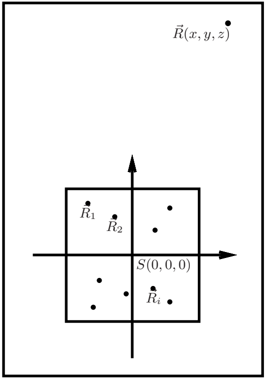
Hierarchical tree structure



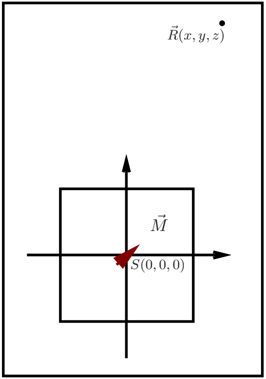
Near region and Far region



Multipole expansion from charges (for the childless boxes)




$\phi_{\vec{R}}(\vec{R}_i) = \phi_{\vec{R}}(\vec{M})$



# Single Level FMM

$$
\begin{aligned}
\phi &= \sum_{i=1}^{n} \frac{q_i}{\sqrt{(x-x_i)^2+(y-y_i)^2+(z-z_i)^2}} \\
&= \sum_{i=1}^{n} \frac{d_r \cdot q_i}{\sqrt{1+(x_i^2+y_i^2+z_i^2)d_r^2-2x_id_x-2y_id_y-2z_id_z}} \\
&= d_r \cdot \bar{\phi}_{\mathrm{c2m}}
\end{aligned}
$$

with

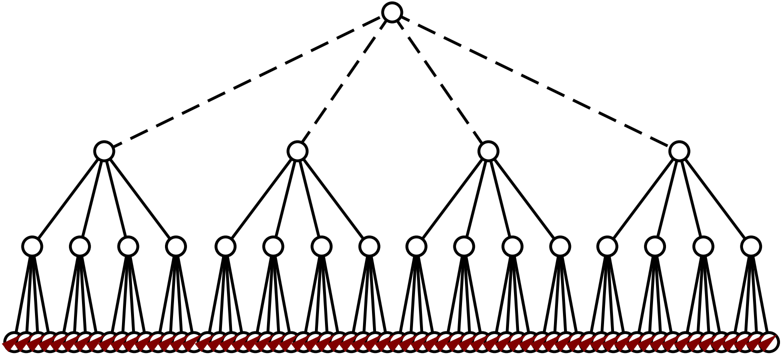
$$
d_x = \frac{x}{x^2+y^2+z^2}, \quad d_y = \frac{y}{x^2+y^2+z^2},
$$

$$
d_z = \frac{z}{x^2+y^2+z^2}, \quad d_r = \sqrt{d_x^2+d_y^2+d_z^2},
$$

$$
\bar{\phi}_{\mathrm{c2m}} = \sum_{i=1}^{n} \left\{ q_i / \sqrt{1+(x_i^2+y_i^2+z_i^2)d_r^2-2x_id_x-2y_id_y-2z_id_z} \right\}.
$$

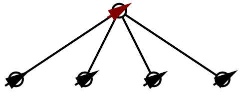
Error $|\epsilon| \leqq C \cdot \left(\frac{a}{r}\right)^{p+1} \cdot \frac{1}{r-a}$, where $C = \sum_{i=1}^{n} |q_i|$ and $r_i \leq a$ for any $i$.

