A NEW APPROACH FOR FAST POTENTIAL EVALUATION IN N-BODY PROBLEMS

A Thesis

by

SREEKANTH JUTTU

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

December 2003

Major Subject: Computer Science

A NEW APPROACH FOR FAST POTENTIAL EVALUATION IN N-BODY PROBLEMS

A Thesis

by

SREEKANTH JUTTU

Submitted to Texas A&M University
in partial fulfillment of the requirements
for the degree of

MASTER OF SCIENCE

Approved as to style and content by:

Vivek Sarin (Chair of Committee)	Paul Nelson (Member)
Weiping Shi (Member)	Valerie E. Taylor (Head of Department)

December 2003

Major Subject: Computer Science

ABSTRACT

A New Approach for Fast Potential Evaluation in N-Body Problems. (December 2003)

Sreekanth Juttu, M.S., Indian Institute of Technology Kanpur

Chair of Advisory Committee: Dr. Vivek Sarin

Fast algorithms for potential evaluation in N-body problems often tend to be extremely abstract and complex. This thesis presents a simple, hierarchical approach to solving the potential evaluation problem in $O(n)$ time. The approach is developed in the field of electrostatics and can be extended to N-body problems in general. Herein, the potential vector is expressed as a product of the potential matrix and the charge vector. The potential matrix itself is a product of component matrices. The potential function satisfies the Laplace equation and is hence expressed as a linear combination of spherical harmonics, which form the general solutions of the Laplace equation. The orthogonality of the spherical harmonics is exploited to reduce execution time. The duality of the various lists in the algorithm is used to reduce storage and computational complexity. A smart tree-construction strategy leads to efficient parallelism at computation intensive stages of the algorithm. The computational complexity of the algorithm is better than that of the Fast Multipole Algorithm, which is one of the fastest contemporary algorithms to solve the potential evaluation problem. Experimental results show that accuracy of the algorithm is comparable to that of the Fast Multipole Algorithm. However, this approach uses some implementation principles from the Fast Multipole Algorithm. Parallel efficiency and scalability of the algorithms are studied by the experiments on IBM p690 multiprocessors.

To my mother

ACKNOWLEDGMENTS

I'd like to express my sincere gratitude to my advisor Dr. Vivek Sarin for his continued guidance and support all through my stay at Texas A&M University. He has been a very good teacher and I have learnt a lot from him over the past two years.

I am extremely grateful to Dr. Paul Nelson and Dr. Weiping Shi for extending their complete support and cooperation throughout the development of my thesis.

I'd like to thank Hemant Mahawar, my office-mate and friend with whom I shared a lot of insightful discussions. My thesis wouldn't be complete on time without his generous assistance during the completion of the document. I am also thankful to Kasturi and Radhika who devoted a lot of time and effort in proofreading the final thesis and helped me a lot in correcting bugs and fine tuning the thesis document. I am indebted to Puneet Singla and Amit Joshi for hosting me during my initial stay at their home. I wouldn't have my first car today if Anees hadn't gotten up before 8:00 am on a Saturday morning and given me a generous ride. Finally, I'd like to thank my friends, Reddy, Bhat, Parag, Neeraj, Parikshit, Pendse, Raghav, and Vivek, who have made my stay in College Station very enjoyable.

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

N-body problems are a class of problems that deal with the mutual interactions among a set of bodies in space. When the name was first coined, the application area of N-body problems was astrophysics. However, N-body problems gradually encompassed a variety of fields such as electrostatics and molecular biology.

Typical problems that are addressed in N-body problems are potential evaluation, force evaluation, and prediction of future movement of bodies. In this thesis, we concentrate on the potential evaluation problem in electrostatics. Potential evaluation of a set of charged particles at the their charge locations is a challenging problem. A straight-forward algorithm 1.2, based solely on particle-particle interactions has $O(n^2)$ time complexity.

This complexity is enormous for large collections of particles that are common in typical problems. There are a number of approximation algorithms, e.g., *Appel's Algorithm* [1], *Barnes-Hut Algorithm* [2] and *Fast Multipole Method* (FMM) [4, 5], that reduce the computational cost of the problem. These algorithms make use of cluster-particle and cluster-cluster interactions to reduce the time complexity to $O(n\log n)$ or $O(n)$. As opposed to the naive $O(n^2)$ algorithm that evaluates the potential exactly, approximation algorithms estimate the potential to desired accuracy that can be controlled by certain parameters. There is a trade off between the accuracy and the computational cost, with a higher accuracy demanding a higher computational cost.

Though FMM has $O(n)$ time complexity, it is extremely complex and hence difficult to implement. In this thesis, a simpler approach is developed to solve the potential evaluation problem. Herein, a new matrix-based hierarchical approach is developed to evaluate the potential in $O(n)$ time. The potential vector (i.e., a vector of potentials of all particles)

The journal model is SIAM Journal on Scientific Computing.

is expressed as a product of the potential matrix and the charge vector. The potential at a point is computed as a sum of potentials due to clusters of particles. An $O(n)$ algorithm is obtained by organizing the clusters in a hierarchical manner. Potential satisfies the Laplace equation and is expressed as a weighted sum of *Spherical Harmonics*, which form the class of general solutions of the Laplace equation. This approach utilizes, and improves upon, some implementation ideas from the Fast Multipole Method, which was developed by Rokhlin and Greengard (see, e.g., [4] for the FMM algorithm and [5] for an improved version). The approach developed in this thesis can be extended to N-body problems in a straight-forward manner.

A. A Simple Algorithm

Suppose $(q_1,x_1),(q_2,x_2),\ldots,(q_n,x_n)$ are n charges distributed in space. The potential at the location x_l due to the charge (q_j,x_j) is given by

$$\phi(x_l) = \frac{q_j}{\|x_l - x_j\|}, \tag{1.1}$$

where $\|x_l - x_j\|$ is the Euclidean distance between x_l and x_j.
A straight forward method to evaluate the total potential at a charge location x_l, due to the rest of the charges is given by

$$\phi(x_l) = \sum_{j=1, j\neq l}^{n} \frac{q_j}{\|x_l - x_j\|}. \tag{1.2}$$

B. Related Work

In order to solve the potential evaluation problem, algorithms based solely on particle-particle interactions would invariably take $O(n^2)$ time. Cluster-particle and cluster-cluster interactions need to be employed in order to improve the time complexity.

Most algorithms, like *Barnes-Hut* and FMM form a tree in order to evaluate the

potential at each particle. A large enough box or node (which is the root) is chosen to enclose all particles. This root node is then recursively sub-divided into eight equal child nodes, until each leaf node has utmost a constant number of particles. The constant is chosen so as to minimize the overall running time of the algorithm.

Barnes and Hut proposed an algorithm to compute gravitational potential due to a set of bodies that makes use of cluster-particle interactions to achieve $O(n \log n)$ time complexity. In order to compute the potential at a location, the set of bodies around it is partitioned into clusters. Starting from the root, each node is ascertained to be far enough or not far enough from the charge location. If a node is far enough, it is considered to be a cluster. If a node is not far enough, and has children, each of its children is ascertained to be far enough or not far enough. If a node is a leaf, its potential is found directly from particle-particle interactions. The potential due to each cluster is evaluated by replacing the cluster with a single body with mass equal to the sum of masses of all bodies, located at the center of mass of the cluster.

Greengard and Rokhlin made a breakthrough by developing the first $O(n)$ algorithm called the Fast Multipole Method, to compute electrostatic potential. To estimate the potential at a location, the set of charges is grouped into clusters, and the effect of each cluster is approximated separately. Some aspects of the algorithm were improved in [5]. The potential function in FMM is approximated, to arbitrary accuracy, by a series expansion with terms consisting of spherical harmonics. However the overall theoretical set up of the algorithm is very complex and it is not intuitive to comprehend potential in terms of series expansions.

CHAPTER II

BACKGROUND

The fastest algorithm to solve the potential evaluation problem is the FMM. It is the first algorithm to achieve $O(n)$ complexity. The hallmark of this algorithm is the use of the cluster-cluster interactions, which had never been done before. Prior to FMM, the fast algorithms used cluster-particle interactions to reduce complexity to $O(n \log n)$. The main reason for not having a lower complexity was not using the cluster-cluster interactions. In fact it can be said heuristically that any $O(n)$ algorithm to solve the potential computation problem must exploit both cluster-particle and cluster-cluster interactions. In FMM, just as any other hierarchical algorithm, an oct-tree is constructed. A root node enclosing all particles is recursively sub-divided into eight child nodes until each leaf node has atmost a constant number of particles. For uniform distributions of particles, the oct-tree is balanced, whereas non-uniform distributions result in an unbalanced oct-tree.

In the FMM, the potential is expressed as a series expansion of *Spherical Harmonics*, which are general solutions of the Laplace equation. The Laplace equation in cartesian and spherical polar coordinates is

- Cartesian Coordinates

$$\bigtriangledown^2 \Phi = \frac{\partial^2 \Phi}{\partial x^2} + \frac{\partial^2 \Phi}{\partial y^2} + \frac{\partial^2 \Phi}{\partial z^2} = 0. \tag{2.1}$$

- Spherical Polar Coordinates

$$\bigtriangledown^2 \Phi = \left(\frac{1}{r^2}\right) \frac{\partial}{\partial r} \left(\frac{r^2 \partial \Phi}{\partial r}\right) + \left(\frac{1}{r^2 \sin\theta}\right) \frac{\partial}{\partial \theta} \left(\sin\theta \frac{\partial \Phi}{\partial \theta}\right) + \left(\frac{1}{r^2 \sin^2\theta}\right) \left(\frac{\partial^2 \Phi}{\partial \phi^2}\right) = 0. \tag{2.2}$$

Solutions of the Laplace equation (2.1, 2.2) are called *Harmonic Functions*. The potential function satisfies the Laplace equation and hence is a harmonic function. Any harmonic

function can be expressed as

$$\Phi(r,\theta,\phi) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \left(L_n^m r^n + \frac{M_n^m}{r^{n+1}} \right) Y_n^m(\theta,\phi), \tag{2.3}$$

where

$$Y_n^m(\theta,\phi) = \sqrt{\frac{(n-|m|)!}{(n+|m|)!}}\, P_n^{|m|}(\cos\theta)\ \exp(im\phi), \tag{2.4}$$

where $P_n^{|m|}(\cos\theta)$ are the associated Legendre functions of degree n and order $|m|$, and $i = \sqrt{-1}$.

