Error-Controlled Hybrid Adaptive Fast Solver for Regularized Vortex Methods

Samer Salloum\textsuperscript{a}, Issam Lakkis\textsuperscript{a,}\textsuperscript{*}

\textsuperscript{a}Department of Mechanical Engineering, American University of Beirut, Beirut, Lebanon

Abstract

In this paper, an error-controlled hybrid adaptive fast solver that combine both $\mathcal{O}(N)$ and $\mathcal{O}(N \log N)$ scheme is proposed. For a given accuracy, the adaptive solver is used in the context of regularized vortex methods to optimize the speed of the velocity and vortex stretching calculation. This is accomplished by introducing three critical numbers in order to limit the depth of the tree division and to balance the near-field and far-field calculations for any hardware architecture. The adaptive solver is analyzed in term of speed and accuracy.

Keywords: Vortex Methods, Fast Multipoles, other

*Corresponding author Tel: 961350000, ext 3636, Email: il01@aub.edu.lb
1. Introduction

Vortex methods are mesh-free methods used to solve the Navier-Stokes equation by tracking the motion of vorticity-carrying fluid elements. In these methods, the calculation of the velocity and vortex stretching vectors at each element location is done through direct summation over all vortex elements. Thus for \( N \) particles simulation, the direct evaluation of the all pairs interaction requires \( O(N^2) \) computational operations. For large \( N \), the cost become prohibitive even with the use of fast computers [10]. This high cost led scientist to develop approximate solutions for the \( N \) body problems.

Historically, scientists have developed two variants of hierarchical \( N \)-body methods, Treecodes and FMMs [8]. The Treecode algorithm [1] reduces the complexity to \( O(N \log N) \) by clustering source particles into progressively larger groups and using multipole expansions to approximate their influence on each target particle. The FMM [6] can achieve \( O(N) \) by clustering not only the source particles, but also the nearby target particles using local expansions to approximate their influence at the targets locations [10]. These two methods have followed very different paths of evolution and differs from each other in several way [4, 5, 10, 9]:

1. For the far field, Treecodes performs cell-particle interactions leading to \( O(N \log N) \) complexity, while FMM performs cell-cell interactions leading to \( O(N) \) complexity
2. Treecodes have conventionally used hard coded cartesian expansions with a fixed order of expansion \( p = 3 \) while FMMs have used spherical harmonic expansions of an arbitrary order
3. Treecodes use the ratio between the size of cells and the distance between them to construct the interaction list. This is known as the multipole acceptance criterion (MAC), and it is used to determine if a cell should be evaluated or subdivided further. In contrast, the FMM uses parent, child, and neighbor relationships to construct the interaction list [10]
4. In Treecodes, the accuracy is controlled by the MAC parameter \( \theta \), while in FMMs, accuracy is controlled by the order of expansion \( p \)
5. In Treecodes, the expansions are centered on the cells' centers of mass rather than geometrical centers in FMMs [5]
6. Treecodes are adaptive in nature and permits interaction of cells at different level. FMMs employ a (usually) non-adaptive structure of hierarchical grids and considers only interactions between nodes on the same grid level according to their geometrical neighborhood [4]

There have been some efforts from both the Treecode and FMM community to take the best of these two methods and produce an optimal algorithm [9]. In 1995, Michael Warren and John Salmon [7] developed a new technique to calculate cell-cell interactions using Taylor series in Cartesian coordinates. Yet they didn’t implement cell-cell interaction in the Treecode framework. In 1999, Cheng et al. [3] extended the work of Greengard [6]
and introduced a more adaptive cell-cell interaction stencil that considers the interaction of cells at different levels in the tree. They also introduced a mechanism to select between cell-particle and cell-cell interactions in FMMs, which is always faster than a pure Treecode or pure FMM. In 2000, Walter Dehnen [4] extended the Treecode method introduced by Barnes and Hut [1] for fast evaluation of gravitational forces by including mutual cell-cell interaction developed in [7]. He also introduced a symmetric multipole-acceptance criterion (MAC) and use it to determine whether a given cell-cell interaction can be executed or the cell must be split. In 2002, Walter Dehnen [5] presented an improvement of his original code by introducing other techniques such as generic tree traversals, mutual cell-cell interactions, and a mass-dependent error-controlled multipole acceptance criteria. Walter code is divided into two phases, namely the interaction phase and the evaluation phase. In the interaction phase, for each cell, the Taylor series coefficients of all its interactions (cell that interact via Multipole criteria) are evaluated and accumulated. In the evaluation phase, for each body, the Taylor series of all relevant cells (those that contain the body) are accumulated by first translating to a common expansion center and then adding coefficients. Walter compared his code directly with the 3D adaptive FMM code by Cheng et al. [3]. On average his code is faster by more than a factor of 10 and twice as accurate. However, his code cannot compete with the Cheng et al. FMM in the regime E (10^{-6}). Walter concluded that traditional FMM is less useful in the low-accuracy regime, such as is needed in stellar dynamics, in agreement with earlier findings [2]. He argued that the reason is that his code increases accuracy by decreasing the MAC number (p is always constant) while FMM increases accuracy by increasing p. He finished by saying that a code for which p and θ can be adapted simultaneously would be superior to both codes. In 2012, Yokota and Barba [10] developed a hybrid Treecode-FMM algorithm that has a similar structure to Dehnen’s method and has control over both the order of expansion and MAC. Their code is based on spherical harmonic expansions, but have the capability to switch to Cartesian expansions if the required accuracy is lower than a certain threshold. In addition they incorporate the capability to auto-tune the kernels on heterogeneous architectures. The key for auto-tuning is to time all the kernels. The algorithm uses this information to select the optimum kernel during the dual tree traversal, choosing between cell-cell, cell-particle, and particle-particle interactions. Yokota and Barba tested their code by fixing the maximum number of elements per leaf to 50 and 200 respectively. For a well chosen critical number (N = 200), the results indicate that the hybrid method always favors cell-cell interactions and doesn’t provide a visible advantage over the pure FMM. However for (N = 50) the FMM suffers from load imbalance between the near-field and far-field evaluations, while the hybrid method doesn’t because it can choose to perform particle-particle interactions even if the cell is not at the leaf level. They concluded that the hybrid method produced a better result by automatically fine tuning the balance between the particle-particle and cell-cell interactions throughout the adaptive tree. It optimizes the balance between particle-particle and cell-cell interactions and achieves optimum performance for all N. There’s no need to tweak parameters (such as particles per cell) to achieve optimal performance. The hybrid
method seems to remove the wavy behavior of the $N$-dependence in pure FMMs [11]. In 2013, Yokota and Barba [9] investigated the effect of changing both the multiple acceptance criteria $\theta$ and the order of expansion $p$ to achieve maximum performance for a given accuracy on both CPUs and GPUs. They have shown that it is never efficient to increase $\theta$ to a value that is larger than that of standard FMM codes ($\theta = 1/2$), for both CPU and GPU implementations. Therefore, when designing a hybrid Treecode-FMM algorithm it is not necessary to have a variable MAC which would have an optimum value of $1/2$. In this case, the hybrid scheme will produce similar interaction list as the one described by Cheng et al. (1999), but the implementation for the hybrid scheme is much simpler and the algorithm itself is more general and flexible.

The hybrid scheme developed by Yokota and Barba was able to tune the balance between the particle-particle and cell-cell interaction when FMM suffers from load imbalance. By using pre-measured kernel, Yokota and Barba also manage to solve the hardware dependence problematic when heterogeneous architectures come into the picture. However, the fact that the maximum number of elements per cell is introduced by the user will pose some problems. If the maximum number of elements per leaf was much higher than the optimal number, Yokota and Barba scheme won’t be able to balance cell-cell and particle-particle interactions. In fact, although the hybrid scheme will always favors cell-cell interactions for non-neighboring cells, the particle to particle interaction will take most of the simulation time since we have a large number of elements within neighboring cells. The interaction between neighboring cells cannot be evaluated using approximate methods since this will introduce an unbounded errors. Moreover, if the maximum number of elements per leaf was much smaller than the optimal number, and since Yokota did not introduce any mechanism to limit the depth of the tree structure, the hybrid scheme will manage to balance cell-cell and particle-particle interaction, but part of the simulation time will be spent to evaluate multipole and local expansions at the center of cell of high level, although these multipole and local expansions are not needed to evaluate the approximate field.

On the other hand, Cheng did not give any particular attention to different hardware architectures and he didn’t present any effective mechanism to limit the Oct-tree depth. In his paper [3], Cheng introduced four different adaptive list associated with each cell and uses particle-particle interaction if the number of elements in a given source cell is less than $p^2$. Moreover, the maximum number of elements per cell is also introduced by the user. This will yield the same problems described in the hybrid scheme case.