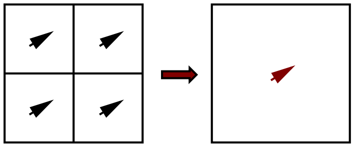
# Single Level FMM



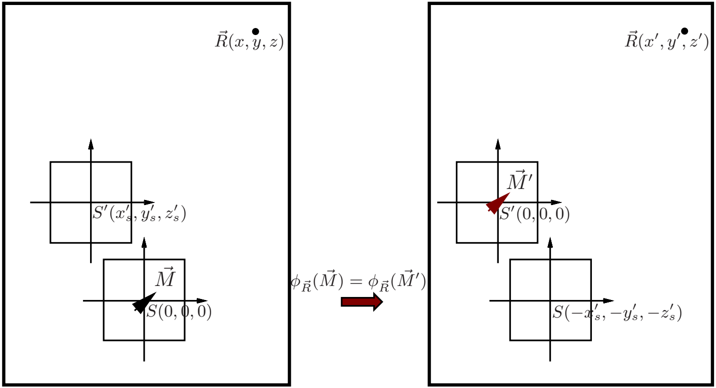
# Single Level FMM

Multipole expansions for the parent boxes





Translate the position of a multipole expansion



# Single Level FMM

In parent box frame, new DA variables are chosen as

$$
d_x^{'} = \frac{x - x_o'}{r'^2} = \frac{x'}{r'^2}, \qquad d_y^{'} = \frac{y - y_o'}{r'^2} = \frac{y'}{r'^2}
$$
$$
d_z^{'} = \frac{z - z_o'}{r'^2} = \frac{z'}{r'^2},
$$

Relation between the old and new DA variables.

$$
\begin{aligned}
d_x &= (d_x' + x_o' \cdot (d_x'^2 + d_y'^2 + d_z'^2)) \cdot R, \\
d_y &= (d_y' + y_o' \cdot (d_x'^2 + d_y'^2 + d_z'^2)) \cdot R, \\
d_z &= (d_z' + z_o' \cdot (d_x'^2 + d_y'^2 + d_z'^2)) \cdot R,
\end{aligned}
$$

$M_{m2m}$

with

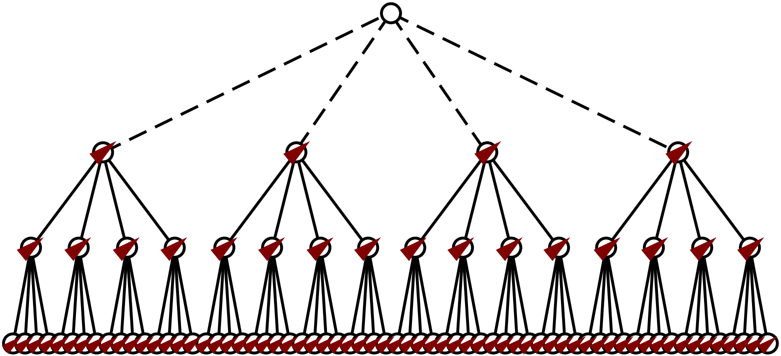
$$
R = \frac{1}{1 + (x_o'^2 + y_o'^2 + z_o'^2)(d_x'^2 + d_y'^2 + d_z'^2) + 2x_o' d_x' + 2y_o' d_y' + 2z_o' d_z'}.
$$

In child box frame $\phi = \boxed{d_r} \cdot \boxed{\bar{\phi}_{c2m}}$.
In the parent box frame

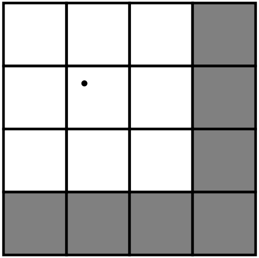
$$\phi' = \boxed{d_r' \cdot \sqrt{R}} \cdot \boxed{\phi_{m2m}} = d_r' \cdot \bar{\phi}_{m2m}$$

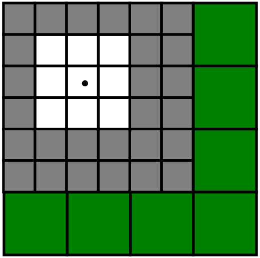
with $d_r' = \sqrt{d_x'^2 + d_y'^2 + d_z'^2}$,
and $\phi_{m2m} = \bar{\phi}_{c2m} \circ M_{m2m}$,
where $M_{m2m}$ is the map from the old DA variables into the new DA variables