The terms $Y_n^m(\theta,\phi)r^n$, $0 \le n \le p$, $n \le m \le -n$ are called spherical harmonics of degree n and the terms $\frac{1}{r^{n+1}} \cdot Y_n^m(\theta,\phi)$, $0 \le n \le p$, $n \le m \le -n$ are called spherical harmonics of degree $(-n-1)$. The coefficients L_n^m and M_n^m are the moments of the expansion, and more specifically, Local and Multipole expansion coefficients, respectively.

A. Mathematical Preliminaries

The following theorem expresses the potential due to charges strictly enclosed within a spherical region at any point outside the region, as shown in Fig. 1. The origin is located at the center of the sphere.

Theorem II.1 (Multipole Expansion) *Suppose there are k charges, $q_j(\rho_j,\alpha_j,\beta_j), j = 1,\ldots,k$, with $|\rho_j| < a$. The potential at any point $P(r,\theta,\phi)$, $r > a$, is given by*

$$\Phi(P) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \frac{M_n^m}{r^{n+1}}\, Y_n^m(\theta,\phi), \tag{2.5}$$

where

$$M_n^m = \sum_{j=1}^{k} q_j\, \rho_j^n\, Y_n^{-m}(\alpha_j,\beta_j). \tag{2.6}$$

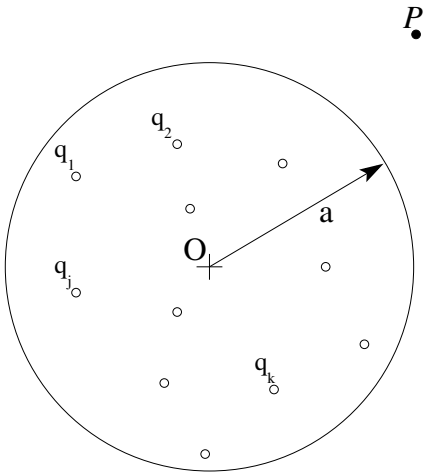

Fig. 1. Multipole expansion of the potential due to a set of charges enclosed in a sphere at any point P outside the sphere.

However, the potential is expressed as an approximation given by

$$\hat{\Phi}(P) = \sum_{n=0}^{p} \sum_{m=-n}^{n} \frac{M_n^m}{r^{n+1}} Y_n^m(\theta,\phi), \tag{2.7}$$

where p is called the *degree* of the multipole expansion. The error in approximating the potential, for a degree $p \geq 1$ of multipole expansion, is given by

$$\left|\Phi(P) - \hat{\Phi}(P)\right| \leq \frac{\sum_{j=1}^{k} |q_j|}{r-a} \left(\frac{a}{r}\right)^{p+1}. \tag{2.8}$$

Thus, the potential due to charges enclosed in a spherical region, irrespective of the number of charges in the region, is expressed as a finite series expansion consisting of $(p+1)^2$ terms.

The next theorem expresses the potential due to charges strictly outside a spherical region at any point inside the region. (See Fig. 2).

Theorem II.2 (Local Expansion) *Suppose there are k charges $q_j(\rho_j,\alpha_j,\beta_j), j = 1,\ldots,k$, with $|\rho_j| > a$. The potential at any point $P(r,\theta,\phi)$, $r < a$, is given by*

$$\Phi(P) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} L_n^m \, r^n \, Y_n^m(\theta,\phi), \tag{2.9}$$

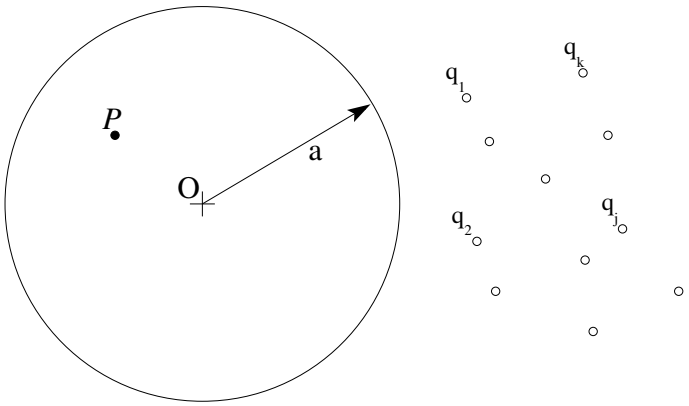

Fig. 2. Local expansion of the potential due to a set of charges outside a sphere at any point P inside the sphere.

where

$$L_n^m = \sum_{j=1}^{k} \frac{q_j \, Y_n^{-m}(\alpha_j, \beta_j)}{\rho_j^n}. \tag{2.10}$$

The potential can be approximated as

$$\hat{\Phi}(P) = \sum_{n=0}^{p} \sum_{m=-n}^{n} L_n^m \, r^n \; Y_n^m(\theta, \phi), \tag{2.11}$$

where p is called the *degree* of the local expansion. The error in approximating the potential, for a degree $p \geq 1$ of local expansion, is given by

$$\left| \Phi(P) - \hat{\Phi}(P) \right| \leq \frac{\sum_{j=1}^{k} |q_j|}{a - r} \left(\frac{r}{a} \right)^{p+1}. \tag{2.12}$$

Thus, the potential due to charges outside a spherical region, irrespective of the number of charges, is expressed as a finite series expansion consisting of $(p+1)^2$ terms.

The next theorem expresses the potential due to a set of charges enclosed in a spherical region, at any point outside a larger sphere that encloses the spherical region containing the charges. The origin is located at the center of the larger sphere, and lies outside the smaller sphere containing the charges. (see Fig. 3).

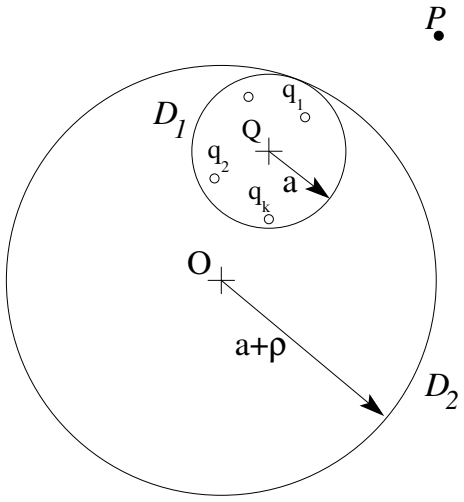

Fig. 3. Translation of multipole expansion of the potential due to a set of charges in a sphere D_1 to the sphere D_2.

Theorem II.3 (Translation of Multipole Expansion) *Suppose there are k charges given by $q_j(\rho_j, \alpha_j, \beta_j)$, $j = 1, \ldots, k$ in a sphere D_1 of radius a, with center at $Q(\rho, \alpha, \beta)$, and for points $P(r, \theta, \phi)$ outside D_1, potential due to these charges is given by the multipole expansion*

$$\Phi(P) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \frac{O_n^m}{r'^{n+1}} Y_n^m(\theta', \phi'), \tag{2.13}$$

where $\vec{PQ} = (r', \theta', \phi')$, and O_n^m is the multipole expansion w.r.t. the origin Q. For any point $P(r', \theta', \phi')$ outside the sphere D_2 of radius $(a+\rho)$ the potential is given by

$$\Phi(P) = \sum_{j=0}^{\infty} \sum_{m=-j}^{j} \frac{M_j^k}{r^{j+1}} Y_j^k(\theta, \phi), \tag{2.14}$$

where

$$M_j^k = \sum_{n=0}^{j} \sum_{m=-n}^{n} \frac{O_{j-n}^{k-m}\, i^{|k|-|m|-|k-m|}\, A_n^m\, A_{j-n}^{k-m}\, \rho^n\, Y_n^{-m}(\alpha, \beta)}{A_j^k}, \tag{2.15}$$

$$A_n^m = \frac{(-1)^n}{\sqrt{(n-m)!(n+m)!}}. \tag{2.16}$$

The potential can be approximated by

$$\hat{\Phi}(P) = \sum_{j=0}^{p} \sum_{k=-j}^{j} \frac{M_j^k}{r^{j+1}} Y_j^k(\theta, \phi), \tag{2.17}$$

where p is called the *degree* of the multipole expansion. The error in approximating the potential, for a degree $p \geq 0$ of the expansion, is given by

$$\left|\Phi(P) - \hat{\Phi}(P)\right| \leq \frac{\sum_{j=1}^{k} |q_j|}{r - (a+\rho)} \left(\frac{a+\rho}{r}\right)^{p+1}. \tag{2.18}$$

Thus, the potential of a set of charges in a spherical region, irrespective of the number of charges, is expressed as a truncated series expansion of $(p+1)^2$ terms.

The following theorem expresses the potential due to a set of charges enclosed in a spherical region, at any point inside another spherical region which is disjoint from the first region, (see Fig. 4).