In treecodes and FMMs, the maximum number of particles per cell $N_{crit}$ is set (by the user) to balance the loads of near-field and far-field evaluations. Setting this number to be too small will result in a deep tree structure with a disproportionately large amount of far-field and too few near-field evaluations. Conversely, if $N_{crit}$ is set to be too large, the resulting tree structure will be too shallow and a large amount of time will be spent in the near-field evaluation [10]. In this paper, we introduce two different critical numbers, $n_T$
and \( n_F \). These critical numbers are used in the adaptive Cheng FMM framework to limit the depth of the tree structure in the upward and downward pass. A variant of these two numbers are also used to determine the type of interactions that need to be evaluated in order to tune the balance between cell-cell, cell-particle and particle-particle interactions. These two critical numbers are evaluated as a function of the ratio of time pre-measured kernel and differs from one architecture to another, thus we manage to solve the hardware dependence problematic in Cheng algorithm. The user does not need to worry about the maximum number of elements per leaf because it is chosen automatically by the code to ensure FMM balance. These critical numbers vary with the order of expansion \( p \), the choice of the cutoff function, the vector fields that need to be evaluated at each time step and the hardware architecture. For a given level of accuracy, which is associated with an order of expansion \( p \), the adaptive solver evaluate \( n_T \) and \( n_F \) and uses these two critical numbers to minimize the amount of computational work needed to evaluate the truncated singular expressions of the velocity and vortex stretching vectors. However, the use of a regularization core function will introduce an additional error since these same expansions are used to approximate the far-field regularized velocity and vortex stretching vectors. This additional error is related to choice of the cutoff function and mainly depends on the size of the smallest box obtained upon tree division. Hence, and in order to maintain the level of accuracy in regularized vortex methods, we introduce a critical level \( l_c \), which is equal to the maximumum level that can be reached upon tree division for a given level of accuracy.

2. Vortex methods and the Vorticity Transport equation

For three dimensional incompressible flow, \( \nabla \cdot \vec{u} = 0 \), the evolution of the vorticity of a fluid particle is described by the Helmholtz’s vorticity equation:

\[
\frac{d\vec{\omega}}{dt} = (\vec{\omega} \cdot \nabla)\vec{u} + \frac{\nabla \rho \times \nabla P}{\rho} + \nabla^2 \vec{\omega}
\]  

(1)

For an incompressible flow in unbounded domain, the velocity vector field \( \vec{u} \) in equation 1 is computed as the curl of a vector potential field \( \vec{\psi} \)

\[
\vec{u} = \nabla \times \vec{\psi}
\]  

(2)

The vorticity vector field \( \vec{\omega} \) in equation 1 is computed as the curl of the velocity vector field

\[
\vec{\omega} = \nabla \times \vec{u} = \nabla \times (\nabla \times \vec{\psi})
\]  

(3)

so that \( \vec{\psi} \) solves the Poisson equation

\[
\nabla^2 \vec{\psi} = -\vec{\omega}
\]  

(4)
Noting that the Green’s function for the Poisson equation \(\nabla^2 G = -\frac{1}{4\pi r}\) in an unbounded domain is \(G(x) = -\frac{1}{4\pi \|x\|}\) we obtain

\[
\vec{\psi}(\vec{x}, t) = \frac{1}{4\pi} \int \frac{\vec{\omega}(\vec{y}, t)}{\|\vec{x} - \vec{y}\|} d^3y \tag{5}
\]

Assuming a singular representation of the vorticity field,

\[
\vec{\omega}_\delta(\vec{x}, t) = \sum_i \vec{\alpha}_i \delta(\vec{x} - \vec{x}_i), \tag{6}
\]
the vector potential field and the velocity vector field can be approximated as

\[
\vec{\psi}_\delta(\vec{x}, t) = \frac{1}{4\pi} \sum_i \frac{\vec{\alpha}_i(t)}{\|\vec{x}(t) - \vec{x}_i(t)\|} \tag{7}
\]

\[
\vec{u}_\delta(\vec{x}, t) = \nabla \times \vec{\psi}_\delta = -\frac{1}{4\pi} \sum_i \frac{(\vec{x}(t) - \vec{x}_i(t))}{\|\vec{x}(t) - \vec{x}_i(t)\|^3} \times \vec{\alpha}_i(t) \tag{8}
\]

Assuming a regularized representation of the vorticity field,

\[
\vec{\omega}_\sigma(\vec{x}, t) = \zeta_\sigma(\vec{x}) \ast \vec{\omega}(\vec{x}, t) = \sum_i \vec{\alpha}_i(t) \zeta_\sigma(\vec{x} - \vec{x}_i) \tag{9}
\]
the vector potential field and the velocity vector field can be approximated as

\[
\vec{\psi}_\sigma(\vec{x}, t) = \sum_i G_\sigma(\vec{x} - \vec{x}_i) \vec{\alpha}_i(t) \tag{10}
\]

\[
\vec{u}_\sigma(\vec{x}, t) = \nabla \times \vec{\psi}_\sigma = -\sum_i q_\sigma(\vec{x}(t) - \vec{x}_i(t)) \frac{(\vec{x}(t) - \vec{x}_i(t))}{\|\vec{x}(t) - \vec{x}_i(t)\|^3} \times \vec{\alpha}_i(t) \tag{11}
\]
\(\zeta_\sigma\) is a radially symmetric regularization function and \(\sigma\) is its associated smoothing radius, \(G\) solves the Poisson equation \(\nabla^2 G = -\zeta(r)\), \(G_\sigma(\vec{r}) = \frac{1}{\sigma} G(\frac{\vec{r}}{\sigma})\), \(q(r) = \int_0^r \zeta(\rho)\rho^2 d\rho\), and \(q_\sigma(\vec{r}) = q(\frac{\vec{r}}{\sigma})\).

To accurately simulate three dimensional flows, a very large number particles is required to resolve the various length scales. Direct evaluation of the velocity and vortex stretching vectors requires \(O(N^2)\) computational operations of all the pairwise interaction in a system of \(N\) particles. The prohibitive cost of the direct evaluation renders the fast multipole method (FMM) an essential tool to execute particle-based simulations in a reasonable amount of time.
3. The Fast Multipole in three dimensions - Adaptive Scheme

The computational cost of the $N$ body problem is tremendously reduced by arranging nearby source particles into a recursively smaller boxes and using multipole expansion coefficients $\vec{M}_n^m$ in order to approximate the far field components of the velocity and vortex stretching vectors induced by all sources within the source box. Further reduction is achieved by clustering nearby target elements, in addition to source elements, and using local expansion coefficients $\vec{L}_n^m$ to approximate the aforementioned vectors at any location within a target box induced by all distant sources in the computational domain. The expansion is based on spherical harmonics $Y_n^m$.

Assuming $s$ vortices with strength $a (\vec{\alpha}_j, j = 1...s)$ located at the points $\vec{Q}_j = (\rho_j, \theta_j, \varphi_j)$ inside the sphere $D_Q$, with $|\rho_j| < a$ (see Figure 1 left), then at any $\vec{P} = (r, \theta, \varphi)$ with $r > a$, the vector potential is approximated by the following multipole expansion

$$\vec{\psi}\delta(\vec{P}, t) \simeq \frac{1}{4\pi} \sum_{n=0}^{p} \sum_{m=-n}^{n} \frac{\vec{M}_n^m}{r^{n+1}} Y_n^m(\theta, \varphi)$$  \hspace{1cm} (12)

with

$$\vec{M}_n^m = \sum_{j=1}^{s} \vec{\alpha}_j \rho_j^n Y_n^{-m}(\theta_j, \varphi_j)$$  \hspace{1cm} (13)

where the coordinates of the target point $\vec{P} = (r, \theta, \varphi)$ are taken with respect to an orthonormal system centered at $Q$.

![Figure 1: Schematic for the multipole and local expansions.](image)

Furthermore, suppose that $s$ vortices having strengths $a (\vec{\alpha}_j, j = 1...s)$ are located inside the sphere $D_Q$ of radius $a$ with center at $Q = (\rho, \alpha, \beta)$ (see Figure 1 right), and that
\[ \rho = (c + 1)a \text{ with } c > 1, \text{ then for any point } P(r, \theta, \phi) \text{ inside } D_0 \text{ of radius } a \text{ centered at the origin, the vector potential due to vortices } (\vec{\alpha}_j, j = 1...s) \text{ inside } D_Q \text{ is approximated by the following local expansion} \]

\[ \vec{\psi}_b(P, t) \simeq \frac{1}{4\pi} \sum_{n=0}^{p} \sum_{m=-n}^{n} \vec{L}_m Y_n^m (\theta, \phi) r^n, \]  

(14)

where

\[ \vec{L}_m = \sum_{j=0}^{\infty} \sum_{k=-j}^{j} \frac{M_j^{k, j} \cdot \rho^{k-m} \cdot A_n^{m} \cdot A_{n-j}^{m} \cdot \rho^{j+n+1} \cdot r^n}{(-1)^j A_{n+j}^{k-m} \cdot (\alpha, \beta)}, \]  

(15)

with \( A_n^m \) defined by

\[ A_n^m = \frac{(-1)^n}{\sqrt{(n-m)!(n+m)!}}. \]  

(16)

Simple expressions for the velocity and vortex stretching vectors in term of multipole and local expansion coefficients are derived in \[. \] These expressions are used in the context of an hybrid adaptive fast multipole scheme to approximate the far-field vectors.

In order to describe and analyse the adaptive solver, we will use the following notation:

- \( \mathcal{N}_{n_D}(I_l) \): For a given target box \( I_l \) at level \( l \), a source box \( J_l \) at the same level belong to the neighborhood of \( I_l \), \( \mathcal{N}_{n_D}(I_l) \), if and only if the distance along \( x, y, \) and \( z \) between the center of the two boxes is less than or equal to \( n_D W_{J_l} \) where \( n_D \) is a positive integer and \( W_{J_l} \) is the width of box \( J_l \).

- \( \mathcal{T}_{n_D}(I_l) \): For a given target box \( I_l \) at level \( l \), the Interaction List \( \mathcal{T}_{n_D}(I_l) \), is the set of all boxes which are children of Neighbors of \( I_l \)'s parent and are not neighbors of \( I_l \) (green boxes in Figure 4). For \( n_D = 1 \), we will have similar interaction list as the one described by Cheng et al. (1999) and by the hybrid scheme described by Yokota for an optimum MAC value of \( \frac{1}{2} \).

