

INRIA

HIEPACS TEAM

BORDEAUX

ScalFMM Periodic Model

Author:

Berenger BRAMAS

Supervisor:

June 11, 2013

Contents

1 Abstract	2
2 Introduction	2
3 State of the art	2
4 Usual Operators	2
4.1 Octree	2
4.2 Top Down Operators	3
4.3 Interactions operators	4
4.4 Global view of the algorithm	4
5 Virtual Repetition	5
5.1 Periodicity during normal pass	5
5.2 Periodicity at root level	6
5.3 Periodicity above the root	6
5.4 Adding a border	8
5.5 Limiting the periodicity direction	8
6 Theoretical repetition	9
6.1 Cost above the root	9
7 Implementation details	10
7.1 The algorithm	10
7.2 Low level details	10
8 Conclusion	11

1 Abstract

This study proposes a new approach to compute a boundary/periodic Fast Multipole Method (FMM) in any direction. The proposed model uses the standard FMM operators ($M2M$, $M2L$, $L2L$, $P2P$) which make it kernel independent. The algorithm can simulate a large repetition of the simulation box for a low computational cost.

Keywords: FMM, octree, boundary.

2 Introduction

The FMM is widely used to run physical simulations where pairs interactions, also known as nbody problem, can have a complexity reduced from $O(n^2)$ to $O(n)$. But many physical problems could also have the need to compute periodic simulations. Many analytical methods have been developed but none of them is using the power of the FMM to compute the periodicity.

3 State of the art

In APPLICATION OF FAST MULTIPOLE METHOD TO MICROMAGNETIC SIMULATION OF PERIODIC SYSTEMS, an approach is proposed but is not using the power of FMM and the usual shape of its operators.

4 Usual Operators

In this section we remind the common FMM operators in a algorithm view.

4.1 Octree

The FMM operators relies on the spatial decomposition of the problem which is done using an octree. The system, which can be particles for example, is included in a cube which is the root of the tree at level 0. Then we divide this root-cell by 2 in each dimension which creates 2^{DIM} sub-cells. We repeat the previous step until a level L . We have a maximum of $2^{DIM \cdot l}$ cells at level l . Some implementation

subdivides a cell until a condition is respected, like for example having less than N particles in each leaves. Such approaches leads to a tree where all leaves may not be at the same level, but our method can even be applied them.

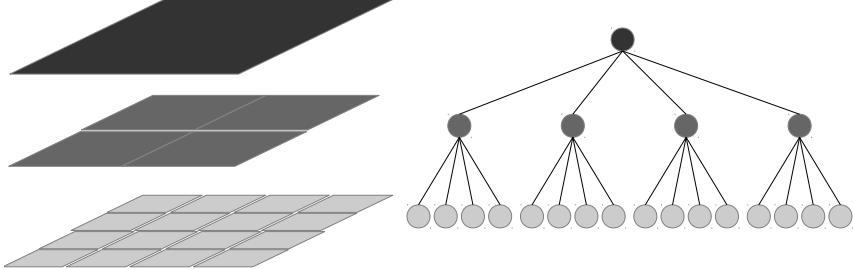


Figure 1: 2D Octree

4.2 Top Down Operators

These operators are performing computation between a cell and its children (the cells it contains one lower level in the tree) called $M2M$ and $L2L$, and inside leaves between cells and particles called $P2M$ and $L2P$. Some implementation defines optimizations which make them having operators that skips level during $M2M$ or $L2L$. But in most FMM it exists a $M2M$ (and a $L2L$) which is able to compute the interaction between a cells and its 2^{DIM} children. The children are positioned at equal distance from their common parent r and can be positioned using spherical coordinates: (r, θ, ϕ) , with $(\theta, \phi) \in \{\pi/4, 3\pi/4\} \times \{\pi/4, 3\pi/4, 5\pi/4, 7\pi/4\}$ in 3D and (r, ϕ) , with $(\phi) \in \{\pi/4, 3\pi/4, 5\pi/4, 7\pi/4\}$ in 2D. Depending on the level where the operator is applied, r is equal to half of a cube diagonal: $r = \text{Diagonal}/2 = \sqrt{3 * \text{Edge}^2}/2 = \sqrt{3 * (\text{BoxWidth}/2^l)^2}/2$.