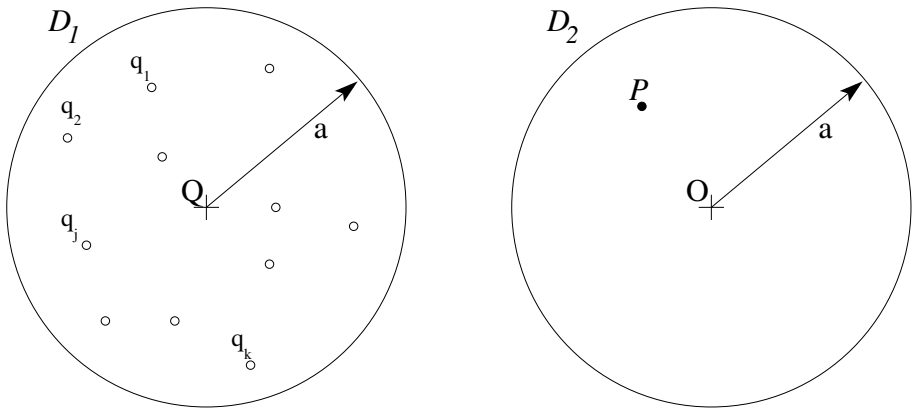

Fig. 4. Translation of multipole expansion of the potential due to a set of charges in a sphere D_1 to the local expansion inside a disjoint sphere D_2.

Theorem II.4 (Conversion of Multipole Expansion into a Local Expansion) *Suppose there are k charges $q_j(\rho_j, \alpha_j, \beta_j)$, $j = 1, \ldots, k$, inside a sphere D_1 of radius a, and center at $Q(\rho, \alpha, \beta)$, and $\rho_j > (c+1)a$, where $c > 1$. The corresponding multipole expansion converges inside the sphere D_2 of radius a centered at the origin. Inside D_2, the potential*

due to the charges $q_1, q_2, \ldots, q_k$ is given by a local expansion

$$\Phi(P) = \sum_{j=0}^{\infty} \sum_{k=-j}^{j} L_j^k \, Y_j^k(\theta, \phi) \, r^j, \tag{2.19}$$

where

$$L_j^k = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \frac{O_n^m \, i^{|k-m|-|k|-|m|} \, A_n^m \, A_j^k \, Y_{j+n}^{m-k}(\alpha, \beta)}{(-1)^n A_{j+n}^{m-k} \rho^{j+n+1}}, \tag{2.20}$$

in which O_n^m, $0 \leq n \leq p$, $-n \leq m \leq n$ is the multipole expansion coefficients of the charges w.r.t. Q, and A_n^m is as defined in (2.16).

The potential is approximated as

$$\hat{\Phi}(P) = \sum_{j=0}^{p} \sum_{k=-j}^{j} L_j^k \, Y_j^k(\theta, \phi) \, r^j, \tag{2.21}$$

where p is the degree of local expansion. The error in approximating the potential for a degree $p \geq 1$ of local expansion is given by

$$\left| \Phi(P) - \sum_{j=0}^{p} \sum_{k=-j}^{j} L_j^k \, Y_j^k(\theta, \phi) \, r^{j+1} \right| \leq \frac{\sum_{j=1}^{k} |q_j|}{ca - a} \left(\frac{1}{c} \right)^{p+1}. \tag{2.22}$$

Thus, the potential due to a set of n charges in a sphere, at a set of m points in another sphere can be computed in $O(m+n)$ operations as against $O(mn)$ operations taken by the naive method (1.2).

The next theorem presents the local expansion of the potential inside D_1 when the local expansion is known inside the sphere D_2. (see Fig. 5).

Theorem II.5 (Translation of Local Expansion) *If $Q(\rho, \alpha, \beta)$ is the origin of the local expansion*

$$\Phi(P) = \sum_{n=0}^{p} \sum_{m=-n}^{n} O_n^m \, Y_n^m(\theta', \phi') \, r'^n, \tag{2.23}$$

where $P = (r, \theta, \phi)$, $r < a$ and $\vec{PQ} = (r', \theta', \phi')$, $r' < a + \rho$, then

$$\Phi(P) = \sum_{j=0}^{p} \sum_{k=-j}^{j} L_j^k \, Y_j^k(\theta, \phi) \, r^j, \tag{2.24}$$

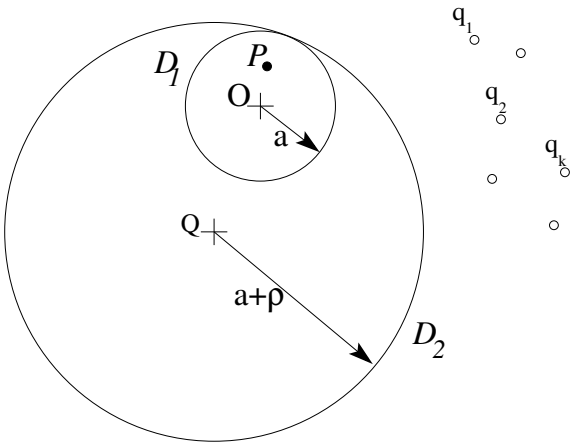

Fig. 5. Translation of local expansion of the potential from sphere D_2 to D_1. The charges are located outside D_2.

where

$$L_j^k = \sum_{n=j}^{p} \sum_{m=-n}^{n} \frac{O_n^m \, i^{|m|-|m-k|-|k|} \, A_{n-j}^{m-k} \, A_j^k \, Y_{n-j}^{m-k}(\alpha,\beta) \, \rho^{n-j}}{(-1)^{n+j} \, A_n^m}, \tag{2.25}$$

where A_n^m is as defined in (2.16).

Thus, given the local expansion of a region due to a set of charges outside the region, we can evaluate the corresponding local expansion in an enclosed region.

B. The Fast Multipole Method

An oct-tree is constructed by enclosing the charges in a root node and recursively subdividing the node till each leaf node has atmost a constant number of charges. By a leaf node, we mean a non-empty leaf node and by an internal node, a node having children. A node a is *far enough* from a node b if they are separated by a distance of at least the length of node b.

The non-adaptive version of FMM, where the oct-tree is balanced is suitable only for uniform point distribution. An adaptive version of FMM is more versatile as it can be used for non-uniform distributions as well.

The basic strategy here is to cluster nodes at various levels and compute interactions

with other far-away clusters by using multipole and local expansions. Potential due to nearby charges are evaluated directly.

1. Computation of Multipole Expansions

The multipole expansions for each leaf are computed directly from its charges by Theorem II.1. The multipole expansions of the parents of the leaves are computed by translating the multipole expansions of each of its children to itself and summing them up (Theorem II.3). Likewise, the multipole expansions of all nodes are evaluated, in a bottom-up manner.

2. Computation of Local Expansions

The local expansions of the nodes are computed after the computation of the multipole expansions of all nodes. The tree is traversed top-down starting from the root and the local expansions of the nodes are computed during the traversal. The local expansions due to charges far enough from a node, but not far enough from the node's parent are computed. Then, the local expansion due to charges far enough from the node's parent is computed from its parent's local expansion.

At each non-empty node, we maintain five lists called $l1$, $l2$, $l3$, $l4$, and, $l5$. Lists $l1$ and $l3$ of internal nodes are empty. For a node b,

- List $l1$ is the set of all leaf nodes adjacent to b.
- List $l2$ consists of all far enough nodes that are children of b's parent's immediate neighbors of the same size. Nodes in list $l2$ of b are of the same size as b and are far enough from b.
- List $l3$ is the set of all descendents of b's immediate neighbors of the same size that are not adjacent to b but their parents are.

- List $l4$ is the set of all nodes x such that b is in the list $l3$ of x. List $l4$ of node b are leaf nodes of size larger than b.

- List $l5$ is the set of all nodes far enough from b's parent. Nodes in list $l5$ contribute to the local expansion in phase 2.

The local expansion due to nodes in list $l2$ is evaluated by translating their multipole expansions to the local expansions w.r.t. b's center using Theorem II.3, and adding them up. Local expansion due to list $l4$ is computed by adding up local expansions of the charges in the nodes in list $l4$. Local expansion due to list $l5$ is computed by translating the local expansion of each parent to each of its children in a top-down manner using Theorem II.5. The total local expansion at the boxes is evaluated by summing up the partial local expansions due to lists $l2$, $l4$, and $l5$.

3. Potential Evaluation

Potential at any charge point is the sum of the three types of partial potentials ϕ_1, ϕ_2, ϕ_3, defined as

- ϕ_1 is the potential due to all charges in list $l1$ and is computed by direct interactions,

- ϕ_2 is the potential due to all charges outside b's immediate neighbors of the same size and is evaluated by applying the local expansion of b at the charge location in b, and

- ϕ_3 is the potential due to all charges in list $l3$ and is found by summing up the partial potentials obtained by applying the multipole expansions of each of the nodes in list $l3$ at the charge location in b.

4. The Fast Multipole Algorithm

The algorithm consists of the following steps:

Step 1 Construct the oct-tree by recursively sub-dividing the root till each leaf has atmost k_0 charges.

Step 2a Evaluate the multipole expansion coefficients of all leaf nodes using Theorem II.1.

Step 2b Compute the multipole expansion coefficients of all internal nodes in a bottom up manner. Multipole expansion coefficients of each parent node is evaluated from the multipole expansion coefficients of its children using Theorem II.3.

Step 3a Form the adjacency lists of the root's children, each of which has rest of its siblings in its list $l1$.

Step 3b Form the adjacency lists of each node by traversing the adjacency list of its parent.

Step 4 Make the lists $l1$, $l2$, $l3$, and $l4$ for each node by traversing its adjacency list.

Step 5a At each node, form its local expansion coefficients due to nodes in its list $l2$ using Theorem II.4 and list $l4$ using Theorem II.2.