- \( \mathcal{H}_{n_D}(I_l) \): For a given target box \( I_l \) at level \( l \), the Inherited List \( \mathcal{H}_{n_D}(I_l) \), is the set of all boxes that belong to the interaction list of \( I_l \) ancestor, and for which the adaptive solver choose not perform cell to cell interaction. As her name suggest, this list is transmitted from a parent box to its children.

- \( \mathcal{C}_{n_D}(I_l) \): For a given target box \( I_l \) at level \( l \), \( \mathcal{C}_{n_D}(I_l) \), is the set of all boxes that belong to the interaction list of \( I_l \) ancestor, and for which the adaptive solver choose to evaluate their influence through cell to cell interaction.

- \( \mathcal{M}_{n_D}(I_l) \): For a given target box \( I_l \) at level \( l \), \( \mathcal{M}_{n_D}(I_l) \), is the set of all source boxes for which the adaptive solver chooses to evaluate their influence through cell to particle interaction.
The adaptive scheme consists of the following steps:

1. Construct the smallest cubic box that contains all the vortex elements in the computational domain. This box is called the root box.
2. Construct a hierarchy of boxes by recursively dividing the root box uniformly into smaller boxes. Refinement level 0 is equivalent to the root box. Refinement level \( l + 1 \) is obtained by subdivision of each box at level \( l \) into eight equally sized children. The division process stops whenever the box level \( l \) is equal to the critical level \( l_c \) or the number of elements in the box is less \( n_C = \min\{n_T, n_F\} \), where \( n_T \) and \( n_F \) are defined in section 4. A box that has no children is called a leaf box.
3. Initial expansion: At the finest level, all sources are expanded at their box centers to obtain the far-field multipole expansion coefficients \( \vec{M}_m^n \) using equation (13).
4. Upward pass: The multipole expansion coefficients for each box are translated via multipole-to-multipole translations from the source box centers to their parent source box center. All these translations are performed in a hierarchical order from bottom to top via the octree.
5. Downward pass: Form local expansion coefficients at the center of each box at every mech level \( l \geq 2 \). These local expansion coefficients are calculated by the following recursive manner: local expansion coefficients for all boxes at level 1 are first set to zero. For any target box \( I_l \) at level \( l \geq 2 \) that contains \( N_T \) targets, initial local expansion coefficients are obtained by translating local coefficients from the center of \( I_l \) parent to \( I_l \) center. For any Source box \( J_l \) that belongs to the interaction list of \( I_l \) (green boxes in Figure 4) and contains \( N_S \) sources, if \( N_T \) is greater than \( \frac{n_T^2}{8n_T} \) and \( N_T \times N_S \) is greater than \( \frac{n_T^2}{64} \), the multipole coefficients at the center of \( J_l \) are converted.
into local expansion coefficients at the center of box \( I_l \) using equation (15) and then added to the initial local coefficients. Otherwise, \( J_l \) is set in the Inherited list of \( I_l \). The downward pass stops whenever the number of elements in a given box is less than a critical number \( n_F \) defined in section 4.

6. Calculate the velocity and vortex stretching vectors: For every target element \( i \) in the computational domain, identify leaf box \( I_{l^*} \) that contains this element.

- Calculate the far-field velocity and vortex stretching vectors at the target position using \( I_{l^*} \) local expansion coefficients.
- Calculate the velocity and vortex stretching vectors at the target position due to all sources contained within \( I_{l^*} \) neighbors and Inherited lists using the following recursive manner, which is identical to the recursive method used in the \( O(N \log N) \) scheme:
  - For each box \( J \in \mathcal{N}_{n_D}(I_l) \) or \( \mathcal{H}_{n_D}(I_l) \), if the number of elements in \( J \) is less than \( n_T/8 \), the velocity and vortex stretching vectors induced at \( P \) by all source elements contained in \( J \) are calculated by direct summation, else if the target element \( i \) is not contained in \( J \) nor in its neighbors (green boxes in Figure 3), the velocity and vortex stretching vectors induced at \( P \) by all elements contained in \( J \) are approximated using multipole expansion coefficients, else if \( J \) is leaf (hatched boxes in Figure 3), these vectors are calculated by direct summation, else, we go to the next level and repeat the same procedure for \( J \) children (such as blue boxes in Figure 3).

![Figure 3: Schematic for the \( O(N \log N) \) FM scheme.](image)
4. Introducing $n_F$ and $n_T$ through cost analysis

The speed of the adaptive scheme is function of $p$, $n_D$, and $l_m$, where $l_m$ is the maximum level reached upon tree division. As $p$ or $n_D$ increases, the speed of the adaptive scheme will always decrease as we are targeting a more accurate approximations. However, for a given $p$ and $n_D$, as $l_m$ increases the speed of the adaptive scheme increases until it reaches a critical level above which it would be faster and more accurate to perform direct summation rather than proceeding to higher level.

Let $a$ be the number of operation needed to calculate the direct interaction at a target $i$ due to a point source $j$, $b$ the number of operation needed to calculate the interaction due to a set of sources located inside a box $J$ at a given target $i$ using multipole expansion coefficients, $c$ the number of operation needed to get local expansion coefficients from multipole expansion coefficients, and $d$ the number of operation needed to translate local expansion coefficients of a certain box to one of its children.

For a given $p$ and $n_D$, let us consider a target box $I_l$ at level $l$ that contains $n$ Elements. Next we present a criteria based on the number of elements contained in a target box $I_l$ that will help us to judge whether it is faster to go further one level down. We denote $n_T$ the critical number of element in box $I_l$ below which dividing this box into eight children and evaluating their influence using multipole expansion coefficients rather than direct summation will lower the speed of the adaptive fast solver, and $n_F$ the critical number of element in box $I_l$ below which dividing this box into eight children and evaluating their influence using local expansion coefficients rather than direct summation will lower the speed of the adaptive fast solver.
4.1. Evaluating \( n_T \) through cost analysis

For a given target element \( i \) located inside a source box \( I_l \) at level \( l \), the number of operation needed to calculate the direct interaction between all elements within neighbor boxes of box \( I_l \) and element \( i \) is equal to \( 8(n_D + 0.5)^3 na \) since we have \( 8(n_D + 0.5)^3 \) neighbor boxes containing \( n \) Elements each.

Now let us consider the same target element \( i \) contained within a box \( I_{l+1} \) at level \( l + 1 \) where box \( I_{l+1} \) is a child of box \( I_l \). The number of operation needed to calculate the direct interaction between all elements within neighbor boxes of box \( I_{l+1} \) and element \( i \) is equal to \( (n_D + 0.5)^3 na \) since we have \( 8(n_D + 0.5)^3 \) neighbor boxes containing \( \frac{n}{8} \) elements each. The number of operation needed to calculate the interaction between all elements within the interaction list of box \( I_{l+1} \) and element \( i \) is equal to \( 7 \times 8(n_D + 0.5)^3 b \) since we have \( 7 \times 8(n_D + 0.5)^3 \) boxes which are children to box \( I_l \) and are not neighbor to box \( I_{l+1} \).

The critical number \( n_T \) can be found when we have the same amount of operations in both cases, that is:

\[
8(n_D + 0.5)^3 n_T a = (n_D + 0.5)^3 n_T a + 7(n_D + 0.5)^3 \times 8 \times b
\]

\[
\begin{equation}
\begin{split}
n_T &= \frac{8b}{a} \\
&= \frac{8b}{a} \quad (18)
\end{split}
\end{equation}
\]

We conclude that if a target box \( I_l \) at level \( l \) contains \( n \) elements, it would be beneficial to go to level \( l + 1 \) and calculate the velocity and vortex stretching vectors induced by all sources in \( I_l \) neighbors using multipole expansions if and only if \( n > n_T \). This critical number of elements is independent of \( n_D \), and since \( b \) is of the order of \( p^2 \), we can conclude that

\[
n_T = xp^2 + yp + z \quad (19)
\]

Where \( x \), \( y \), and \( z \) are three constants determined by the computer system, language, implementation, core function, etc. These constants also depends on the field that need to be evaluated at every step.