# Single Level FMM

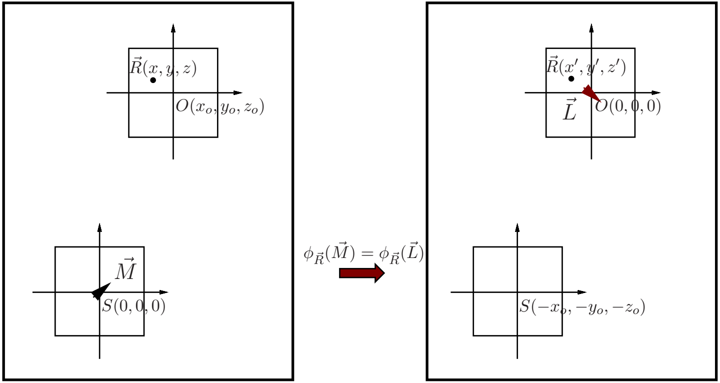


# Single Level FMM





Convert a multipole expansion into a local expansion



New DA variables in the observer frame

$$\begin{aligned} d'_x &= x - x'_o = x', \\ d'_y &= y - y'_o = y', \\ d'_z &= z - z'_o = z'. \end{aligned}$$

The relation between the new and the old DA variables

$$\boxed{\begin{aligned} d_x &= (x'_o + d'_x) \cdot R, \\ d_y &= (y'_o + d'_y) \cdot R, \\ d_z &= (z'_o + d'_z) \cdot R. \end{aligned}} \quad M_{m2l}$$

with

$$R = \frac{1}{(x'_o + d'_x)^2 + (y'_o + d'_y)^2 + (z'_o + d'_z)^2}.$$

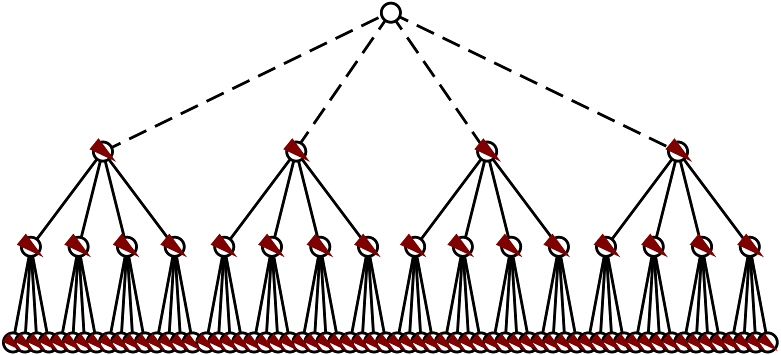
The multipole expansion in the source frame $\phi = \boxed{d_r} \cdot \boxed{\bar{\phi}}$
The local expansion in the observer frame

$$\phi = \boxed{\sqrt{R}} \cdot \boxed{\bar{\phi}_{m2l}} = \phi_{m2l}$$

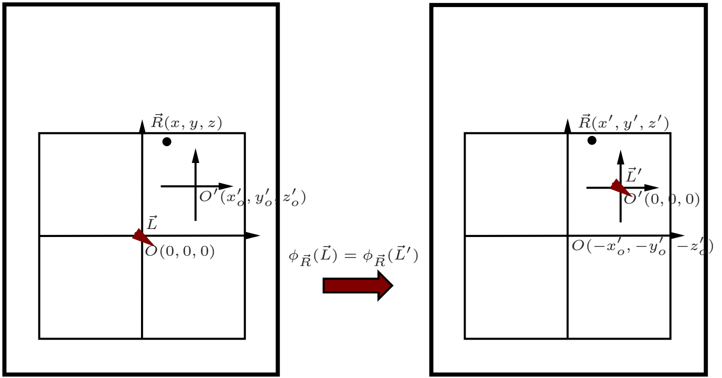
where $\sqrt{R}$ is converted from $d_r$,$\bar{\phi}_{m2l} = \bar{\phi} \circ M_{m2l}$,
and $M_{m2l}$ is the map between the DA variables.
Error

$$|\epsilon| \leq C \cdot \left(\frac{a}{r_o'}\right)^{p+1} \cdot \frac{1}{r_o' - a} + C \cdot \left(\frac{r'}{b}\right)^{p+1} \cdot \frac{1}{b - r'}.$$