Step 5b Compute the local expansion coefficients of each node due to charges far enough from its parent by translating the local expansion (evaluated in Step 5a) of its parent to itself using Theorem II.5.

Step 5c Add the local expansion coefficients computed in Steps 5a and 5b to evaluate the complete local expansion coefficients of each node.

Step 5d At every charge location in each leaf node, compute the partial potential, ϕ_2 due to nodes in the leaf's list $l2$, list $l4$, and list $l5$ from its local expansion coefficients by using Theorem II.2.

Step 6a At every charge location in each leaf node, compute the partial potential, ϕ_3 due to nodes in the leaf's list $l3$ by using Theorem II.1.

Step 6b Compute at each charge location the partial potential, ϕ_1 due to direct interactions using 1.2.

Step 7 Compute the potential at each charge location as a sum of partial potentials ϕ_1, ϕ_2 and ϕ_3.

5. Analysis of FMM

The following analysis of FMM is for a uniform particle distribution. Consider the following parameters for the potential evaluation problem

(i) a set of N charges,

(ii) a degree p for the multipole and local expansions, and

(iii) the maximum number k_0 of charges per leaf.

In the following analysis

(a) flops stand for floating point operations,

(b) *mult-coeff* unit is a unit of multipole coefficient evaluation,

(c) *mult-mult* unit is a unit of translation of multipole coefficients,

(d) *mult-loc* unit is a unit of translation of multipole to local coefficients,

(e) *loc-loc* is a unit of local translation.

The complexity of various steps of the algorithm is given by

- Step 1: $(N/k_0)\log(N/k_0)$ flops
- Step 2a: $N(p+1)^2$ mult-coeff units
- Step 2b: $\frac{8N}{7k_0}\ (p+1)^4$ mult-mult conv units
- Steps 3a, 3b, 4: $430\ \frac{8N}{7k_0}$ flops

- Step 5a: 189 $\frac{8N}{7k_0}\ (p+1)^4$ mult-loc conv units
- Step 5b: $\frac{8N}{7k_0}\ (p+1)^4$ loc-loc conv units
- Step 5c: $\frac{27}{2}\ Nk_0$ flops
- Step 5d: $N(p+1)^2$ loc-coeff flops
- Step 6a, 6b: $2N$ integer operations

A *mult-coeff* unit equals 18 flops, *mult-mult conv* unit equals 44 flops, *mult-loc conv* unit equals 44 flops, *loc-loc conv* unit equals 44 flops, and *loc-coeff* unit equals 18 flops. Thus, the computational complexity can be approximated by $44\ \times\ 189\ (p+1)^4\ N$.

The storage requirements for FMM are discussed next. A node unit is a memory sufficient to hold a node of the tree, which is about 128 bytes, and a complex unit is a memory large enough to hold two double precision members, which is 16 bytes.

- Step 1: $\frac{8N}{k_0}$ node units = 128 $(\frac{8N}{k_0})$ bytes,
- Steps 2a, 2b: $\frac{8N}{7k_0}\ (p+1)^2$ complex units = 16 $(\frac{8N}{7k_0}\ (p+1)^2)$ bytes,
- Steps 3a, 3b, 4: 430 $\frac{8N}{7k_0}$ single precision units = 4 $(430\ \frac{8N}{7k_0})$ bytes,
- Steps 5a, 5b, 5c, 5d: $\frac{8N}{7K_0}\ (p+1)^2$ complex units = 16 $(\frac{8N}{7K_0}\ (p+1)^2)$ bytes,
- Steps 6a, 6b: $\frac{8N}{7K_0}$ double precision units = 8 $(\frac{8N}{7K_0})$ bytes.

Overall, the algorithm requires approximately $(32(p+1)^2+1720)N$ bytes.

CHAPTER III

THE NEW APPROACH TO POTENTIAL EVALUATION

In this chapter, we develop a new framework for evaluating potential due to a set of charges. The framework is described using *Radial Basis Functions* (RBFs) [3], which are radially symmetric functions. RBFs have excellent approximation properties due to which they are used to approximate multivariate functions such as potential due to a point charge. To approximate the potential function, a class of RBFs known as *Spherical Harmonics* are used. The spherical harmonics are the general solution of the *Laplace Equation* (2.2). Since the potential function due to a charge satisfies the Laplace equation, it is logical to express the potential as a weighted sum of spherical harmonics centered at the charge point.

Another reason for using the spherical harmonics is that they are orthogonal functions. Orthogonal functions are linearly independent and hence the set of spherical harmonics forms a basis for the potential functions. Orthogonality of the spherical harmonics is most useful, when, given the potential w.r.t. a basis of spherical harmonics centered at x_1, it is required to express the potential w.r.t. a basis of spherical harmonics centered at x_2. We can evaluate the weights of the spherical harmonics centered at x_2 by using orthogonal projection of the first basis onto the second.

As in Barnes-Hut and FMM, an oct-tree is formed by starting from a root box (or node) enclosing all particles, which is recursively sub-divided into eight child nodes until each leaf node has atmost a constant number of particles. This constant, k_0, is found experimentally and fixed such that the overall running time of the algorithm is minimized. We use a value of 35 for k_0 that was experimentally determined to be the optimal value. Empty leaf boxes are addressed as empty nodes. For uniform distributions of particles, the oct-tree will be balanced, and, non-uniform distributions would result in an unbalanced oct-tree.

A. Multipole Coefficients of Charge Clusters

For a given basis of spherical harmonics, the *Multipole Coefficients* represent the potential due to the charges inside a region at an evaluation point that is sufficiently far from the region. The highest degree of the functions in the basis, denoted by p, dictates the accuracy of the potential. The number of terms in the multipole coefficient vector is $(p+1)^2$.

Consider a set of charges $q_j(\rho_j, \alpha_j, \beta_j), j = 1, \ldots, k$ with $|\rho_j| < a$, i.e., in space strictly enclosed by a sphere of radius a. The multipole coefficients due to the charges at any point $P(r, \theta, \phi)$ with $r > a$ is given by

$$M_n^m = \sum_{j=1}^{k} q_j \, \rho_j^n \, Y_n^{-m}(\alpha_j, \beta_j), \tag{3.1}$$

where $0 \leq n \leq p, -n \leq m \leq n$, and

$$Y_n^{-m}(\alpha, \beta) = \sqrt{\frac{(n-|m|)!}{(n+|m|)!}} \, P_n^{|m|}(\cos\alpha) \, \exp(-im\beta), \tag{3.2}$$

where $P_n^{|m|}(\cos\alpha)$ is the Legendre function of degree n and order $|m|$.

The vector $M_n^m, 0 \leq n \leq p, -n \leq m \leq n$ is called the *multipole coefficient vector* or simply the *multipole vector*.

The potential due to the charges in the sphere is expressed as the weighted sum of a class of spherical harmonic functions called multipole spherical harmonic functions w.r.t. the center of the sphere. The potential at a point $P(r, \theta, \phi)$, $r > a$, is given as

$$\Phi(P) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} M_n^m \, \frac{Y_n^m(\theta, \phi)}{r^{n+1}}, \tag{3.3}$$

and can be approximated using the multipole vector as

$$\hat{\Phi}(P) = \sum_{n=0}^{p} \sum_{m=-n}^{n} M_n^m \, \frac{Y_n^m(\theta, \phi)}{r^{n+1}}, \tag{3.4}$$

where M_n^m is a term in the multipole coefficient vector and p is the multipole degree.

The vector $\frac{1}{r^{n+1}} Y_n^m(\theta,\phi), 0 \leq n \leq p, -n \leq m \leq n$ is called the multipole spherical harmonic function vector. The potential $\Phi(P)$ is an inner-product of the multipole coefficient vector and the vector of multipole spherical harmonic functions evaluated at P.

The multipole coefficient vectors of the leaf nodes are computed directly by (3.1). The multipole coefficients of the internal nodes of the tree are computed in a bottom-up manner. Multipole coefficients of a node are computed by adding the projections of the multipole coefficients of its child nodes. Consider a parent node with a set of charges strictly enclosed in it. The potential due to the charges in the node can be expressed as the weighted sum of its multipole spherical harmonics (3.3)-(3.4). The potential can also be computed as the sum of partial potentials of the charges in each of its child nodes, given by the weighted sum of their respective multipole spherical harmonics. Thus

$$f_p^T \; c_p = \sum_{j=1}^{8} f_j^T \; c_j, \tag{3.5}$$

where f_p is the vector of multipole spherical harmonics of the parent, c_p is the vector of multipole coefficients of the parent, f_j is the vector of multipole spherical harmonics (3.4) of the jth child, and c_j is the vector of multipole coefficients of the jth child. The only unknown c_p, can be isolated by taking the orthogonal complement of f_p on either side of (3.5). After appropriate scaling, we have

$$c_p = \sum_{j=1}^{8} M_j \; c_j, \tag{3.6}$$

where M_j is the lower triangular projection matrix for the jth child and is given as

$$M_j(n,m,n_1,m_1) = J_{m-m_1}^{m_1} \; C(n,m,n_1,m_1) \; \rho^{n-n_1} \; Y_{n-n_1}^{-(m-m_1)}(\alpha,\beta), \tag{3.7}$$

where $0 \leq n \leq p, -n \leq m \leq n, 0 \leq n_1 \leq p, -n_1 \leq m_1 \leq n_1$. The row is denoted by (n,m) and the column is denoted by (n_1,m_1). The coordinates of the jth child's center w.r.t. to the parent's center are assumed to be (ρ,α,β), and

$$J_{m-m_1}^{m_1} = i^{|m|-|m-m_1|-|m_1|}, \tag{3.8}$$

$$C(n,m,n_1,m_1) = \sqrt{\frac{(n+m)!\,(n-m)!}{(n_1+m_1)!(n-n_1+m-m_1)!\,(n_1-m_1)!((n-n_1)-(m-m_1))!}}. \tag{3.9}$$