Now, if we consider a source box \( J \) that contains \( n \) elements, the number of operation needed to calculate the direct interaction between \( J \) and any target elements \( i \) outside \( \mathcal{N}_{n_D}(J) \) is equal to \( na \). However, since \( i \) is outside \( \mathcal{N}_{n_D}(J) \), the interaction between \( J \) and \( i \) can be performed using one Multipole expansion. We conclude that if a source box \( J \) contains \( n \) elements, it would be beneficial to calculate the velocity and vortex stretching vectors induced by all sources in \( J \) at any target element outside \( \mathcal{N}_{n_D}(J) \) using multipole expansions rather than direct summation if and only if \( n > \frac{b}{a} = \frac{n_T}{8} \).
4.2. Evaluating $n_F$ through cost analysis

For a given target element $i$ located inside a box $I_l$, the number of operation needed to calculate the direct interaction between all elements within neighbor boxes of box $I_l$ and element $j$ is equal to $8(n_D+0.5)^3 na$ since we have $8(n_D+0.5)^3$ neighbor boxes containing $n$ elements each. Remembering that we have $n$ Elements in box $I_l$, the number of operation needed to calculate the direct interaction between all source elements contained in neighbor boxes of $I_l$ and all target elements contained within box $J_l$ is equal to $8(n_D+0.5)^3 n^2 a$

Now let us consider the same target element $i$ contained within a box $I_{l+1}$ at level $l+1$ where box $I_{l+1}$ is box a child of $I_l$. The number of operation needed to calculate the direct interaction between all the elements contained in neighbor boxes of box $I_{l+1}$ and the element $j$ is equal to $(n_D+0.5)^3 na$ since we have $8(n_D+0.5)^3$ neighbor boxes containing $n/8$ elements each. Remembering that we have $n$ Elements in box $I_l$, the number of operation needed to calculate the direct interaction between all source elements contained in neighbor boxes of $I_{l+1}$ and all target elements within box $I_{l+1}$ is equal to $(n_D+0.5)^3 n^2 a$

Furthermore, if we decided to go one level down, we will have additional cost as a consequence of the conversion and translation operations. The number of operation needed to perform the conversing operations is equal to $8 \times 7 \times 8(n_D+0.5)^3 c$ since we have $8$ children boxes and for each box we have $7 \times 8(n_D+0.5)^3$ boxes which are children of box $I_{l+1}$ parent and are not neighbor to box $I_{l+1}$. In addition, the number of operation needed to perform all the translation operations is equal to $8d$

The critical number $n_F$ can be found when we have the same amount of operations in both cases, that is:

$$8(n_D+0.5)^3 n_F^2 a = (n_D+0.5)^3 n_F^2 a + 8 \times 7 \times 8(n_D+0.5)^3 c + 8d$$

$$n_F^2 = 64 \frac{c}{a} + \frac{8}{7(n_D+0.5)^3 a} d$$

$$n_F^2 = 64 \left( \frac{c}{a} + \frac{1}{n_f} \frac{d}{a} \right)$$

Where $n_f$ is the maximum number of box in the interaction list. We conclude that if a box at level $l$ contains $n$ elements, it would be beneficial to go to level $l+1$ and calculate the velocity and vortex stretching vectors using cell-cell interaction if and only if $n > n_F$.

This critical number is independent of $n_D$, and since $c$ and $d$ are of the order of $p^4$, we can conclude that:

$$n_F^2 = xp^4 + yp^3 + zp^2 + wp + w$$
Where \( x, y, z, u, w \) are five constants determined by the computer system, language, implementation, etc. These constants also depend on the field that need to be evaluated at every step.

Now, if we consider a source box \( J \) that contains \( N_S \) elements, the number of operations needed to calculate the direct interaction between \( J \) and any target box \( I \) outside \( \mathcal{N}_{n_D}(J) \) containing \( N_T \) elements is equal to \( N_S \times N_T \times a \). However, since \( I \) is outside \( \mathcal{N}_{n_D}(J) \), the interaction between \( J \) and \( I \) can be performed using cell-to-cell interaction with \( c \) operations, and can also be performed using cell to particles with \( N_T \times b \) operations. We conclude that if a source box \( J \) contains \( N_S \) elements and a target box contains \( N_T \) elements, it would be beneficial to calculate the velocity and vortex stretching vectors induced by all sources in \( J \) at any target element inside \( J \) with \( I \) outside \( \mathcal{N}_{n_D}(J) \) using cell-cell interaction rather than direct summation or cell to particle interaction if and only if \( N_S \times N_T > \frac{c}{a} \simeq \frac{n_2^2}{64} \) and \( N_T > \frac{c}{b} \simeq \frac{n_2^2}{8 n_T} \).

5. Error analysis of the vector potential for the adaptive scheme with regularized vortices

In the case of regularized vortex method where the vorticity field is approximated using equation (9), the use of equation 12 and 14 to approximate the far-field vector potential will induce an error that has three major components, namely:

1. Error resulting from truncating Multipole Expansions at some order \( p \), \( \vec{E}_M, \vec{\psi} \): This error is related to the series expansion used to represent the kernel \( \frac{1}{r} \) at large distance. The Multipole expansion error is related to the order of expansion \( p \) and the Multipole Acceptance Criteria. \( \| \vec{E}_M, \vec{\psi} \| \) decreases exponentially as \( p \) increases and will eventually converge to zero as \( p \to \infty \). However, increasing \( p \) will have a high negative impact on the adaptive FM speed.

2. Error resulting from truncating Local Expansions at some order \( p \), \( \vec{E}_L, \vec{\psi} \): This error is also related to the series expansion used to represent the kernel \( \frac{1}{r} \) at large distance and decreases exponentially as \( p \) increases.

3. Error resulting from approximating the Biot-Savart kernel by a \( \frac{1}{r} \) kernel, \( \vec{E}_\sigma, \vec{\psi} \): This error arose due to the fact that \( G_\sigma \) deviates from \( \frac{1}{4\pi r} \). This error mainly depends on the size of the smallest boxes (at the deepest level of the tree). In order to keep \( \| \vec{E}_\sigma, \vec{\psi} \| \) small, we must keep the leaf boxes width above a minimum multiple of the core function smoothing radius. In three dimensional flows, this will pose a considerable computational overload on the FM schemes, since it will dramatically increase the particle to particle interaction.

Now let us consider a target element \( i \) within a leaf box \( I_i^* \). The adaptive scheme can perform cell-cell, cell-particle, and particle-particle interactions. \( \mathcal{M} \) is the set of all boxes.
At different levels that interact with element $i$ via cell-particle interaction, that is, the vector potential induced at $i$ by all vorticies contained in $J$ is calculated using equation 12. $\mathcal{L}$ is the set of all boxes at different levels that interact with cell $I_l$ via cell-cell interaction, that is, the vector potential induced at $i$ by all vorticies contained in $J$ is calculated using equation 14. The total error can be expressed as:

$$\|\mathbf{E}_{t,\psi}\| = \|\mathbf{E}_{\sigma,\psi} + \mathbf{E}_{M,\psi} + \mathbf{E}_{L,\psi}\|$$

(24)

with

$$E_{\sigma,\psi} \leq \Gamma_{\mathcal{ML}} \beta_\sigma (r^*)$$

(25)

$$\|\mathbf{E}_{M,\psi}\| \leq \frac{1}{4\pi} \frac{\Gamma_M}{W_0} \frac{2^l m}{n_D} \left( \frac{\sqrt{3}}{2n_D + 1} \right)^{p+1}$$

(26)

$$\|\mathbf{E}_{L,\psi}\| \leq \frac{1}{4\pi} \frac{\Gamma_L}{W_0} \frac{2^l m}{n_D + 1} \left( \frac{1}{2 \sqrt{3} (n_D + 1) - 1} \right)^{p+1}$$

(27)

where $r^* = (n_D + 0.5) \frac{W_0}{2^l m}$, $\beta(r) = |G(r) - \frac{1}{4\pi r}|$, $\beta_\sigma(r) = \frac{1}{2} \beta \left( \frac{r}{r_0} \right)$, $W_0$ is the width of the root box, and $\mathcal{ML} = \mathcal{M} \cup \mathcal{L}$. Furthermore, for any set of boxes $S$, we have:

$$\Gamma_S = \sum_{j \in S} \|\mathbf{\overline{a}}_j\|$$

(28)

Let us consider a specific core function with core radius equal to $\sigma$. For a given tree depth $l_m$, as $p$ increases $\|\mathbf{E}_{\sigma,\psi}\|$ remains constant while $\|\mathbf{E}_{M,\psi}\|$ and $\|\mathbf{E}_{L,\psi}\|$ decrease continuously. Thus the total error converge to $\|\mathbf{E}_{\sigma,\psi}\|$ as $p \to \infty$. So, for each tree depth $l_m$, there is a order of expansion $p_m$ above which increasing $p$ won’t considerably improve the accuracy since the core function (regularization) error would be dominant.

6. Error analysis of the velocity vector for adaptive FM scheme with regularized vortices

The velocity vector is also expressed as a truncation of spherical harmonics. However it has different expressions than those obtained for vector potential, and thus it has different error bounds. The error in the velocity vector can also be decomposed into three major components, namely, the Multipole truncation error $E_{M,\mathbf{u}}$, the Local truncation error $E_{L,\mathbf{u}}$ and the regularizing error $E_{\sigma,\mathbf{u}}$. The total error can be expressed as:
\[ \| \vec{E}_{t, \vec{u}} \| = \| \vec{E}_{\sigma, \vec{u}} + \vec{E}_{M, \vec{u}} + \vec{E}_{L, \vec{u}} \| \] (29)

with

\[ \| \vec{E}_{\sigma, \vec{u}} \| \leq \Gamma_{ML} \kappa_{\sigma}(r^*) \] (30)

\[ \| \vec{E}_{M, \vec{u}} \| \leq \frac{1}{4\pi W_0^2} \frac{4^{l_m}}{(n_D - \sqrt{3} - 1)^2} \left( \frac{2}{n_D + 1} \right)^{p+1} \left[ p + 2 - \frac{\sqrt{3}}{2n_D + 1} (p + 1) \right] \] (31)

\[ \| \vec{E}_{L, \vec{u}} \| \leq \frac{1}{4\pi W_0^2} \frac{4^{l_m}}{(n_D + 1 - \sqrt{3})^2} \left( \frac{2}{n_D + 1} \right)^{1/p+1} \left[ p + 2 - \frac{1}{2n_D + 1} (p + 1) \right] \] (32)

where \( \kappa(r) = \frac{1}{4\pi^2} |1 - 4\pi q(r)| \) and \( \kappa_{\sigma}(r) = \frac{1}{\sigma^2} \kappa \left( \frac{r}{\sigma} \right) \).