# Single Level FMM



Translate a local expansion from a parent box to its child boxes



DA variables in the child box frame

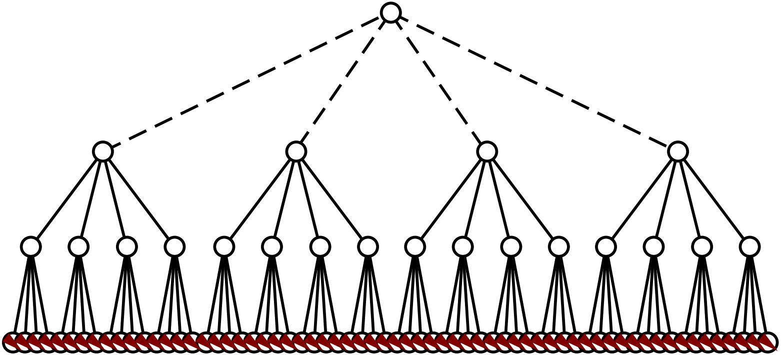
$$\boxed{\begin{array}{rcl} d_x &=& x_o' + d_x', \\ d_y &=& y_o' + d_y', \\ d_z &=& z_o' + d_z'. \end{array}} \quad M_{l2l}$$

The local expasion in the parent box frame is $\phi_{m2l}$.
The local expansion in the child box frame is

$$\phi = \phi_{m2l} \circ M_{l2l} = \phi_{l2l},$$

where $M_{l2l}$ is the map between the old and the new DA variables.

# Single Level FMM



# Single Level FMM

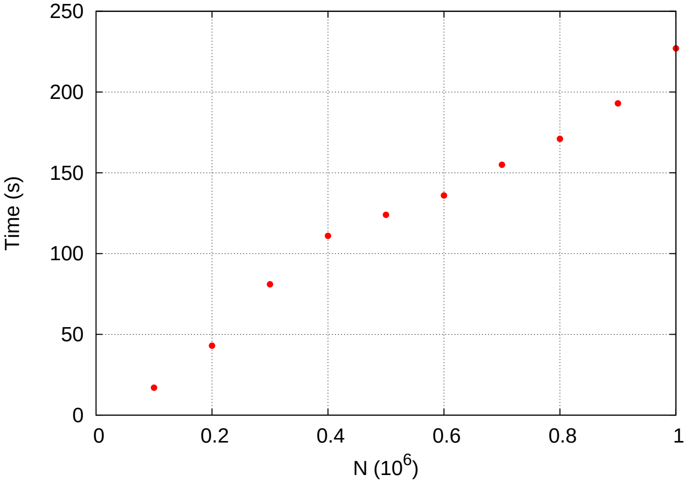
- Now we have the potential expressed as a polynomial of coordinates up to order $p$.
- Take the derivative of a coordinates to get the field expression in a polynomial of coordinates up to order $p - 1$.
- Submit the charge positions into the expression to calculate the potential/field.

# Single Level FMM

Description of the single level FMM

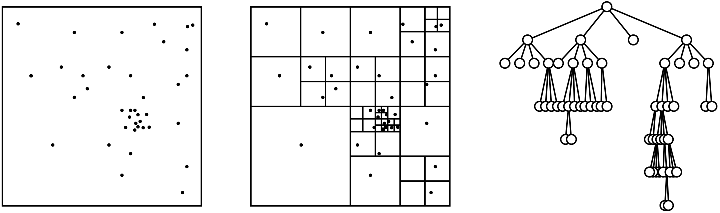
- Tree construction. Include all the charges in a box, then cut the box into small boxes until each childless box includes less than $s$ charged particles. Thus we get a heirarchical tree tructure of boxes.
- Upwards. Calculate the multipole expansion of childless boxes from charges, and then calculate the multipole expansion of parent boxes from child boxes.
- Downwards. For each box, calculate the local expansion from the multipole expansions in the interaction list. Then translate the local expansion of parent boxes into child boxes.
- Potential/Field calculation. For each childless boxes, calculate the far region field from the local expansion, and calculate the near region field directly by Coulumb theorem. Take the summation.