Further, M_j can be factored as follows

$$M_j = D_\rho\, \tilde{M}_j\, D_\rho^{-1}, \tag{3.10}$$

where $\tilde{M}_j$ is the conversion matrix between a parent and its jth child, independent of the level. D_ρ is a diagonal matrix such that

$$D_\rho(n,m,n_1,m_1) = \begin{cases} \rho^n, & n = n_1, m = m_1 \\ 0, & \text{otherwise} \end{cases} \tag{3.11}$$

Combining equations (3.6) and (3.10), we get

$$c_p = \sum_{j=1}^{8} D_\rho\, \tilde{M}_j\, D_\rho^{-1}\, c_j. \tag{3.12}$$

B. Local Coefficients

Local Coefficients of a region represent the potential due to charges sufficiently far from the region at an evaluation point inside the region. The local coefficients have a degree which dictates the accuracy of the potential. For a degree p, the number of terms in the local coefficient vector is $(p+1)^2$. Consider a set of charges $q_j(\rho_j,\alpha_j,\beta_j), j = 1,\ldots,k$ with $|\rho_j| > a$. The local coefficients due to the charges at any point $P(r,\theta,\phi)$, $r < a$, is given by

$$L_n^m = \sum_{j=1}^{k} q_j\, \frac{Y_n^{-m}(\alpha_j,\beta_j)}{\rho_j^{n+1}}, \tag{3.13}$$

where $0 \leq n \leq p, -n \leq m \leq n$. The vector $L_n^m, 0 \leq n \leq p, -n \leq m \leq n$ is called the *local coefficient vector*, or simply the *local vector*.

The potential due to the charges in the sphere is expressed as the weighted sum of the local spherical harmonic functions w.r.t. the center of the sphere. The potential at a point $P(r, \theta, \phi)$, $r < a$, is given as

$$\Phi(P) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} L_n^m \, Y_n^m(\theta, \phi) \, r^n, \tag{3.14}$$

and can be approximated using the local coefficient vector as

$$\hat{\Phi}(P) = \sum_{n=0}^{p} \sum_{m=-n}^{n} L_n^m \, Y_n^m(\theta, \phi) \, r^n, \tag{3.15}$$

where L_n^m is a term in the local coefficient vector, p is the multipole degree. The vector $Y_n^m(\theta, \phi) r^n$, $0 \leq n \leq p$, $-n \leq m \leq n$ is called the local spherical harmonic function vector. The potential $\Phi(P)$ is an inner-product of the local coefficient vector and the vector of local spherical harmonic functions evaluated at P.

A node a is said to be *far enough* from a node b if they are separated by at least the length of node b. The local coefficients of the nodes are computed in two phases. The local expansions due to charges far enough from the node, but not far enough from the node's parent are computed using Theorem II.4. Then, the local expansion due to charges far enough from the node's parent is accumulated (i.e., passed on) at the node from its parent using Theorem II.5.

1. Phase 1

As in FMM, lists $l1$, $l2$, $l3$, and $l4$ are maintained at each non-empty node. However, the characterization of the lists is slightly different from that in FMM. Lists $l1$ and $l3$ of internal nodes are empty. For a node b,

- List $l1$ consists of leaf nodes either adjacent to b or that are descendents of b's

colleagues whose parents are adjacent to b but are themselves not adjacent to b. A colleague is an adjacent node of the same size.

- List $l2$ consists of all far enough internal nodes which are children of b's parent's colleagues. Nodes in list $l2$ of b are of the same size as b and are far enough from b.

- List $l3$ is the set of all internal nodes that are descendents of b's colleagues, and whose parents are adjacent to b, but the nodes are themselves not adjacent to b.

- List $l4$ is the set of all nodes x such that b is in the list $l3$ of x. List $l4$ of node b are all leaf nodes of size larger than or equal to b.

Phase 1 consists of computing local coefficients due to the charges in lists $l2$ and $l4$. Local coefficients due to list $l4$ are computed by adding the local coefficients due to each charge in the nodes in list $l4$ using equation (3.13). Local coefficients due to list $l2$ are computed by taking the sum of the projections of the multipole series of the nodes in list $l2$ onto the local spherical harmonic functions of the current node, b.

The potential due to the charges in a node y at any point in node x can be expressed as a weighted sum of local spherical harmonics of x as shown in (3.14). The potential can also be computed as a weighted sum of the multipole spherical harmonics of y given in (3.3). Thus

$$g_x^T \; d_x = f_y^T \; c_y, \tag{3.16}$$

where g_x is the vector of local spherical harmonics of node x, d_x is the vector of local coefficients or the weights of node x, f_y is the vector of multipole spherical harmonics of node y, and c_y is the vector of multipole coefficients or the weights of node y. The only unknown d_x can be isolated by multiplying on either sides of the equation 3.16 with the orthogonal complement of g_x. After appropriate scaling, we have

$$d_x = T_{xy}\, c_y, \tag{3.17}$$

where T_{xy} is the conversion matrix for node y, and is given by

$$T_{xy}(n,m,n_1,m_1) = J_m^{m_1}\, \frac{A_n^m\, A_{n_1}^{m_1}}{A_{n+n_1}^{m-m_1}}\, \frac{Y_{n+n_1}^{-(m-m_1)}(\alpha,\beta)}{\rho^{n+n_1+1}}, \tag{3.18}$$

where, $0 \le n \le p, -n \le m \le n, 0 \le n_1 \le p, -n_1 \le m_1 \le n_1$. The row is denoted by (n,m) and the column is denoted by (n_1,m_1). The coordinates of node y's center w.r.t. to the x's center are assumed to be (ρ,α,β) , and

$$J_m^{m_1} = \frac{i^{|m-m_1|-|m|-|m_1|}}{(-1)^{n_1}}, \tag{3.19}$$

$$A_n^m = \frac{(-1)^n}{\sqrt{(n-m)!(n+m)!}}. \tag{3.20}$$

The local coefficient vector of x can be obtained from matrix-multiplication with multipole coefficient vector of y. However, this is an implementation detail. Theoretically, the local coefficient vector of x is obtained by taking the projections of spherical harmonic function of y onto the local spherical harmonic functions of x.

2. Phase 2

The local coefficients at each node are computed by adding the local coefficients obtained from lists $l2$ and $l4$ and the projections of the local coefficients of the node's parent onto the node's local spherical harmonic functions. This process is called as accumulation of local coefficients from parent to child. The potential at a point inside a node due to a set of charges outside the node's parent can be expressed as the weighted sum of the node's local spherical harmonics and also as the weighted sum of its parent's local spherical harmonics. Thus

$$g_c^T \; d_c = g_p^T \; d_p, \tag{3.21}$$

where g_c is the vector of local spherical harmonics of the child, and d_c is the vector of local coefficients of the child, g_p is the vector of local spherical harmonics of the parent, d_p is the vector of local coefficients of the parent. The unknown vector d_c can be isolated by multiplying on either side of the equation (3.21) with the orthogonal complement of g_c. Appropriate scaling leads to

$$d_c = L_p \; d_p, \tag{3.22}$$

where L_p is the upper triangular projection matrix

$$L_p(n,m,n_1,m_1) = J_m^{m-m_1,n-n_1} \; \frac{A_{n-n_1}^{m-m_1} \; A_{n_1}^{m_1}}{A_n^m} \; \rho^{n-n_1} \; Y_{n-n_1}^{m-m_1}(\alpha,\beta), \tag{3.23}$$

where $0 \leq n \leq p, -n \leq m \leq n, 0 \leq n_1 \leq p, -n_1 \leq m_1 \leq n_1$. The row is denoted by (n,m) and the column is denoted by (n_1,m_1). The coordinates of the parent's center w.r.t. to the child's center are assumed to be (ρ,α,β) , and

$$J_m^{m-m_1,n-n_1} = \frac{i^{|m|-|m-m_1|-|m_1|}}{(-1)^{n-n_1}}. \tag{3.24}$$

Further, L_p can be factored as follows

$$L_p = D_\rho^{-1} \; \tilde{L_\rho} \; D_\rho, \tag{3.25}$$

where $\tilde{L_\rho}$ is the conversion matrix between a node and its parent, independent of the level. Combining equations (3.22) and (3.25), we get

$$d_c = D_\rho^{-1} \; \tilde{L_\rho} \; D_\rho \; d_p. \tag{3.26}$$

Thus, the local coefficients of each parent are accumulated at each of its children. The local coefficients of a node are computed as the sum of the local coefficients due to list $l2$ and list $l4$ evaluated in phase 1, and the local coefficients obtained by accumulation from its parent in phase 2.

C. Potential Evaluation

To evaluate the potential at an observation point inside a leaf node, the domain can be partitioned into *near*, *far*, and *very far* regions. For the leaf node, the *near* region consists of nodes in list $l1$, the *far* region consists of lists $l2$, $l3$, and $l4$, and the *very far* region is accounted for by the accumulation of local coefficients from each parent node to each of its children. The charges in these regions are grouped into clusters, and the effect of each cluster is computed separately. The potential is computed as the sum of the potentials due to such clusters, resulting in an $O(n)$ algorithm.

For a point located in a leaf node n_l, the potential is computed as a sum of the *near-potential* and the *far-potential*. The *near-potential* is computed directly, i.e., from particle-particle interactions. The *far-potential* is expressed as the weighted sum of the local spherical harmonic functions of the node, with the local coefficients as the corresponding weights.