\( E_{\sigma, \vec{u}} \) is function of the choice of the core function and its radius, and mainly depends on the maximum level \( l_m \) reached upon tree division. For a given level of accuracy, there exist a critical level \( l_c \) above which the regularization error would be higher than the maximum allowable error. Thus the tree division must stop whenever the tree level reaches \( l_c \). On the other hand, for a constant \( n_D \), the truncation errors, \( E_{M, \vec{u}} \) and \( E_{L, \vec{u}} \), mainly depend on the order of expansion \( p \). For a given level of accuracy, there exist a critical level \( p_c \) below which the truncation error would be higher than the maximum allowable error. For a given order of expansion \( p_c \), the adaptive solver evaluates two critical numbers \( n_T \) and \( n_F \) and also uses them to limit the depth of the tree in order to balance near-field and far-field evaluations and optimize the speed of the fast solver. In fact, the division process will stop whenever the number of elements in a given box is less than \( n_C = \min\{n_T, n_F\} \).

Let \( l^* \) be the tree level reached when we stop the division when number of elements in a given box is less than \( n_C = \min\{n_T, n_F\} \) regardless of \( l_c \). This would be the case when we use a Dirac delta function. If \( l_c > l^* \) the fast solver will manage to balance the loads. However if \( l_c < l^* \) the tree structure would be too shallow and the near field evaluations dominates. In this case, the adaptive solver behaves exactly the same as Greengard FMM since the division process is halted at a given level.
7. Results and Discussion

7.1. Computational cost

In the context of vortex methods, the adaptive fast multipole scheme developed throughout this paper uses both multipole and local spherical harmonic based expansions to approximate the far-field velocity and vortex stretching vectors. This scheme extends Greenegard [6] fast multipole algorithm by introducing a new mechanism to optimize the speed of the fast multipole method. The introduced mechanism make use of two critical number $n_T$ and $n_F$ developed in section 4 to balance the far-field and near field evaluations for different architectures. Both $n_T$ and $n_F$ depends on the order of expansion $p$ and the scalar and vector fields (kernel) evaluated by the adaptive scheme. For a given architecture and given order of expansion $p$, execution time of four different kernels is measured. This information is used to find the optimum values of $n_T$ and $n_F$ through equations 18 and 22. These values are used in the context of $O(N)$ fast multipole to limit the depth of the tree structure and to balance the computational cost of particle to particle, cell to particle, and cell to cell interaction as described in section 3.

Using singular vorticies, we have performed several simulation for different order of expansion $p$ ranging from 2 to 20. If the objective were to optimize the cost of the velocity and vortex stretching vectors calculation, we found out that $n_T = 4.802p^2 + 16.74p + 63.52$ and $n_F = 4.167p^2 + 6.689p + 13.91$ would maximize the speed of the adaptive fast multipole scheme. If the objective were to optimize the cost of the velocity calculation, we found out that $n_T = 3.112p^2 + 13.35p + 19.11$ and $n_F = 5.897p^2 + 9.468p + 19.69$ would then maximize the speed of the adaptive fast multipole scheme.

The $p$-dependence of the critical numbers $n_T$ and $n_F$ is shown in figures 6. It is clear that, for all cases, as $p$ increases both $n_T$ and $n_F$ increase. This is due to the fact that it is more expensive to calculate cell to particle and cell to cell interaction for larger $p$. Figure 6 shows the critical numbers used when only the velocity vector is calculated by the fast scheme. For this case, $n_F$ takes larger values than $n_T$. Thus further cost reduction can be achieved by reaching higher level and by using cell to particle interaction at these levels to approximate the far-field velocity. At these levels, local expansion coefficients are not calculated since cell to cell interaction would increase the overall computational cost. However when both velocity and vortex stretching vectors are evaluated by the fast scheme, $n_F$ takes on smaller values than $n_T$ for any order of expansion $p$. Thus local expansion coefficients are calculated for all cells at leaf level and the use of cell to particle interaction won’t present much advantages. In this case, the adaptive scheme will basically calibrate the balance between particle to particle and cell to cell interaction to obtain the optimum speed.
The fact that \( n_F \) becomes smaller than \( n_T \) when both velocity and stretching vectors are calculated can be logically justified. In fact, cell to cell computational effort is mainly due to conversion and translation operations which are used to find local expansions coefficients at the center of each leaf cell. Once obtained, these coefficients are used to approximate the velocity, vortex stretching, and any higher order derivative of the velocity vector field. So basically, this is a fixed cost that must be paid whether we want to calculate the velocity vector alone or both velocity and vortex stretching vectors and it is by far the most time consuming part in the cell- cell evaluations. Thus calculating the vortex stretching vector, in addition to velocity vector, would not impact cell to cell computational load since local expansion coefficients must be calculated either way. However, calculating the vortex stretching, in addition to the velocity, would dramatically increase the particle to particle load and would also increase cell to particle computational effort. Thus, to achieve the optimum speed, \( n_F \) must decrease in order to balance particle to particle and cell to cell interactions. On the other hand, the decrease of \( n_T \) is rather limited because both particle to particle and cell to particle loads are increasing.
The values of $n_T$ and $n_F$ also depend on the regularization function used to evaluate the near filed velocity and vortex stretching via particle to particle interaction. In fact, compared to the optimum values obtained for the singular Dirac Delta function case, $n_T$ must be divided by $R$ and $n_F$ must be divided by $\sqrt{R}$, where $R = \frac{a_\sigma}{a_\delta}$. $a_\delta$ and $a_\sigma$ are the number of operations needed to calculate the direct interaction for the case of singular and regularized vortices respectively. Thus, the regularization function has higher effect on $n_T$ than $n_F$.

To investigate the performance of the adaptive scheme, we carried out several simulations where both velocity and vortex stretching vectors are calculated simultaneously. A vortex ring with a unit radius and a core radius equal to 0.1 is considered. The initial vorticity within the core of the ring, taken as a second-order Gaussian distribution, is discretized using $N$ singular vortex elements where $N$ ranges between $10^4$ and $20^6$. Throughout all the simulations, $n_D$ is set to 1. This is equivalent to a typical fast multipole with a $3 \times 3 \times 3$ neighbor list. The adaptive scheme performs particle to particle, cell-to particle, and cell
to cell interaction. A mechanism to choose among these interactions is based upon the critical numbers defined in section 4.

The timings on a single CPU core for the hybrid method are shown in Figure 7. The order of expansions is $p = 5$ and the simulations are conducted for a wide range of number of elements and for the following values of critical number $n_F = 40, 129, 250, 500$, where $n_F = 129$ is the optimum value chosen by the scheme that should optimize the speed of the adaptive scheme for $p = 5$ by balancing near-field and far-field evaluations. As it can be seen from figure 7 and for all values of the number of elements, the optimum value chosen by the scheme $n_F = 129$, gives the best performance. In fact for $n_F = 40, 250, 500$, the adaptive scheme suffers from load imbalance between the near-field and far-field evaluations. Particle to particle interaction load is dominant for $n_D = 250$ and 500, while cell to cell interaction are dominant for the case of $n_D = 40$ since we have to calculate local expansions for cells at a deep tree structure.

![Figure 7: Measured simulation time versus the number of elements for different $n_F$ using an order of expansion $p = 5$](image)

Figure 7: Measured simulation time versus the number of elements for different $n_F$ using an order of expansion $p = 5$
Figure 7 also shows that as we deviate from the optimum value of $n_F$, simulation time begin to oscillate above the time measured for the optimum case of $n_F = 129$. These oscillations takes higher peak value and smaller frequency as we move away from the optimal value. Load imbalance caused by improper value of $n_F$ might double the simulation time.

Same simulations were conducted for an order of expansion $p = 8$. In this case the optimum critical value is $n_F = 285$. The timing measured in this case is compared with that obtained for $n_F = 100$, and 500. Figure 8 shows that optimum value of $n_F$ gives the best performance and the results are similar to those obtained for $p = 5$.

![Figure 8: Measured simulation time versus the number of elements for different $n_F$ using an order of expansion $p = 8$](image)

We conclude that the optimum case has major advantages. By limiting the depth of the tree structure and balancing near-field and far-field evaluation, the adaptive scheme optimizes the speed of the fast multipole method for a given order of expansion. This scheme is flexible and has auto-tuning capabilities on heterogeneous architectures, and
can run on any machine without changing anything. Moreover, the current method will automatically choose the critical numbers and alleviate the user from this burden. In addition, the adaptive scheme scale as $O(N^{1.045})$ and eliminates the number of elements dependence of the method.

7.2. Error analysis

The proper evaluation of $n_T$ and $n_F$ would optimize the speed of the adaptive solver for a given order of expansion $p$ while using a Dirac Delta function. In this case, the accuracy of the fast summation is completely determined by the order of expansion. However, the use of a regularization core function instead of dirac function will induce an additional error, which depends on the used core function and the neighborhood dimension. For a given level of accuracy, there always exist a critical level $l_c$ that we should not cross over.

To investigate the accuracy of the adaptive scheme, we have conducted several simulation for different order of expansion $p$ and using different core function and we have reported the $L_2$ norm of the truncation and regularization error. A vortex ring with similar geometric parameter to that described earlier is descritized using $N = 1108880$ elements. Figures 9, 10, 11 show the $L_2$ error norm versus $p$ for a second order algebraic core function, second order Gaussian core function, and fourth order algebraic core function respectively obtained for the adaptive solver. As it can be seen in all cases, while the truncation error always decreases exponentially, the regularization error and thus the total error decreases in steps from one stable value to another. These error values correspond to regularization error calculated when we force the tree division to stop at a certain level. In fact, the regularization error depends mainly on the core function and on the distance to neighbor cells which is related to the tree depth $l_m$. For a given core function, increasing $p$ will lead to an increase in $n_T$ and $n_F$ which might lead to decrease the tree depth. This mainly explain the step behaviour of the regularization and total error.