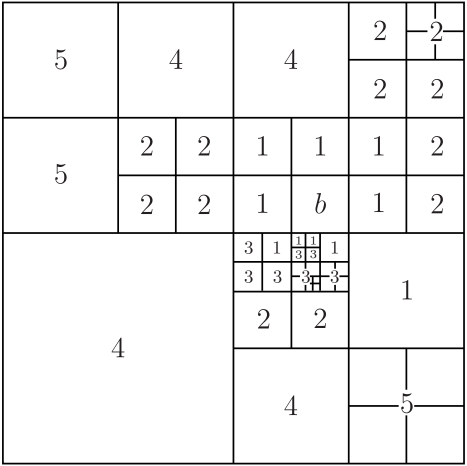
Efficiency scales with $O(N)$



## II.2 MULTIPLE LEVEL FAST MULTIPOLE ALGORITHM

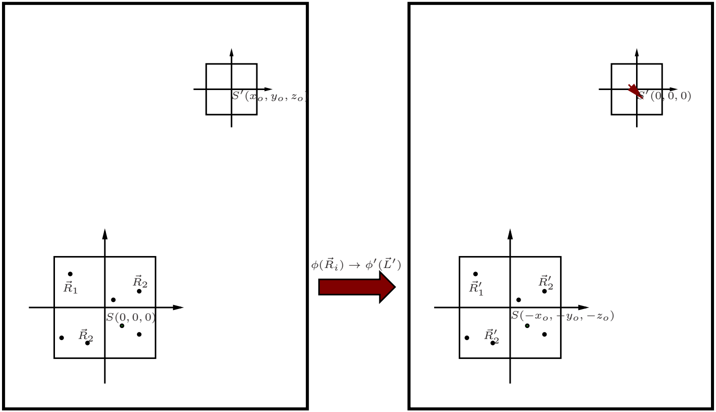
# MLFMA






| Relations | | Operations |
|---|---|---|
| $c \in 1_b$ | $b \in 1_c$ | $C_c \to C_b$, $C_b \to C_c$ |
| $c \in 2_b$ | $b \in 2_c$ | $M_c \to L_b$, $M_b \to L_c$ |
| $c \in 3_b$ | $b \in 4_c$ | $M_c \to C_b$, $C_b \to L_c$ |
| $c \in 4_b$ | $b \in 3_c$ | $C_c \to L_b$, $M_b \to C_c$ |
| $c \in 5_b$ | $b \in 5_c$ | Do nothing |

Calculate the local expansion from charges.



In the observer (small box) frame, the new DA variables are

$$\begin{aligned} d'_x &= x - x'_o = x', \\ d'_y &= y - y'_o = y', \\ d'_z &= z - z'_o = z'. \end{aligned}$$

Then the local expansion is

$$\begin{aligned} \phi_{\mathrm{L}} &= \sum_{i=1}^{n} \frac{q_i}{\sqrt{(x-x_i)^2 + (y-y_i)^2 + (z-z_i)^2}} \\ &= \sum_{i=1}^{n} \frac{q_i}{\sqrt{(x'_o - x_i + d'_x)^2 + (y'_o - y_i + d'_y)^2 + (z'_o - z_i + d'_z)^2}} \end{aligned}$$

Error

$$|\epsilon| \leqq C \cdot \left(\frac{r'}{b}\right)^{p+1} \cdot \frac{1}{b - r'}$$

Calculate the field from the multipole expansion.
The multipole expansion is $\phi = d_r \cdot \bar{\phi}$, then

$$
\begin{aligned}
E_x &= \{-\frac{\partial\bar{\phi}}{\partial d_x} \cdot (d_r^2 - 2d_x^2) + 2\frac{\partial\bar{\phi}}{\partial d_y} \cdot d_x d_y + 2\frac{\partial\bar{\phi}}{\partial d_z} \cdot d_x d_z + \bar{\phi} \cdot d_x\} \cdot d_r \\
E_y &= \{2\frac{\partial\bar{\phi}}{\partial d_x} \cdot d_y d_x - \frac{\partial\bar{\phi}}{\partial d_y}(d_r^2 - 2d_y^2) + 2\frac{\partial\bar{\phi}}{\partial d_z} \cdot d_y d_z + \bar{\phi} \cdot d_y\} \cdot d_r \\
E_z &= \{2\frac{\partial\bar{\phi}}{\partial d_x} \cdot d_z d_x + 2\frac{\partial\bar{\phi}}{\partial d_y} \cdot d_z d_y - \frac{\partial\bar{\phi}}{\partial d_z} \cdot (d_r^2 - d_z^2) + \bar{\phi} \cdot d_z\} \cdot d_r
\end{aligned}
$$