D. Matrix Structure

The hallmark of this approach is its simplicity in terms of structure. The evaluation of potential due to a set of charges is represented as a matrix-vector product

$$v = P\, q, \tag{3.27}$$

where q is the vector of n charges and v is the vector of potentials at the n charge locations. The $n \times n$ potential matrix, P is constructed as a product of five matrices as follows

$$P = B\,L^{-1}\,T\,M^{-1}\,A, \tag{3.28}$$

where A is an $(m(p+1)^2+n) \times n$ matrix that is constructed from the multipole vectors of unit charges in the leaf nodes of the oct-tree. Here, m is the number of nodes in the oct-tree. The $(m(p+1)^2+n) \times (m(p+1)^2+n)$ matrix M^{-1} is used to evaluate the multipole coefficients for the internal nodes of the oct-tree from the multipole coefficients of the leaves. The conversion of multipole coefficients into partial local coefficients of nearby nodes is represented by the $(m(p+1)^2+n) \times (m(p+1)^2+n)$ matrix T, and the $(m(p+1)^2+n) \times (m(p+1)^2+n)$ matrix L^{-1} is used to evaluate the local coefficients at the leaves. Finally, the $n \times (m(p+1)^2+n)$ matrix B is used to compute the potential using the local coefficients.

Composition of A: A is an $(m(p+1)^2+n) \times n$ matrix that converts the charge vector into a vector of multipole vectors of the charges in the tree. A consists of an $n \times n$ identity matrix at the bottom, on top of which is a $n(p+1)^2 \times n$ block diagonal matrix, and each block in it is the first column of the identity matrix of order $(p+1)^2$, and the rest of A consists of zeros. The vector

$$v_1 = A\,q, \tag{3.29}$$

has the vector of charges at the bottom, the multipole vectors of the leaves on top of it, and zeros on top of the vector. v_1 is in a form that can readily be multiplied by M^{-1}, to get v_2, the vector of multipole vectors of all nodes.

Composition of M_0^{-1}: The matrix M^{-1} encompasses the information for computation of the multipole coefficients of all the nodes in the tree in a bottom-up manner, using the multipole coefficients of the charges present in v_1. M is block upper-triangular, and can be written as

$$M = \begin{pmatrix} M_1 & & \\ & & \\ & & I_n \end{pmatrix}, \tag{3.30}$$

where M_1 is an upper triangular block matrix of size $m(p+1)^2$ with each block row corresponding to a node in the tree. The block rows in the lowest level of M correspond to the leaf nodes and the higher block rows correspond to the higher levels of the oct-tree in a bottom-up manner. A block row for an internal node represents the computation of its multipole coefficients from the multipole coefficients of the its children, as given in (3.6)

$$I\, c_p - \sum_{j=1}^{8} M_j\, c_j = 0, \tag{3.31}$$

The block row corresponding to a leaf has only the identity matrix on its diagonal. The block row of an internal node has an identity as its diagonal block and eight off-diagonal blocks of $-M_j, 1 \leq j \leq 8$, corresponding to each of its children. The column number where an M_j occurs is the same as the row number where the multipole coefficients of M_j are evaluated. The vector

$$v_2 = M^{-1}\, v_1, \tag{3.32}$$

consists of the charge vector q at the bottom, and multipole vectors of all nodes at the top.

Composition of T: T converts the vector of multipole vectors of all nodes into the vector of partial local vectors (i.e., local coefficients due to lists $l2$, $l3$, and $l4$), as follows

$$v_3 = T\, v_1, \tag{3.33}$$

T is a block matrix and can be written as

$$T = \begin{pmatrix} T_{11} & T_{12} \\ T_{21} & T_{22} \end{pmatrix}, \tag{3.34}$$

where T_{22}, is a matrix of size n which represents the partial potential due to direct interactions. T_{11} is a matrix of size $m(p+1)^2$ which captures the effect of nodes in list $l2$ converts the multipole coefficients in v_2 to the corresponding local coefficients in v_3. T_{12} captures the effect of nodes in list $l4$, and T_{21} captures the effect of nodes in list $l3$.

T_{12} can be considered as a set of block rows, with the lowest rows corresponding to the leaves and the higher rows corresponding to the higher level internal nodes. The columns of a block row corresponding to a node are the local coefficient vectors of the corresponding leaf nodes, which are in the list $l4$ of the node. T_{21} can be considered as a set of block columns, with the rightmost rows corresponding to the leaves and the rows to the left corresponding to the higher level internal nodes. A block column corresponding to a node has the multipole spherical harmonic functions evaluated at the charge locations of those leaf nodes that have this node in their list $l3$.

Each block row in T corresponds to a node in the tree, with the lowest rows corresponding to the leaves and the higher rows corresponding to the higher level internal nodes of the tree. A nonzero block (x,y) in T represents the equation (3.17) where the local coefficients due a node y in list $l2$ of the current node x are computed. The diagonal block in each block row is zero. The matrix T has a symmetric non-zero block structure. The vector

$$v_3 = T\ v_2, \tag{3.35}$$

has the sum of partial potential due to charges in list $l1$ and direct interactions at the bottom, on top of which are the local coefficient vectors due to lists $l2$, $l3$, and $l4$, corresponding to each node.

Composition of L: L^{-1} contains the information for computation of the local coefficients of all the nodes, as a sum of the partial local coefficients due to lists $l2$, $l3$, $l4$ and due to accumulation from parent. L is block lower-triangular, and can be written as

$$L = \begin{pmatrix} L_1 & & \\ & & \\ & & I_n \end{pmatrix}, \tag{3.36}$$

where L_1 is a block lower-triangular matrix of size $m(p+1)^2$ with each block row corresponding to a node in the tree. The block rows in the lowest level of L_1 correspond to the leaf nodes and the higher block rows correspond to the nodes at higher levels of the oct-tree. Each block row represents the computation of the local coefficients of the corresponding node from the local coefficients due to lists $l2$, $l3$, $l4$ and the local coefficients due to accumulation from its parent, as given in (3.22)

$$I\, d_c - L_p\, d_p = 0. \tag{3.37}$$

The block row corresponding to a leaf has only the identity matrix on its diagonal. The block row of an internal node has an identity as its diagonal block and an off-diagonal block of $-L_p$, corresponding to its parent. The column number where an L_p occurs is the same as the row number where the multipole coefficients of the parent are evaluated. The vector

$$v_4 = L^{-1}\, v_3, \tag{3.38}$$

consists of the sum of partial potential due to list $l1$ and direct interactions at the bottom, on top of which are the complete local vectors of all nodes.

Composition of B: B is an $n \times (m(p+1)^2 + n)$ matrix which converts v_4 into the potential vector, v. B consists of an $n \times n$ identity matrix at the right end, to the left of which is an $n \times n(p+1)^2$ block diagonal matrix, and each block is simply the vector of local spherical harmonics evaluated at the corresponding charge location , and the rest

of B consists of zeros. B is so constructed that the potential vector v, is obtained by the matrix-vector product of B and v_4, as follows

$$v = B\, v_4. \tag{3.39}$$

1. Benefits of Matrix Structure

The matrix structure makes it easy to comprehend the process of converting the charge vector into the potential vector. It also provides a convenient top level view of the algorithm. Our approach leads to a number of algorithmic optimizations over existing techniques. Appropriate transformations of the multipole and local coefficients leads to conversion matrices that are independent of the tree hierarchy. Precomputation of these matrices results in significant reduction in computation.

2. The Algorithm

Step 1 Construct the oct-tree by recursively sub-dividing the root node till each leaf has a maximum of k_0 charges.

Step 2a Evaluate the multipole coefficients of all leaf nodes using (3.1).

Step 2b Construct recursively the multipole coefficients of all nodes in the tree in a bottom up manner. Multipole coefficients of each parent node is evaluated as a sum of matrix vector products of the multipole conversion matrix and the multipole coefficients of its children (3.12).

Step 3a Form the adjacency lists of the root's children, each of which has rest of its siblings in its list.

Step 3b Form the adjacency lists of each node by traversing the adjacency list of its parent. Note that the adjacency lists here are of half the length of their counterparts

in FMM.

Step 4 Make the lists $l1$, $l2$, $l4$ for each node by traversing its adjacency list. Note that here, the list $l3$ is not constructed at all, and the lengths of the lists $l1$, $l2$, and $l4$, are half the lengths of their counterparts in FMM.

Step 5a At each node, form its local coefficients due to nodes in its list $l2$ and list $l4$ using (3.17) and (3.13), respectively.

Step 5b Compute the local coefficients of each node due to the charges far enough from its parent as the product of the local conversion matrix and the local coefficient vector of the node's parent (evaluated in (3.26)).

Step 5c Add the local coefficients computed in Steps 5a and 5b to evaluate the complete local coefficients of each node. The local coefficients of a leaf represents the potential due to all charges far enough from the leaf.

Step 5d At every charge location in each leaf node, compute the partial potential ϕ_2 due to all charges far enough from the leaf from its local coefficient vector by using (3.15).

Step 6a At every charge location, compute the partial potential ϕ_3 due to nodes in the leaf's list $l3$ by using (3.1).

Step 6b At every charge location, compute the partial potential ϕ_1 due to nodes in the leaf's list $l1$ using (1.2).

Step 7 Compute the potential at each charge location as the sum of partial potentials ϕ_1, ϕ_2 and ϕ_3.

3. Analysis of the New Approach

The complexity of this algorithm, in terms of order and the constant multiplying the order, is similar to that of FMM. However, new approach is faster due to smaller constants. For the purpose of the following analysis, the charge distribution is assumed to be uniform. The analysis uses the terms defined for the analysis of FMM in Chapter 2.