Figure 12, 13, 14 show the regularization error obtained when we have force the division of the tree to stop at different levels for the three core functions for a fixed order od expansion $p = 5$. It is clear that as the maximum level $l_m$ increases, the regularization error increases while the truncation error remains constant. The increase in the regularization error is because increasing the tree level $l_m$ will decrease the distance between neighboring cell and thus increase $\kappa$. From all the above, we conclude that, for a given level of accuracy, there exist a critical level $l_c$ above which the regularization error would be higher than the allowable error. For this critical level, there exist a critical order of expansion $p_c$ above which increasing $p$ will only increase the computational cost without improving the total error. The critical level $l_c$ and its associated critical order of expansion $p_c$ are first evaluated by evaluating the errors on a representative sample of vortices. Then, the critical number of elements $n_T$ and $n_F$ are evaluated as a function of $p_c$. The division process stops whenever the box level $l$ is
equal to the critical level $l_c$ or the number of elements in the box is less $n_C = \min\{n_T, n_F\}$, where $n_T$ and $n_F$ are defined in section 4.

8. Conclusions

We presented an error-controlled adaptive fast solver for approximating the velocity and vortex stretching vectors in the context of grid free three dimensional vortex methods. We introduced three critical numbers, $n_T$, $n_F$, and $l_c$. These critical numbers are used to limit the tree depth and to balance near and far field evaluations in order to obtain the highest speed for any core function and any level of accuracy. The adaptive scheme scale as $O(N^{1.045})$ and eliminates the number of elements dependence of the method. This scheme is flexible and has auto-tuning capabilities on heterogeneous architectures, and can run on any machine without changing anything. Moreover, the current method will automatically choose the critical numbers and alleviate the user from this burden.
Figure 10: $L_2$ norm for the truncation and regularization error versus $p$ using a second order Gaussian core function in the context of adaptive scheme.

9. Acknowledgments

This work is supported by the University Research Board (URB) of the American University of Beirut.

10. Bibliography

References

[1] Barnes, J. E., Hut, P., 1986. A hierarchical $o(n \log n)$ force calculations algorithm. Nature 324, 446.

[2] Capuzzo-Dolcetta, R., Miocchi, P., 1998. A comparison between the fast multipole algorithm and the tree-code to evaluate gravitational forces in 3-d. Journal of Computational Physics 143 (1), 29–48.
Figure 11: $L_2$ norm for the truncation and regularization error versus $p$ using a fourth order algebraic core function in the context of adaptive scheme.

[3] Cheng, H., Greengard, L., Rokhlin, V., 1999. A fast adaptive multipole algorithm in three dimensions. Journal of Computational Physics 155 (2), 468 – 498. URL [http://www.sciencedirect.com/science/article/pii/S0021999199963556](http://www.sciencedirect.com/science/article/pii/S0021999199963556)

[4] Dehnen, W., 07 2000. A very fast and momentum-conserving tree code. The Astrophysical journal 536, L39–L42.

[5] Dehnen, W., 2002. A hierarchical $O(n)$ force calculation algorithm. arXiv preprint astro-ph/0202512.

[6] Greengard, L., Rokhlin, V., 1987. A fast algorithm for particle simulations,. J. Comput. Phys. 73, 325.

[7] Warren, M. S., Salmon, J. K., 1995. A portable parallel particle program. Computer
Figure 12: \(L_2\) norm for the truncation and regularization errors versus \(l_m\) for a fixed order of expansion \(p = 5\) using a second order algebraic core function in the adaptive scheme.

[8] Yokota, R., 2012. An FMM based on dual tree traversal for many-core architectures. CoRR abs/1209.3516. URL http://arxiv.org/abs/1209.3516

[9] Yokota, R., Barba, L., 2011. Parameter tuning of a hybrid treecode-fmm on gpus.

[10] Yokota, R., Barba, L., May 2012. Hierarchical n-body simulations with autotuning for heterogeneous systems. Computing in Science Engineering 14 (3), 30–39.

[11] Yokota, R., Barba, L. A., 2011. Fast n-body simulations on gpus. CoRR, abs/1108.5815.

26
11. Appendix A: Error bounds for the potential and velocity vectors

11.1. Error bounds for the potential vector induced by a singular source element

Consider a singular source element $j$ of strength $\tilde{\alpha}_j$ located at $Q = (\rho_j, \theta_j, \varphi_j)$. The potential vector induced by $j$ at a target element located at $P = (r, \theta, \phi)$ has the following expression:

$$\vec{\psi}_j(P) = \frac{\tilde{\alpha}_j}{4\pi d_j},$$

(33)

where $d_j = \|QP\|$ is the distance between source and target elements. Let $\gamma$ be the angle between the vectors $P$ and $Q$. From the law of cosines, we have:

$$d_j^2 = r^2 + \rho_j^2 - 2r\rho_j \cos \gamma$$

(34)
From this relation, we may write

\[ \vec{\psi}_j(P) = \frac{\tilde{\alpha}_j}{4\pi r} \frac{1}{\sqrt{1 - 2 \left( \frac{\rho_j}{r} \right) \cos \gamma + \left( \frac{\rho_j}{r} \right)^2}} \] (35)

Let \( \mu_j = \frac{\rho_j}{r} \). For \( \rho_j < r \), that is \( \mu_j < 1 \), if we expand \( \vec{\psi}(P) \) in term of Legendre Polynomial we obtain:

\[ \vec{\psi}_j(P) = \frac{\tilde{\alpha}_j}{4\pi r} \sum_{n=0}^\infty \mu_j^n P_n(\cos \gamma) \] (36)

The error resulting from truncating \( \vec{\psi}_j(P) \) at some order \( p \) is:

\[ E_{\vec{\psi}_j} = \left\| \frac{\tilde{\alpha}_j}{4\pi d_j} - \frac{\tilde{\alpha}_j}{4\pi r} \sum_{n=0}^p \mu_j^n P_n(\cos \gamma) \right\| = \left\| \frac{\tilde{\alpha}_j}{4\pi r} \sum_{n=p+1}^\infty \mu_j^n P_n(\cos \gamma) \right\| \] (37)
Knowing that $P_n(cos\gamma) \leq 1$, and using the triangle inequality, we obtain an error bound for the multipole expansion

$$E_{\psi_j} \leq \left\| \frac{\alpha_j}{4\pi d_j} \sum_{n=p+1}^{\infty} \mu_j^n \right\| \leq \frac{\| \alpha_j \|}{4\pi r} \frac{\mu_{p+1}^j}{1 - \mu_j}$$

(38)

$$E_{\psi_j} \leq \frac{1}{4\pi} \left\| \frac{\alpha_j}{\rho_j} \right\| \left( \frac{\rho_j}{r} \right)^{p+1}$$

(39)

Similarly, when $r < \rho_j$, we define $\mu_j = \frac{r}{\rho_j}$. The potential vector and the error bound for the local expansion would have the following expressions:

$$\tilde{\psi}_j(P) = \frac{\tilde{\alpha}_j}{4\pi d_j} = \frac{\tilde{\alpha}_j}{4\pi \rho_j} \sum_{n=0}^{\infty} \mu_j^n P_n(cos\gamma)$$

(40)

$$E_{\tilde{\psi}_j} \leq \frac{1}{4\pi} \left\| \frac{\tilde{\alpha}_j}{\rho_j} \right\| \left( \frac{\rho_j}{r} \right)^{p+1}$$

(41)

11.2. Error bounds for the potential vector induced by a cluster of source elements

Suppose that $s$ vortices with strength $(\tilde{\alpha}_j, j = 1...s)$ are located at the points $\bar{Q}_j = (\rho_j, \theta_j, \varphi_j)$ inside the sphere $D_Q$ of radius $a$ with center at $Q = (0,0,0)$, then at any $\bar{P} = (r, \theta, \varphi)$ with $r > a$, the potential vector field is given by:

$$\tilde{\psi} = \sum_j \tilde{\psi}_j = \sum_j \frac{\tilde{\alpha}_j}{4\pi d_j} = \sum_j \frac{\tilde{\alpha}_j}{4\pi r} \sum_{n=0}^{\infty} \mu_j^n P_n(cos\gamma)$$

(42)

The error resulting from truncating $\tilde{\psi}(P)$ at some order $p$ is:

$$E_{\tilde{\psi}} = \left\| \sum_j \left( \frac{\tilde{\alpha}_j}{4\pi d_j} - \frac{\tilde{\alpha}_j}{4\pi r} \sum_{n=0}^{p} \mu_j^n P_n(cos\gamma) \right) \right\|$$

(43)

Using the triangle inequality, and since $\mu_j \leq \frac{a}{r}$, the multipole expansion error will be bounded by:

$$E_{\tilde{\psi}} \leq \frac{1}{4\pi} \left( \frac{\Gamma_{D_Q}}{r-a} \right) \frac{a}{r}^{p+1}$$

(44)

where $\Gamma_{D_Q} = \sum_j \| \tilde{\alpha}_j \|$

Now suppose that $s$ vortices with strength $(\tilde{\alpha}_j, j = 1...s)$ are located inside a cubic box $J_l$ at level $l$ with center at $Q = (0,0,0)$. The radius of the smallest sphere which encloses box $J_l$ is $a_{J_l} = \frac{\sqrt{3}}{2} W_{J_l}$ where $W_{J_l}$ is the box width. Following the adaptive solver Scheme described
in section ??, the distance between the center of box $J_l$ and any point $P = (r, \theta, \varphi)$ located outside the neighborhood of box $J_l$ is $r_{J_l} \geq (n_D + 0.5)W_{J_l}$. For any $p \geq 1$, the multipole expansion error will be bounded by:

$$E_{M,\psi} \leq \frac{1}{4\pi} \frac{\Gamma_{J_l}}{W_{J_l}} \left( \frac{\sqrt{3}}{2n_D + 1} \right)^{p+1}$$

(45)

Noting that $W_{J_l} = W_0/2^l$, we get

$$E_{M,\psi} \leq \frac{1}{4\pi} \frac{\Gamma_{J_l}}{W_0} \left( \frac{2^l}{n_D - \frac{\sqrt{3}}{2}} \right)^{p+1}$$

(46)

where $W_0$ is the width of the root box, $\Gamma_{J_l} = \sum_j ||\vec{\alpha}_j||$ and the summation is done over all the source elements within box $J_l$.