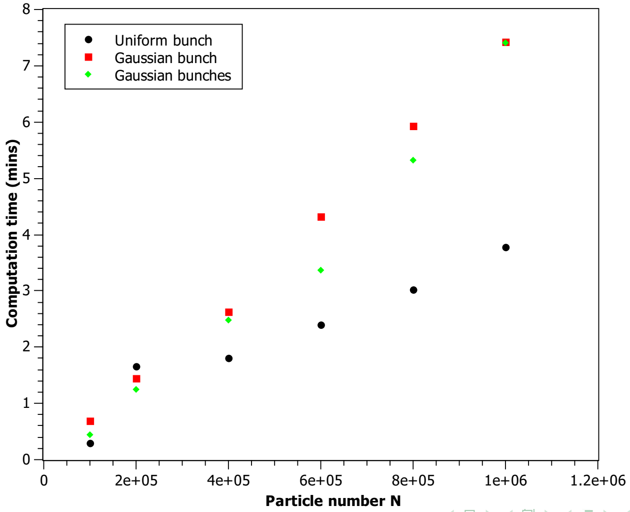
with

$$
d_r = \sqrt{d_x^2 + d_y^2 + d_z^2}.
$$

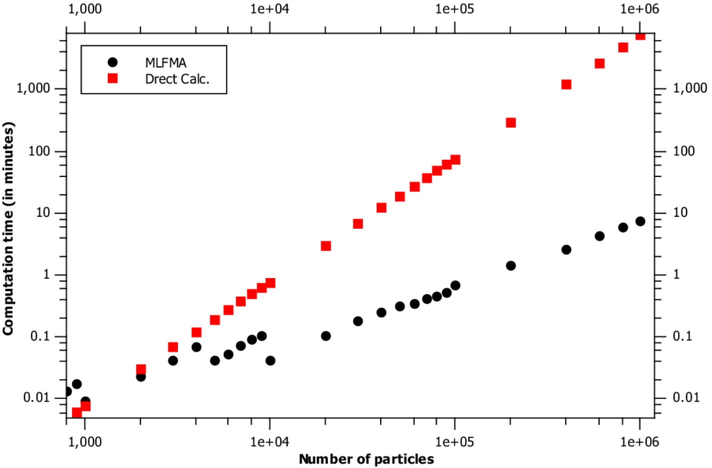
Description of the MLFMA

- Construct the hierarchical box structure (partial tree).
- Upwards: Calculate the multipole expansions for all the boxes.
- Downwards: For each box, check its the relation with other boxes and operate according to the above table. Then translate the local expansion from parent boxes to the child boxes.
- Calculate the potential/field, which comes from direct calculation and multipole or local expansions.

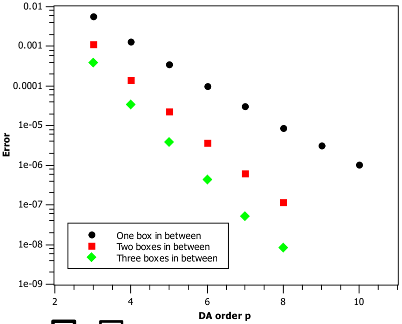
Computation time for different charge distribution

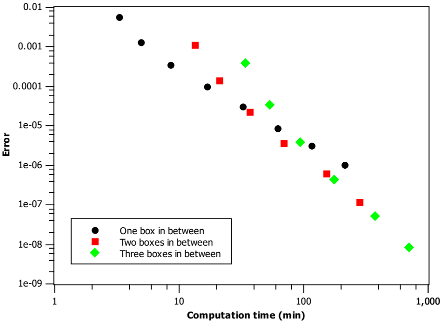


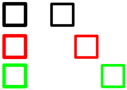
Compare the MLFMA with direct calculation