4.3 Interactions operators

The operators that operate on one level and do not rely on the parent or sub-cells are the $P2P$ and the $M2L$. The $P2P$ operates between neighbors leaves. If the tree is full (each cell exists), each leaf has a number of neighbors ng with $2^{DIM} - 1 \leq ng \leq 3^{DIM} - 1$. The minimum is for leaves that are positioned in the corners. If the octree is not full then the minimum can be 0 since leaf may not have any neighbors.

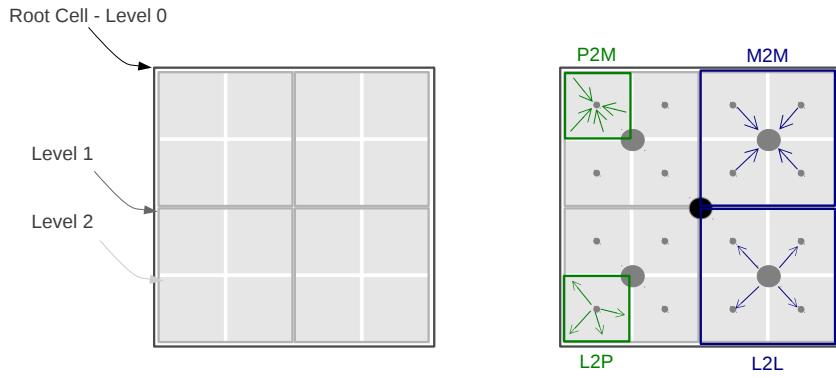


Figure 2: Top-Down Operators

The $M2L$ operates between a cell and what is called its interaction list. The interaction list is composed of cells from the same level and which are not neighbors but are closed. The closed properties is true for the cells which have their parents as neighbors respectively. So each cell has a unique interactions list. Depending on the position of a cell relatively to its parent, the interactions list is taken cells from $[-3; +2]$ or $[-2; +3]$ in each dimension. In $3D$ and with a full octree, each cell has a maximum of $6^3 - 3^3 = 189$ interactions. The 6^3 is coming from the cube of size 5 which is encapsulate all the potential interactions and the 3^3 represents the cell itself and its neighbors. Any FMM implementation has the capacity to compute the $M2L$ between two cell of the same level that are separated by at least one and a maximum of two in each direction.

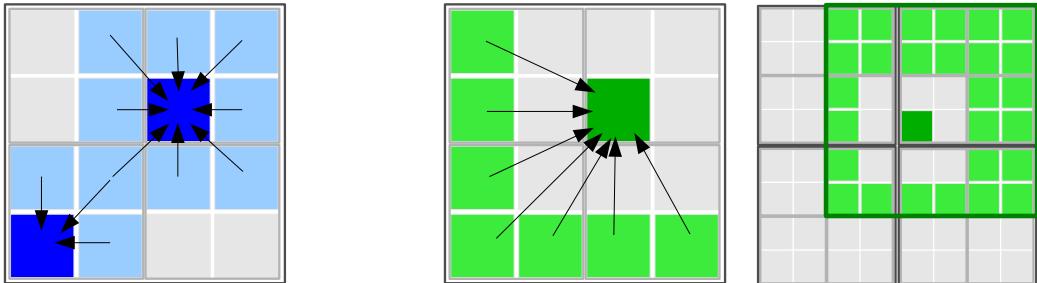


Figure 3: Interactions Operators, P2P (Blue), M2L (Green)

4.4 Global view of the algorithm

The FMM algorithm can be split in three major parts, an upward pass, a transfer pass and a downward pass. The upward pass is composed by the $P2M$ and the $M2M$, it centralized the information contained by each cell from leaf level to the top of the tree. The transfer pass, composed by the $M2L$, is translating the values at each level using the interactions lists. Then the downward pass, using $L2L$ and $L2P$, merge the results of the transfer pass and impact the leaves by starting from the root. In the non periodic FMM, there is no computation above level 2 since no $M2L$ happens at level 1 because the interactions lists are empty.