Similarly, suppose that $s$ vortices with strength $(\vec{\alpha}_i, i = 1...s)$ are located inside the sphere $D_Q$ of radius $a$ with center at $Q = (\rho, \alpha, \beta)$, and that $\rho = (c + 1)a$ with $c > 1$, then for any target $P = (r, \theta, \varphi)$ inside the sphere $D_0$ of radius $a$ centred at the Origin, the potential vector is given by

$$\vec{\psi} = \sum_j \vec{\psi}_j = \sum_j \frac{\vec{\alpha}_j}{4\pi d_j} = \sum_j \frac{\vec{\alpha}_j}{4\pi \rho_j} \sum_{n=0}^{\infty} \mu_j^n P_n(\cos\gamma)$$

(47)

The error resulting from truncating $\vec{\psi}(P)$ at some order $p$ is:

$$E_{\vec{\psi}} = \left| \sum_j \left( \frac{\vec{\alpha}_j}{4\pi d_j} - \frac{\vec{\alpha}_j}{4\pi \rho_j} \sum_{n=0}^{p} \mu_j^n P_n(\cos\gamma) \right) \right|$$

(48)

Since $\mu_j = \frac{\rho}{\rho_j} \leq \frac{a}{p-a} = \frac{1}{c}$, using the triangle inequality, the local expansion error will be bounded by:

$$E_{\vec{\psi}} \leq \frac{1}{4\pi} \frac{\Gamma_{D_Q}}{(ca - a)} \left( \frac{1}{c} \right)^{p+1}$$

(49)

where $\Gamma_{D_Q} = \sum_j ||\vec{\alpha}_j||$.

Now suppose that $s$ vortices with strength $(\vec{\alpha}_j, j = 1...s)$ are located inside a cubic box $J_l$ at level $l$ with center at $Q = (\rho, \alpha, \beta)$. The radius of the smallest sphere which encloses box $J_l$ is $a_{J_l} = \frac{\sqrt{3}}{2}W_{J_l}$ where $W_{J_l}$ is the box width. Following the adaptive solver Scheme described in section ??, an upper bound for $c$ is given by

$$c \geq \frac{2\sqrt{3}}{3}(n_D + 1) - 1,$$

(50)
and for any \( p \geq 1 \), the local expansion error that results upon approximating the potential vector at any point \( P \) within a box \( I_l \) centered at the origin is bounded by:

\[
E_{L, \hat{\psi}} \leq \frac{1}{4\pi} \Gamma_j \frac{2l}{W_0 n_D + 1 - \frac{\sqrt{3}}{2}} \left( \frac{1}{\frac{2\sqrt{3}}{3}(n_D + 1) - 1} \right)^{p+1}
\]  

(51)

11.3. Error bounds for the velocity vector induced by a singular source element

Once again, let us consider a singular source element \( j \) of strength \( \vec{\alpha}_j \) located at \( Q = (\rho_j, \theta_j, \varphi_j) \). The velocity at a target element \( P = (r, \theta, \varphi) \) is calculated as the curl of the potential vector \( \vec{\psi}_j \)

\[
\vec{u}_j(P) = \nabla \times \vec{\psi}_j(P) = \nabla \times \left( \frac{\vec{\alpha}_j}{4\pi r} \sum_{n=0}^{\infty} \mu_j^n P_n(\cos\gamma) \right) = \sum_{n=0}^{\infty} \frac{\rho_j^n}{4\pi} \nabla \left( \frac{P_n(\cos\gamma)}{r^{n+1}} \right) \times \vec{\alpha}_j
\]  

(52)

The error resulting from truncating \( \vec{u}_j(P) \) at some order \( p \) is:

\[
E_{\vec{u}_j} = \left\| \sum_{n=p+1}^{\infty} \frac{\rho_j^n}{4\pi} \nabla \left( \frac{P_n(\cos\gamma)}{r^{n+1}} \right) \times \vec{\alpha}_j \right\|
\]  

(53)

Since \( \| \vec{a} \times \vec{b} \| \leq \| \vec{a} \| \| \vec{b} \| \) for any two vectors \( \vec{a} \) and \( \vec{b} \), using the triangle inequality, the truncating error will be bounded by:

\[
E_{\vec{u}_j} \leq \frac{\| \vec{\alpha}_j \|}{4\pi} \sum_{n=p+1}^{\infty} \rho_j^n \left\| \nabla \left( \frac{P_n(\cos\gamma)}{r^{n+1}} \right) \right\|
\]  

(54)

Since the norm of a vector does not vary when we change the reference frame, we will evaluate the expression of \( \left\| \nabla \left( \frac{P_n(\cos\gamma)}{r^{n+1}} \right) \right\| \) in a new frame obtained by rotating the old frame such that the source element \( Q \) will be located on the new \( zz' \) axe, that is we have \( \cos\gamma = \cos\theta \)

\[
\nabla \left( \frac{P_n(\cos\theta)}{r^{n+1}} \right) = -\frac{1}{r^{n+2}} ( (n + 1) P_n(\cos\theta), \sin(\theta)P'_n(\cos\theta), 0)^T
\]  

(55)

\[
\left\| \nabla \left( \frac{P_n(\cos\theta)}{r^{n+1}} \right) \right\| = \frac{1}{r^{n+2}} \sqrt{(n + 1)^2 P^2_n(\cos\theta) + \sin^2(\theta)P'^2_n(\cos\theta)}
\]  

(56)

Knowing that

\[
(n + 1)^2 P^2_n(x) + (1 - x^2)P'^2_n(x) \leq (n + 1)^2
\]  

(57)

we obtain:

\[
\left\| \nabla \left( \frac{P_n(\cos\theta)}{r^{n+1}} \right) \right\| \leq \frac{n + 1}{r^n + 2}
\]  

(58)
Replacing in equation (54), the multipole expansion error of the velocity vector is bounded by:

\[
E \vec{u}_j \leq \left\| \frac{\vec{\alpha}_j}{4\pi r^2} \sum_{n=p+1}^{\infty} (n+1)\mu_n^p \right\| \tag{59}
\]

\[
E \vec{u}_j \leq \left\| \frac{\vec{\alpha}_j}{4\pi r^2} \right\| \left[ \frac{\mu_j^{p+1}}{1-\mu_j} + \mu \frac{d}{d\mu_j} \left( \frac{\mu_j^{p+1}}{1-\mu_j} \right) \right] \tag{60}
\]

\[
E \vec{u}_j \leq \left\| \frac{\vec{\alpha}_j}{4\pi r^2} \right\| \frac{\mu_j^{p+1}}{(1-\mu_j)^2} \left[ p + 2 - \mu_j (p + 1) \right] \tag{61}
\]

\[
E \vec{u}_j \leq \frac{1}{4\pi} \left( \frac{\rho_j}{r} - 1 \right)^{p+1} \left[ p + 2 - \rho_j \left( \frac{p + 1}{r} \right) \right] \tag{62}
\]

Similarly, when \( r < \rho_j \), \( \mu_j = \frac{r}{\rho_j} \), the velocity vector induced by \( j \) at a target \( P = (r, \theta, \phi) \) is calculated as the curl of the potential vector and has the following expression:

\[
\vec{u}_j(P) = \nabla \times \vec{\psi}_j(P) = \nabla \times \left( \frac{\vec{\alpha}_j}{4\pi \rho_j} \sum_{n=0}^{\infty} \mu_n^p P_n(\cos \gamma) \right) = \sum_{n=0}^{\infty} \frac{1}{4\pi \rho_j} \nabla (r^n P_n(\cos \gamma)) \times \vec{\alpha}_j \tag{63}
\]

The error resulting from truncating \( \vec{u}_j(P) \) at some order \( p \) is:

\[
E \vec{u}_j = \left\| \sum_{n=p+1}^{\infty} \frac{1}{4\pi \rho_j} \nabla (r^n P_n(\cos \gamma)) \times \vec{\alpha}_j \right\| \tag{64}
\]

\[
E \vec{u}_j \leq \frac{1}{4\pi} \sum_{n=p+1}^{\infty} \frac{1}{\rho_j} \left\| \nabla (r^n P_n(\cos \gamma)) \right\| \tag{65}
\]