Accuracy increases with DA order

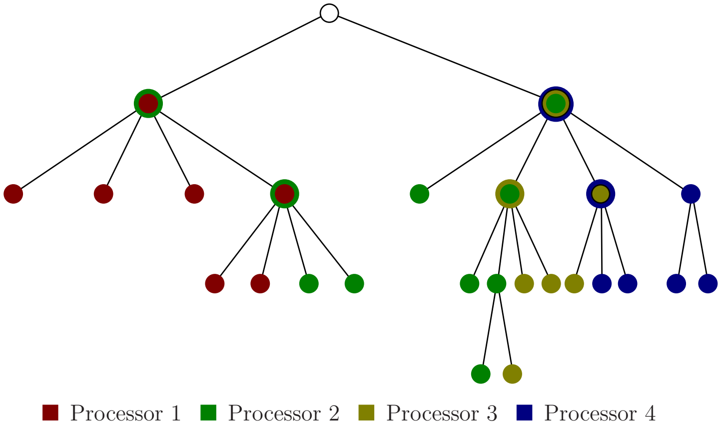


II.3 PARALLEL MULTIPLE LEVEL FAST MULTIPOLE ALGORITHM BASED ON ALL-TO-ALL COMMUNICATION

Distribute the tree

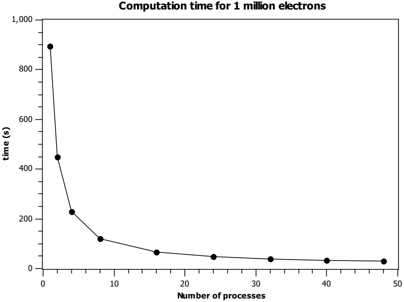


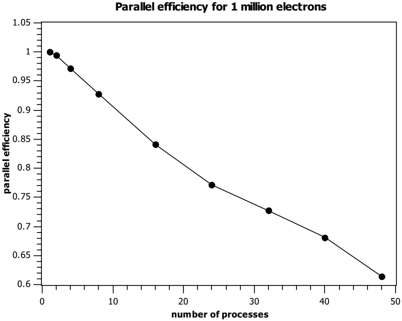
Description of PMLFMA

- Construct the compressed tree structure in parallel.
- Upward: Calculate the multipole expansion in parallel.
- Downward: Calculate the loacal expansion in parallel.
- Calculate the potential/field in parallel.
- Distribute the potential/field to the corresponding process.

Parallel Efficiency

$$E(N,P) = \frac{1}{P}\frac{T_{\text{seq}}(N)}{T(N,P)},\ \ N = 1,000,000.$$



Bottleneck of the current parallel algorithm: Memory usage!

- Calculate at most 10 million particles now.
- COSY Infinity 9.1 is 32bit program, Fortran 77/90
- Intel Fortran compiler has 2GB memory limit for 32 bit program
- Each process has no more than 2GB memory
- Number of particles and multipole expansions saved in the local memory is limited.

III USING PMLFMA IN SIMULATION

Photo emission process

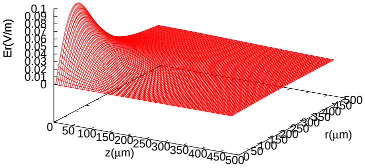
Assumptions

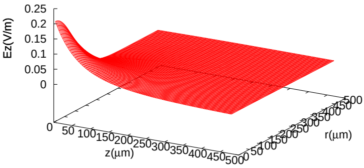
- Gaussian (FWHM $= 50$ fs) profile laser pulse applied on a surface, $E_{ph} = 4.65$ eV
- Originally electrons have Fermi energy ($E_f = 5$ eV), and they starts to move after getting energy from photons
- To overcome the work function ($W = 4.45$ eV), only those electrons inside a velocity cone can come out of the surface
- 200 million electrons come out, represented by 2 million macro-particles

Photo emission process

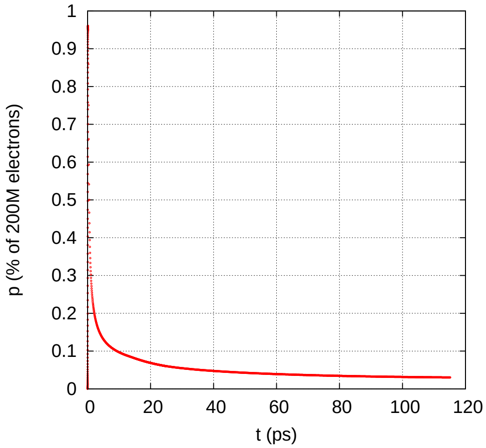
Field model

- Self-field between charged particles (MLFMA based on DA)
- Field of the positive hole (Gaussian distribution) on the surface
- Constant external field