- Step 1: $(N/k_0)\log(N/k_0)$ flops
- Step 2a: $N(p+1)^2$ mult-coeff units
- Step 2b: $\frac{8N}{7k_0}\ (p+1)^4$ mult-mult conv units
- Steps 3a, 3b, 4: 430 $\frac{8N}{7k_0}$ flops
- Step 5a: 189 $\frac{8N}{7K_0}\ (p+1)^4$ mult-loc conv units
- Step 5b: $\frac{8N}{7K_0}\ (p+1)^4$ loc-loc conv units
- Step 5c: $\frac{27}{2}\ Nk_0$ flops
- Step 5d: $N(p+1)^2$ loc-coeff flops
- Step 6a, 6b: $2N$ integer operations

In this approach, a *mult-coeff* unit equals 18 flops, *mult-mult conv* unit equals 10 flops, *mult-loc conv* unit equals 15 flops, *loc-loc conv* unit equals 10 flops, and *loc-coeff* unit equals 18 flops. Thus, the computational complexity can be approximated by $15 \times 189\ (p+1)^4\ N$ flops, whereas the corresponding term in FMM is $44 \times 189\ (p+1)^4\ N$ flops as shown earlier.

Thus, the complexity of the novel approach is similar in terms of order, but the constant in the order term is about three times lesser in this approach as compared to FMM. However, it should be noted that this improvement in the constant is only in the

computation intensive steps and the rest of the steps take the same amount of time as FMM.

The storage requirement of the proposed approach are discussed here. A node unit is memory sufficient to hold a node of the tree, which is about 128 bytes, and a complex unit is a memory large enough to hold two double precision units, which is 16 bytes.

- Step 1: $\frac{8N}{k_0}$ node units = 128 $(\frac{8N}{k_0})$ bytes,
- Steps 2a, 2b: $\frac{8N}{7k_0}\ (p+1)^2$ complex units = 16 $(\frac{8N}{7k_0}\ (p+1)^2)$ bytes,
- Steps 3a, 3b, 4: 215 $\frac{8N}{7k_0}$ single precision units = 4 $(215\ \frac{8N}{7k_0})$ bytes,
- Steps 5a, 5b, 5c, 5d: $\frac{8N}{7K_0}\ (p+1)^2$ complex units = 16 $(\frac{8N}{7K_0}\ (p+1)^2)$ bytes,
- Steps 6a, 6b: $\frac{8N}{7K_0}$ double precision units = 8 $(\frac{8N}{7K_0})$ bytes.

Thus, the algorithm approximately requires a storage of $(32(p+1)^2+860)N$ bytes, which is $860N$ bytes lesser than that of FMM. For small values of p, this is about a 50% reduction in storage.

E. Error Analysis

The potential function is expressed as a weighted sum of multipole or local spherical harmonic basis functions, with the corresponding weights being multipole or local coefficients, respectively. The potential function belongs to the space of multivariate functions, which is an infinite dimensional space. Thus, in order to express the potential exactly, we need infinite spherical harmonic basis functions. However, the potential function can be expressed up to arbitrary accuracy by a finite set of spherical harmonic basis functions. The accuracy is governed by the degree, p, of the corresponding multipole or local spherical harmonic functions. The number of spherical harmonic functions of degree p used to express the potential is $(p+1)^2$. Thus, the corresponding number of weights or coefficients is also $(p+1)^2$.

An error is incurred in expressing the potential by a finite set of spherical harmonic basis functions. This error is basically the weighted sum of those spherical harmonic basis functions, which were not considered in the finite set used to express the potential.

The potential at any charge location consists of partial potentials due to

- direct interactions
- local spherical harmonic basis functions
- multipole spherical harmonic basis functions

The partial potential due to direct interactions is evaluated exactly by direct particle-particle interactions. The only errors incurred are in the partial potentials due to the local and multipole spherical harmonics corresponding to the charge location. The error incurred by the partial potential due to a set of multipole spherical harmonics is similar to the corresponding error incurred by the multipole expansions in the Fast Multipole Algorithm, given by (2.8). Similarly, the error incurred by the partial potential due to a set of local spherical harmonics is also similar to the corresponding error incurred by the local expansions in the Fast Multipole Algorithm, given by (2.12).

Here is a description of the error in potential due to multipole coefficients. Let $P(r,\theta,\phi)$ be the evaluation point, and $q_j(\rho_j,\alpha_j,\beta_j),\ j = 1 \ldots N$ be a set of charges, enclosed in a sphere of radius $r_0 < r$. p is the degree of spherical harmonics and N, the number of charges creating the potential. The results presented here mostly follow from the error results in FMM [4, 5].

The exact potential is expressed as

$$\Phi(P) = \sum_{j=1}^{N} \sum_{n=0}^{\infty} \sum_{m=-n}^{n} q_j \left(Y_n^{-m}(\alpha_j,\beta_j)\rho_j^n\right) \left(\frac{Y_n^m(\theta,\phi)}{r^{n+1}}\right). \tag{3.40}$$

For a degree p of spherical harmonics, the potential is approximated as

$$\Phi(\hat{P}) = \sum_{j=1}^{N} \sum_{n=0}^{p} \sum_{m=-n}^{n} q_j \left(Y_n^{-m}(\alpha_j, \beta_j)\rho_j^n\right) \left(\frac{Y_n^m(\theta,\phi)}{r^{n+1}}\right). \tag{3.41}$$

Thus, the error in approximating the potential is given by

$$E(P) = \Phi(P) - \Phi(\hat{P}) = \sum_{j=1}^{N} \sum_{n=p+1}^{\infty} \sum_{m=-n}^{n} q_j \left(Y_n^{-m}(\alpha_j, \beta_j)\rho_j^n\right) \left(\frac{Y_n^m(\theta,\phi)}{r^{n+1}}\right). \tag{3.42}$$

Simplification leads to,

$$E(P) = \sum_{j=1}^{N} q_j \sum_{n=p+1}^{\infty} \frac{\rho_j^n}{r^{n+1}} \left(\sum_{m=-n}^{n} Y_n^{-m}(\alpha_j, \beta_j) Y_n^m(\theta,\phi)\right). \tag{3.43}$$

It is known that

$$P_n(\cos\,\gamma_j) = \sum_{m=-n}^{n} Y_n^{-m}(\alpha_j, \beta_j) Y_n^m(\theta,\phi), \tag{3.44}$$

where γ_j is the angle subtended between $P(r,\theta,\phi)$ and $q_j(\rho_j, \alpha_j, \beta_j)$. (See e.g. equation 3.35 in [4]) Substituting (3.44) in (3.43), we have

$$E(P) = \sum_{j=1}^{N} q_j \sum_{n=p+1}^{\infty} \frac{\rho_j^n}{r^{n+1}} P_n(\cos\,\gamma_j). \tag{3.45}$$

Thus, the absolute value of the error is given by

$$|E(P)| = \sum_{j=1}^{N} |q_j| \sum_{n=p+1}^{\infty} \left|\frac{\rho_j^n}{r^{n+1}} P_n(\cos\,\gamma_j)\right|. \tag{3.46}$$

As the Legendre function, $P_n(x)$ is always bounded by 1, we get

$$|E(P)| \leq \sum_{j=1}^{N} |q_j| \sum_{n=p+1}^{\infty} \left(\frac{\rho_j^n}{r^{n+1}}\right). \tag{3.47}$$

Incorporating $\rho_j < r_0$, we have

$$|E(P)| \leq \sum_{j=1}^{N} |q_j| \sum_{n=p+1}^{\infty} \left(\frac{r_0^n}{r^{n+1}}\right). \tag{3.48}$$

Further simplifying, we can express the above equation (3.48) as

$$|E(P)| \leq \frac{\sum_{j=1}^{N} |q_j|}{r - r_0} \left(\frac{r_0}{r}\right)^{p+1}. \tag{3.49}$$

Thus, the order of error for using multipole coefficients is given by

$$|E(P)| = O\left(\frac{r_0}{r}\right)^{n+1}. \tag{3.50}$$

In our implementation, we have $\left(\frac{r_0}{r}\right) < \frac{1}{3}$, thus

$$|E(P)| = O\left(3^{-(n+1)}\right). \tag{3.51}$$

The rest of the error analysis for potential evaluation, i.e., the error in for potential due to local coefficients, error in computing multipole coefficients of a parent node from that of its children, error in multipole to local coefficient conversion, and, error in accumulation of local coefficient from parent node to each of its child nodes, have a similar bound. Detailed derivations can be found in [4, 5]. The error here is similar since the implementation details are similar to those in FMM. Hence, the overall error for potential in our implementation is given by

$$|E(P)| = O\left(3^{-(n+1)}\right). \tag{3.52}$$

In general if we have $\frac{r_0}{r} < \theta_0 < 1$, for a constant parameter θ_0, the overall error in potential is given by

$$|E(P)| = O\left(\theta_0^{n+1}\right). \tag{3.53}$$

Thus the error is a function of and converges exponentially w.r.t. the degree p of the corresponding spherical harmonics.

CHAPTER IV

EXPERIMENTS

The serial algorithm based on the proposed approach for potential evaluation was run on Linux workstations with 1.5GHz Pentium 4 dual processors with $1GB$ main memory. The parallel implementation of the algorithms was run on the 11-node 32 processor (7 nodes with 64MB, 4 nodes with 256MB) IBM p690 supercomputing machines in the National Center for Supercomputing Applications (NCSA) at University of Illinois.

A. Implementation Details

The *object-oriented design methodology* was adopted for the implementation of this algorithm. Object-oriented design makes it natural to model the objects like nodes of a tree. The object-oriented code can be used to write concise, clear, code with encapsulation (hiding the implementation details from its functionality), low-coupling (minimal interdependencies across different classes of functions) and high-cohesion (each class or function does something logically specific, and does it well).