In addition of these steps called the far field, we compute the near field which is the $P2P$ operators.

5 Virtual Repetition

When using periodicity, a cell or a leaf can have its interactions or neighbors lists extended. In fact, when working with periodicity the simulation box should be repeated. A cell on the border or separated by one cell from the border should have interactions from the other side of the box. In the next paragraph we supposed when speaking about interactions or neighbors lists that they are a periodic lists.

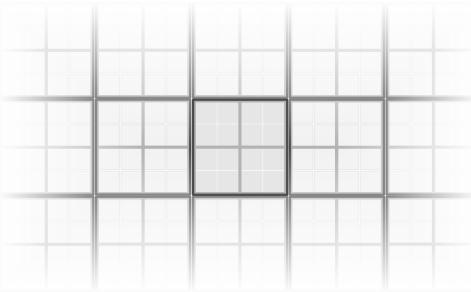


Figure 4: Repetitions of the simulation box create new neighbors

5.1 Periodicity during normal pass

During the upward pass the periodicity does not change the operators. In fact, they operate between cells and sub-cells.

Whereas, the transfer pass ($M2L$ and $P2P$) should not work with periodic lists. The normal FMM stops at level 2 because interactions lists are empty at level 1. By doing this, it makes a repetition of original box $rep(2) = 1/2$. If we use the upward pass, transfer pass and downward pass at level 1 then $rep(1) = 1$. In fact, at level 1 the periodic interactions lists are not empty and the $M2L$ at level 1 is covering the neighbors of our root. It is important to have in mind that without any special operation the box is already repeated once it we go at level 1 and use periodic lists.

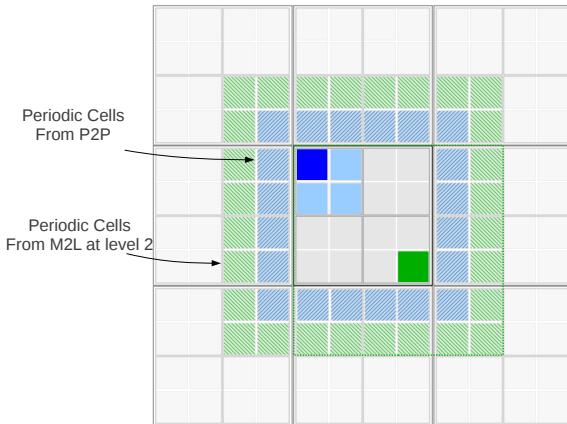


Figure 5: Extension of the box with a periodicity until level 2

5.2 Periodicity at root level

If the $M2M$ goes until the root at level 0, the root cell is containing all the information of the entire tree. Using this cell we can perform a $M2L$ from -3 to 3 by using it same cell inside the interactions lists. This $M2L$ will not compute the interaction with the neighbors cells, but this has been already done. We then have a repetition of 2 cells in each direction so it like having a grid of $(2 * 2) + 1$ and working on the cell in the middle. Finally we can compute the downward pass from level 0 to the leaves.

5.3 Periodicity above the root

In the previous paragraph we stopped the operation at level 0. But the key idea of the periodic algorithm is to continue with the usual operators above the level 0