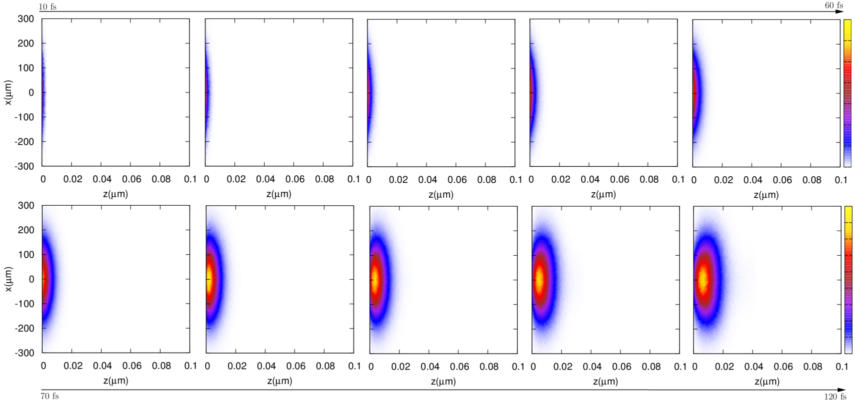



# Simulation



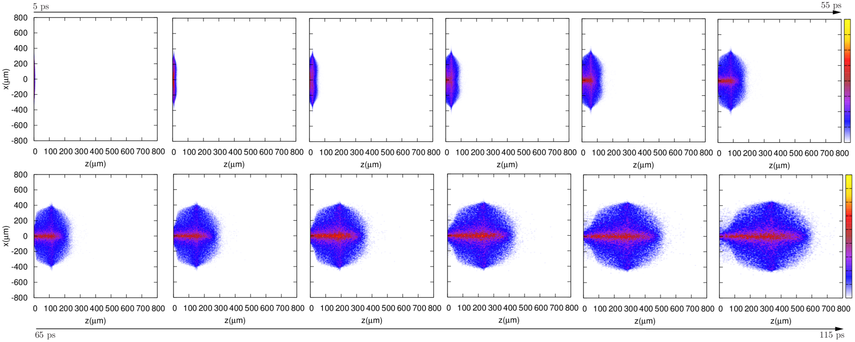
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
p (% of 200M electrons)
0
20
40
60
80
100
120
t (ps)

# Simulation

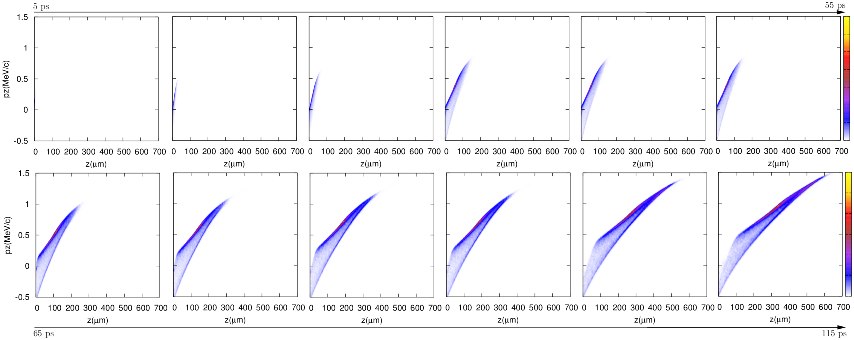


10 fs
60 fs
x(μm)
z(μm)
300
200
100
0
-100
-200
-300
0
0.02
0.04
0.06
0.08
0.1
70 fs
120 fs

# Simulation



# Simulation


5 ps
55 ps
1.5
1
0.5
0
-0.5
pz(MeV/c)
0 100 200 300 400 500 600 700
z(μm)
65 ps
115 ps

Summary

- Combined the FMM with DA for a new algorithm, sacles with $O(N)$.
- Single Level FMM works for uniform distribution.
- MLFMA works for arbitrary charge distribution.
- Parrallel MLFMA, 10 million.

Future work

- Keep polishing the algorithm.
- Boundary conditions.
- Simulation.
- Map method.


# THANK YOU!