The programming language C++ is used owing to its speed, flexibility, portability, and widespread usage and understanding. The following abstract data types are created in the program:

- Treenode: Each node of the tree is an object of class Treenode.
- Charge: Each charge is an object of class Charge.
- Location: Each location in cartesian coordinate is an object of class Location.
- Location_sph: Each location in spherical polar coordinate is an object of class Location_sph.
- List: The adjacency list and the lists $l1$, $l2$, and $l4$ are objects of class List.

All data associated with an object and all functions that act on it are encapsulated into a single class.

Firstly as many objects of class *Charge* are created as the number of charges. The root of the oct-tree is created as an object of class *Treenode* and a message is passed to this object to construct the oct-tree recursively. This message invokes a member function in class Treenode, which creates the oct-tree as instructed. The lists $l1$, $l2$, and $l4$ of the each node are created, by passing a message to the root, which creates these lists recursively in a top-down tree traversal. The conversion matrices for multipole coefficient computation and local coefficient computation are precomputed in the main program.

Multipole coefficients are computed by sending a message to the root, which computes the multipole coefficients of all nodes recursively in a bottom-up manner. Local coefficients due to lists $l2$ and $l4$ are evaluated by sending a message to the root, and the local coefficients of all nodes are recursively computed. Local coefficients due to list $l3$ are computed in a similar manner. Local coefficients of a node due to nodes far enough from its parent are computed from its parent's local coefficients. This is accomplished by a recursive subroutine.

Finally, the potential at each charge location is computed by sending a message to the subtree (in the parallel part of the subtree) in which the charge is present.

1. Serial Experiments

For the experiments reported in Tables I and II, the performance of the system (running time) is observed as a function of the degree of spherical harmonics, p, for three different values of N, the number of charges in the system. The experiments in this section were run on the 1.5GHz Pentium 4 dual processor Linux workstation with $1GB$ of memory.

As can be observed in the Table I, the running time of the system increases linearly with N, the number of charges in the system, for any given degree of spherical harmonics. Thus the experiments validate the fact that the algorithm is of $O(n)$ complexity.

Table I. Execution time (in seconds) as a function of degree, p and number of charges, N.

Degree	N		
(p)	1000	10000	100000
1	0.19	3.99	54.0
2	0.30	6.43	87.0
4	0.61	12.90	175.3

2. Experiments for Error Analysis

The error is in infinity norm ($||\ ||_\infty$), which is defined as

$$||E||_\infty = max_{j=1}^{N} \left| \frac{\hat{\phi}_j - \phi_j}{\phi_j} \right|, \tag{4.1}$$

where E is the error vector, N is the number of charges, ϕ_j is the exact potential and $\hat{\phi}_j$, the calculated potential at the jth charge location.

The experiments illustrate the exponential convergence of error w.r.t. the degree. In Table II, the error characteristics are observed as a function of the degree of spherical harmonics, p for three different values of N, the number of charges. As observed in Table II, the error reduces exponentially with increasing multipole degree. Small degrees of spherical harmonics (2, 3, 4) result in fairly accurate results.

Table II. Percentage error as a function of degree, p.

Degree	N		
(p)	1000	10000	100000
1	2.03	3.11	3.04
2	2.9×10^{-1}	2.9×10^{-1}	3.7×10^{-1}
4	2.0×10^{-2}	3.0×10^{-2}	5.1×10^{-2}
7	3.0×10^{-4}	4.1×10^{-3}	6.9×10^{-3}
10	2.0×10^{-5}	7.7×10^{-4}	9.5×10^{-4}

B. Parallel Formulation

In order to improve the overall running time of the algorithm, a parallel version of the multipole algorithm is developed. The primary goal in parallelization is to divide the entire task into smaller independent sub-tasks which can be executed in parallel on different processors, thereby, reducing the work per processor, resulting in an reduced overall running time. Parallelization was performed by employing Open MP directives, and the parallel code was run on the IBM p690 machines at the National Center for Supercomputing Applications (NCSA) at University of Illinois. All the computation intensive modules of the proposed algorithm, namely, the tree construction, the multipole coefficient evaluation, local coefficient evaluation and potential computation are efficiently parallelized.

1. Tree Formation

The tree formation is done in two parts: a serial part, and a parallel part. In the serial part, we start from the root box which encloses all charges, and subdivide it till each leaf has atmost a constant number of charges in it. At this stage only the serial part of the tree is formed and in the final tree, the leaves formed in the serial part of the tree will be replaced by sub-trees rooted at those leaves.

In the parallel part, we partition the leaves (formed at the serial part of the tree) equally among the processors, each of which will form a subtree by recursively subdividing the node. This part of the tree constructed in parallel will be referred as the parallel part of the tree.

Herein, we have a concurrent parallel execution followed by a single serial execution. In the parallel execution, the multipole vector of the nodes in the parallel part of the tree are evaluated in parallel in a bottom-up manner. After the parallel execution, the multipole vectors of the roots of each of the sub-trees (in the parallel part of the tree) are available.

The roots of the sub-trees are leaves of the serial tree. Now, the multipole vectors of all the other nodes, i.e., the nodes in the serial part of the tree are evaluated serially in a bottom-up manner.

The local coefficients of the nodes in the serial part of the tree are first computed. The leaves of the serial part of the tree, which are the roots of the sub-trees in the parallel part of the tree are divided equally among the processors and the local vectors of the nodes in parallel part of the tree are computed in parallel.

The local coefficients of the serial part of the tree are first accumulated in a top-down manner, followed by the accumulation of the nodes in the parallel part of the tree, again, in a top-down manner. The sub-trees in the parallel part of the tree are divided equally among all the processors.

Potential computation at the charge locations is done in parallel, by dividing the sub-trees in the parallel part of the tree equally among all the processors, each of which computes potential of charges in its subtrees.

2. Parallel Experiments

In each of the tables below, the number of charges is kept constant and the parallel performance of the system is observed as a function the number of parallel processors. The parallel performance is also studied as a function of the degree of spherical harmonics p.

The speedup, as shown in Table III, is linear for number of processors, $p_r \leq 4$, after which the speedup becomes slightly sub-linear for $p_r = 8$ and even more sub-linear for $p_r = 16$. This reduction in efficiency can be attributed to increased communication for more number of parallel processors, p_r.

As can be seen in Table IV, the parallel efficiency decreases as we go from degree, $p = 1$ to degree, $p = 2$, and increases from then on for higher degrees. The initial reduction in efficiency can be attributed to increased communication and the general increase in efficiency for higher degrees is due to higher computation to communication ratio.

Table III. Parallel efficiency for a degree, $p = 2$.

Processors (p_r)	Number of charges, N					
	10000			100000		
	Time (s)	Speedup	Eff. (%)	Time (s)	Speedup	Eff. (%)
1	4.99	1.00	100.0	65.26	1.00	100.0
2	2.47	2.02	101.0	33.69	1.94	96.9
4	1.26	3.97	99.1	16.91	3.86	96.5
8	0.66	7.57	94.7	8.84	7.39	92.3
16	0.41	12.26	76.7	4.84	13.5	84.4

Table IV. Parallel performance as a function of degree p, for $N = 100000$ charges.

Degree (p)	Number of Processors (p_r)							
	2		4		8		16	
	Time (s)	Eff. (%)	Time (s)	Eff. (%)	Time (s)	Eff. (%)	Time (s)	Eff. (%)
1	24.53	98.4	12.13	99.5	6.50	92.9	3.45	87.5
2	33.69	96.9	16.91	96.5	8.84	92.3	4.84	84.4
4	61.26	97.6	31.65	94.4	15.47	96.6	8.37	89.3

CHAPTER V

CONCLUSIONS

A simpler matrix-based hierarchical approach is developed for the potential evaluation problem. The simplicity of this approach makes it easier to comprehend. Optimization is accomplished in this novel approach by extracting redundancy in the problem, leading to precomputation of the conversion matrices and storing mostly the non-zero entries of these matrices. Further optimizations include relationships among the nodes in the various lists of each node and thereby reducing storage and computation cost. A strategy of constructing the top part of the tree first and then constructing the rest of the tree in parallel facilitates a logical and efficient parallelization of the algorithm at every cost-intensive step, thereby a near linear speedup is achieved. The algorithm is scalable and high efficiency is achieved for increasing problem sizes and degree of parallelization. Efficient object-oriented design is implemented for the potential evaluation problem as this design provides a simple top-down approach for understanding and solving the potential evaluation problem.

REFERENCES

[1] A. W. Appel, *An Investigation of Galaxy Clustering Using an Asymptotically Fast N-Body Algorithm*, Senior Thesis, Princeton University, Princeton, New Jersey, April 1981.

[2] J. Barnes and P. Hut, *A hierarchical $O(n \log n)$ force calculation algorithm*, Nature, 324 (1986), pp. 446–449.

[3] M. B. Buhmann, *Radial basis functions*, in Acta Numerica, Cambridge University Press, Cambridge, United Kingdom, 2000, pp. 1–38.

[4] L. Greengard, *The Rapid Evaluation of Potential Fields in Particle Systems*, Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, 1988.

[5] L. Greengard and V. Rokhlin, *A new version of the fast multipole method for the Laplace equation in three dimensions*, in Acta Numerica, Cambridge University Press, Cambridge, United Kingdom, 1997, pp. 229–269.

VITA

Name:	Sreekanth Juttu
Permanent Address:	c/o Dr. Vivek Sarin
	Department of Computer Science
	Texas A&M University
	College Station, Texas 77843-3112
Education:	M.S., Computer Science, Texas A&M University, 2003
	M.S., Mathematics and Scientific Computing,
	Indian Institute of Technology Kanpur, India, 2001