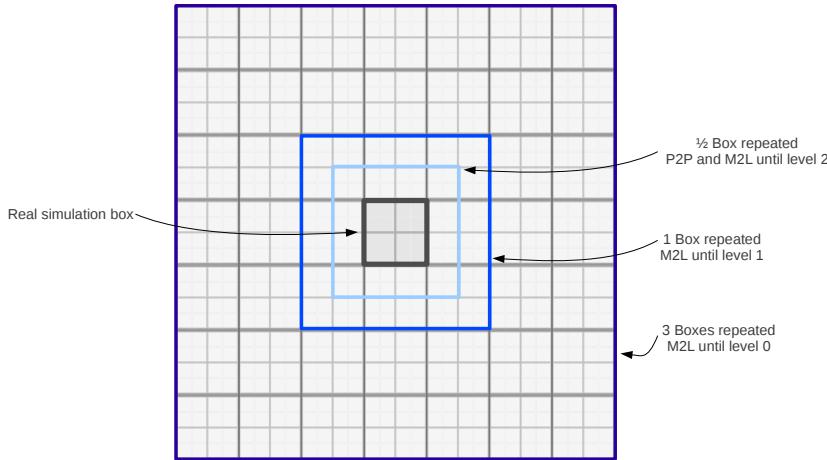


Figure 6: Extension of the box with a periodicity until level 0

of the tree.

If we compute the $M2L$ between $[-3; +3]$ at level 0 it is then impossible to continue with the usual operators. In fact, the $M2L$ is covering what sub-cells do not cover and should let too far cells to upper $M2L$. The $M2L$ at level 0 has to be limited to $[-3; +2]$ or $[-2; +3]$ in each direction.

This choice is made by choosing where does our current root is situated relatively to its parent. Any choice can be made, but we consider that the root cell is at position $(1, 1, 1)$ relatively to its parent. So when performing the $M2L$ at level 0 the repetition of the box is not odd and so it is not equilibrated. It is similar as having a grid of width 6 and working on the cell of coordinate $(4, 4, 4)$. We can continue to perform similar operations, $M2M$ followed by $M2L$ and considering that our working cell is in $(1, 1, 1)$ relatively to its parent.

Then, at the last level above the root, the final $M2L$ should be performed by considering that the working cell is in position $(0, 0, 0)$ relatively to its parent. This puts our working cell in the middle of the grid.

It remains that the repetition is not symmetric. In fact, the grid has a width of $(6 * 2^d)$ which is odd. There are $(6 * 2^d)/2$ repetitions in one side and $((6 * 2^d)/2) - 1$ in the other side for all the dimensions. To have a perfect symmetric periodicity we need to add a border.

We can group the root cell to perform a $M2M$ from level 0 to level -1 . The results contain the information of a box which is nothing more than our original

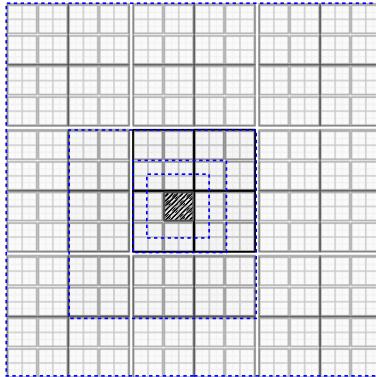


Figure 7: Extension of the box with a periodicity until level -1 without border

box repeated 2 times in each direction. Again, we can do a $M2L$ at level -1 and we choose that our cell is at position $(0, 0, 0)$ relatively to its parent. These operations can be repeated until we have enough repetition.

Every time we perform a $M2M$ d levels above 0 we create a new cell which contain 2^d times our original box. Then when we perform a $M2M$ at a level we repeat the cell from -2 to 3 in any direction. So at d levels above 0 performing a $M2L$ leads to create a repetition of $6 * 2^d$ original box in all the directions. So we create a grid of size $(6 * 2^d)^3$ and we are working on the center cell at position $((6 * 2^d)/2, (6 * 2^d)/2, (6 * 2^d)/2)$.

5.4 Adding a border

After having proceed d levels above 0, a border of the simulation box is missing. To take into consideration this border, we need to use the root cell and create all the configuration in order to have create cells one levels above d which contains all the information. Then we can apply a final $M2L$ between those cells and our working cell.

In $2D$ there are three kinds of border cells: the horizontal border, the vertical border and the angle. In $3D$ there are seven kinds of border cells: the top, the side, the back and the four different links between them. More generally, there are DIM different sides and 2^{DIM-1} different connexion between sides.