Once again, the norm of a vector does not vary when we change the reference frame. Calculating the norm in the new reference frame described above, we obtain:

\[
\nabla (r^n P_n(\cos \gamma)) = r^{n-1}(n P_n(\cos \theta), -\sin \theta P_n'(\cos \theta), 0)^T \tag{66}
\]

\[
\left\| r^n P_n(\cos \gamma) \right\| = r^{n-1} \sqrt{n^2 P_n^2(\cos \theta) + \sin^2 \theta P_n'(\cos \theta)^2} \tag{67}
\]

Knowing that

\[
n^2 P_n^2(x) + (1 - x^2) P_n'(x)^2 \leq n^2 \tag{68}
\]

we obtain:

\[
\left\| r^n P_n(\cos \gamma) \right\| = nr^{n-1} \tag{69}
\]

32
Replacing in equation (65), the local expansion error of the velocity vector is bounded by:

\[ E_{\vec{u}_j} \leq \frac{\| \vec{\alpha}_j \|}{4\pi \rho_j} \sum_{n=p+1}^{\infty} n \mu_j^n \]  

(70)

\[ E_{\vec{u}_j} \leq \frac{\| \vec{\alpha}_j \| d}{4\pi \rho_j (1-\mu_j)^2} \left( \frac{\mu_j^{p+1}}{1-\mu_j} \right) \]  

(71)

\[ E_{\vec{u}_j} \leq \frac{\| \vec{\alpha}_j \| \mu_j^{p+1}}{4\pi \rho_j^2 (1-\mu_j)^2} [p + 2 - \mu_j p] \]  

(72)

\[ E_{\vec{u}_j} \leq \frac{1}{4\pi (\rho_j - r)^2} \left( \frac{r}{\rho_j} \right)^{p+1} \left[ p + 2 - \frac{r}{\rho_j} p \right] \]  

(73)

11.4. Error bounds for the velocity vector induced by a cluster of source elements

Suppose that \( s \) vortices with strength \( (\vec{\alpha}_j, j = 1...s) \) are located at the points \( \vec{Q}_j = (\rho_j, \theta_j, \phi_j) \) inside the sphere \( D_Q \) of radius \( a \) with center at \( Q = (0,0,0) \), then at any \( P = (r, \theta, \varphi) \) with \( r > a \), the velocity vector field is given by:

\[ \vec{u} = \sum_j \vec{a}_j = \sum_j \sum_{n=0}^{\infty} \frac{\rho_j^n}{4\pi} \nabla \left( \frac{P_n(\cos\gamma)}{r^{n+1}} \right) \times \vec{\alpha}_j \]  

(74)

Using the triangle inequality, and since \( \mu_j \leq \frac{a}{r} \), the multipole expansion error will be bounded by:

\[ E_{\vec{u}} \leq \sum_i E_{\vec{u}_i} \leq \frac{1}{4\pi} \frac{\Gamma_D}{(r-a)^2} \left( \frac{a}{r} \right)^{p+1} \left[ p + 2 - \frac{a}{r} (p+1) \right] \]  

(75)

where \( \Gamma_D = \sum_j \| \vec{\alpha}_j \| \).

Now suppose that \( s \) vortices with strength \( (\vec{\alpha}_j, j = 1...s) \) are located inside a cubic box \( J_l \) at level \( l \) with center at \( Q = (0,0,0) \). For any \( p \geq 1 \), the multipole expansion error that results upon approximating the velocity vector at any point \( P \) outside the neighbours of \( J_l \) is bounded by:

\[ E_{M,\vec{u}} \leq \frac{1}{4\pi} \frac{\Gamma_{J_l}}{W_0^2} \left( \frac{4l^*}{n_D - \sqrt{3}-1} \right)^2 \left( \frac{\sqrt{3}}{2n_D + 1} \right)^{p+1} \left[ p + 2 - \frac{\sqrt{3}}{2n_D + 1} (p+1) \right] \]  

(76)

Similary, suppose that \( s \) vortices having circulation \( (\vec{\alpha}_i, i = 1...s) \) are located inside the sphere \( D_Q \) of radius \( a \) with center at \( Q = (\rho, \alpha, \beta) \), and that \( \rho = (c+1)a \) with \( c > 1 \),
then for any target $P = (r, \theta, \phi)$ inside the sphere $D_0$ of radius $a$ centred at the Origin, the velocity vector is given by

$$\vec{u} = \sum_j \vec{u}_j = \sum_j \sum_{n=0}^{\infty} \frac{1}{4\pi \rho_j^n} \nabla (r^n P_n(\cos \gamma)) \times \vec{a}_j$$

(77)

Since $\mu_j = \frac{r}{\rho_j} \leq \frac{a}{\rho - a} = \frac{1}{c}$, using the triangle inequality, the local expansion error will be bounded by:

$$E_{\vec{u}} \leq \sum_j E_{\vec{u}_j} \leq \frac{1}{4\pi (ca - c)^2} \left( \frac{1}{c} \right)^{p+1} \left[ p + 2 - \frac{1}{c^p} \right]$$

(78)

where $\Gamma_{DQ} = \sum_j \| \vec{a}_j \|$.

Now suppose that $s$ vortices with strength $(\vec{a}_j, j = 1...s)$ are located inside a cubic box $J_l$ at level $l$ with center at $Q = (\rho, \alpha, \beta)$. For any $p \geq 1$, the local expansion error that results upon approximating the velocity vector at any point $P$ within a box $I_l$ centered at the origin is bounded by:

$$E_{L, \vec{u}} \leq \frac{1}{4\pi W_0^2} \frac{4^{r^*}}{(n_D + 1 - \sqrt{3})^2} \left( \frac{1}{2\sqrt{3}(n_D + 1) - 1} \right)^{p+1} \left[ p + 2 - \left( \frac{1}{2\sqrt{3}(n_D + 1) - 1} \right)^p \right]$$

(79)

11.5. Regularization error bounds for the potential vector

In the case of regularized vortex method, an additional error $E_{\sigma, \vec{\psi}}$ arises since $G_\sigma$ deviates from $\frac{1}{4\pi r}$. Let us consider $s$ vortices with strength $(\vec{a}_j, j = 1...s)$ located inside a cubic box $J_l$. Introducing the kernel $\beta_\sigma(r) = \frac{1}{\sigma} \beta \left( \frac{r}{\sigma} \right)$, and $\beta(r) = |G(r) - \frac{1}{4\pi r}|$ we obtain:

$$E_{\sigma, \vec{\psi}} = \left\| \sum_{j \in J_l} \vec{a}_j \left( G_\sigma(r_j) - \frac{1}{4\pi r_j} \right) \right\|
\leq \sum_{j \in J_l} \| \vec{a}_j \| \left| G_\sigma(r_j) - \frac{1}{4\pi r_j} \right|
\leq \left| G_\sigma(r^*) - \frac{1}{4\pi r^*} \right| \sum_{j \in J_l} \| \vec{a}_j \|
\leq \Gamma_{J_l} \beta_\sigma(r^*)$$

(80)

where $\Gamma_{J_l} = \sum_j \| \vec{a}_j \|$, $r_j$ is the distance between source element $j$ and $P$, $r^*$ is the distance that maximizes the kernel $\beta_\sigma(r)$. $r^*$ depends only on the choice of the core function and for most cases we have $r^* = (n_D + 0.5) W_{\text{min}} = (n_D + 0.5) \frac{W_0}{2\pi}$.
11.6. Regularization error bounds for the velocity vector

The regularizing error for the velocity arose since \( q_\sigma(r) \) deviates from \( \frac{1}{4\pi} \). Let us consider \( s \) vortices with strength \((\vec{\alpha}_j, j = 1...s)\) located inside a cubic box \( J_1 \). For any target element \( i \) we have:

\[
E_{\sigma, \vec{u}} = \left\| \sum_{j \in J_1} \frac{1}{4\pi r_j^3}(\vec{x}_i - \vec{x}_j) \wedge \vec{\alpha}_j - \frac{q_\sigma(r_j)}{r_j^3}(\vec{x}_i - \vec{x}_j) \wedge \vec{\alpha}_j \right\|
\]

\[
\leq \sum_{j \in J_1} \left| \frac{1}{4\pi r_j^3}(\vec{x}_i - \vec{x}_j) \wedge \vec{\alpha}_j - \frac{q_\sigma(r_j)}{r_j^3}(\vec{x}_i - \vec{x}_j) \wedge \vec{\alpha}_j \right|
\]

\[
\leq \sum_{j \in J_1} \frac{1}{4\pi r_j^3}|1 - 4\pi q_\sigma(r_j)| \| (\vec{x}_i - \vec{x}_j) \wedge \vec{\alpha}_j \|
\]

\[
\leq \sum_{j \in J_1} \frac{1}{4\pi r_j^2}|1 - 4\pi q_\sigma(r_j)| \| \vec{\alpha}_j \|
\]

\[
\leq \frac{\Gamma_{J_1}}{4\pi r^*} \left[ 1 - 4\pi q \left( \frac{r^*}{\sigma} \right) \right]
\]

\[
\leq \Gamma_{J_1} \kappa_\sigma(r^*)
\]

where \( \Gamma_{J_1} = \sum_{j} \| \vec{\alpha}_j \| \), \( \kappa(r) = \frac{1}{4\pi r^2}|1 - 4\pi q(r)| \) and \( \kappa_\sigma(r) = \frac{1}{\sigma^2} \kappa \left( \frac{r}{\sigma} \right) \), \( r_j \) is the distance between source element and target elements, \( r^* \) is the distance that maximizes the kernel \( \kappa_\sigma(r) \).