Computing the cell which represent a side is done in $d - 1$ $M2M$. Each of this $M2M$ is computing the relation between one or two sub-cells and their parent.

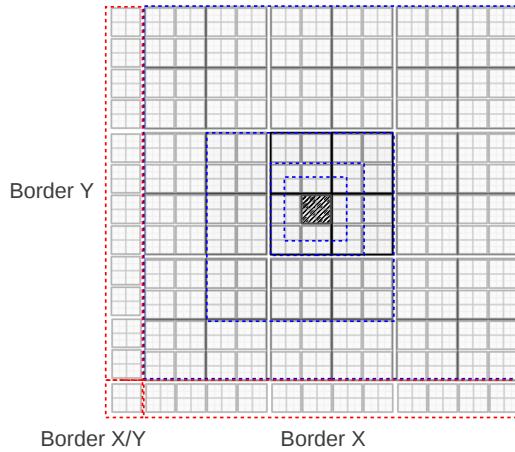


Figure 8: Extension of the box with a periodicity until level -1 with border

5.5 Limiting the periodicity direction

If the simulation needs that the periodicity is limited in some direction, for example $(+X, + - Y, No Z)$, the only changes is in the construction to the periodic lists including those made above the root. Then, during the addition of the border only the periodic directions have to be applied.

6 Theoretical repetition

The periodic-algorithm does not allow to choose a specific grid size. The only parameter D is the number of level above the root of the box where the algorithm is applied. As it has been illustrated the final grid size has a width of $dim = 3 * 2^{Abs(D)+1} + 1$

D	Grid Dim
1	3
0	7
1	13
5	193
20	6 291 457

6.1 Cost above the root

We determine the cost of the periodicity in term of FMM operators. The following cost is based on a full octree.

There are $2^{DIM} M2M$, 1 $L2L$, and $6^{DIM} - 3^{DIM} M2L$ per level above the root. For the border the cost is $\sum_{i=1}^{DIM} i^2 M2M$ per level above root.

For example, in 3D, the cost for $D = 5$ is $5 * (2^3 + 1^2 + 2^2 + 3^2) = 110 M2M$ and $5 * 189 = 945 M2L$. For example, in 3D, the cost for $D = 20$ is $20 * (2^3 + 1^2 + 2^2 + 3^2) = 440 M2M$ and $20 * 189 = 3780 M2L$.

7 Implementation details

7.1 The algorithm

The algorithm above the root can be spited in 4 steps:

- Upward pass, for level 0 to $-d$ we compute the $2^{DIM} M2M$ (8 per level in 3D).
- Transfer pass, for level 0 to $-d$ we compute the $6^{DIM} - 3^{DIM} M2L$ (189 per level in 3D). For level 0 to $-d + 1$ we put the cell at position $(1, 1, 1)$ relatively to its parent. For the last $M2L$ at level $-d$ we put the cell at position $(0, 0, 0)$ relatively to its parent.
- Border pass, at level $-d - 1$ we compute the border and apply it using $M2L$.
- Downward pass, from level $-d - 1$ to 0 we perform one $L2L$. There is only one sub-cell per $L2L$ and the position has to be the same as the one used during the Transfer pass.

7.2 Low level details

In our library, all the main part of the FMM are dissociated. To ensure that our kernel is able to compute all the periodic operation, we pass to our kernel initialization method an offset for the box with and for the tree height. Our kernel compute the requested operation without any difference between the real FMM and the operation above the root for the periodic FMM. In fact, let the original

FMM box width be w and a tree height be h . Then, if is required to apply periodicity until level d above the root. The kernel should be able to proceed usual FMM operator in a tree of height of size $h + d + 5$ with a initial box width of $w * 2^{d+3}$.

8 Conclusion

A new approach has been proposed to compute a periodic FMM. This approach can be used with any kernel and without the need of developing new operators. Moreover, it is possible to compute an important repetition size for a small cost.