8th SIAM Conference on Parallel Processing for Scientific Computing, Minneapolis, MN, March 1997.

On the Accuracy of Anderson's Fast N–body Algorithm*

Yu Charlie Hu† S. Lennart Johnsson‡

Abstract

We present an empirical study of the accuracy–cost tradeoffs of Anderson's method. The various parameters that control the degree of approximation of the computational elements and the separateness of interacting computational elements govern both the arithmetic complexity and the accuracy of the method. Our experiment shows that for a given error requirement, using a near–field containing only nearest neighbor boxes and a hierarchy depth that minimizes the number of arithmetic operations minimizes the total number of arithmetic operations.

1 Introduction

Although $O(N)$ N–body methods [1, 4, 5, 13, 14] are superior to the direct $O(N^2)$ methods in terms of arithmetic complexity for large scale simulations, their complex computational structures relative to the direct method and consequent difficulty in achieving efficient computer implementations have so far hindered them from gaining wide acceptance in the application community. One other important reason for the slow acceptance of the $O(N)$ methods by the application community is the limited understanding of the error behavior of these methods on large–scale simulations, especially of the accuracy–cost tradeoffs.

Like the Multi–grid methods [2], the multipole–based N–body methods and their variations are approximation methods. For these methods, higher accuracy can be achieved at increased computational effort. Though the computational time is directly proportional to the number of particles, the constant of proportionality depends upon the desired accuracy. In Anderson's method there are four potential sources of error. The choice of these parameters affect the arithmetic complexity of the methods directly or indirectly.

Degree of approximation The degree of approximation of the aggregated effects of clusters of particles represented by computational elements determines the error decay rate. Greengard and Rokhlin [5, 4], and Zhao [14] give error bounds on truncated multipole expansions used to represent computational elements. Similarly, Anderson [1] provides some insights into the error characteristics of his method, but no rigorous tight bounds. Clearly, the degree of approximation directly affects the arithmetic complexity of translating the computational elements, and therefore that of the complete methods.

Degree of separation The separateness of interacting computational elements determines the base constant of the exponentially growing error bound. In the complete method, the separateness is determined by the degree of separation of the near–field and the use of supernodes.

*Supported by the Air Force Office of Scientific Research through grants F49620-93-1-0480 and F49620-96-1-0289.

†Aiken Computation Laboratory, Harvard University, Cambridge, MA 02138.

‡Aiken Computation Laboratory, Harvard University, Cambridge, MA 02138 and University of Houston, Houston, TX 77204.

TABLE 1

Arithmetic complexities at the five stages of $O(N)$ hierarchical methods, where $f(p)$ and $g(p)$ are the translation cost and the number of coefficients for computational elements with error decay rate p, and N_{int} and N_{near} are the numbers of interactive–field boxes and near–field boxes for interior nodes, respectively. Without using supernodes, $N_{near} = (2d+1)^3$ and $N_{int} = 7(2d+1)^3$ for d–separation near–fields in three dimensions.

Stage	Arithmetic complexity
Init–potential	$O(g(p) \cdot N)$
Upward pass	$O(f(p) \cdot 8^h)$
Downward pass	$O((N_{int} \cdot f(p) + f(p)) \cdot 8^h)$
Far–field	$O(g(p) \cdot N)$
Near–field	$O(N_{near} \cdot N^2/8^h)$
Minimized total	$O((g(p) + \sqrt{N_{near} \cdot N_{int} \cdot f(p)}) \cdot N)$

The use of supernodes [14, 6] reduces the effective value of the number of computational elements, “boxes”, in the *interactive–field*. This field is defined as the part of the far–field that is in the parent box's near–field. In [14], Zhao observes that for a near–field extending two boxes away in each direction from a box, *two–separation*, in three dimensions there are many groups of eight sibling nodes of common parents among the 567 interactive–field boxes of a destination box for the far–field to local–field conversion. Zhao suggests converting the far–field of the parent node instead of the far–field of each of the sibling nodes whenever all eight sibling nodes belong to the interactive–field. In three dimensions, 61 parent nodes have all their children in the interactive–field Hence, using the concept of supernodes reduces the number of far–field to local–field conversions per destination node from 567 to 140. In Greengard–Rokhlin's fast multipole method in three dimensions, using supernodes reduces the number of interactive–field interactions per box from 875 to 189. The same reduction in computational effort holds for Anderson's method in three dimensions.

Hierarchy depth Although varying the hierarchy depth does not change the separateness of interacting computational elements, it changes the portion of the particles in the near–field and far–field. Since the contribution from the particles in the two fields are computed differently, the overall error may differ with the hierarchy depth. For N uniformly distributed particles and a hierarchy of depth h having 8^h leaf–level boxes, the number of floating–point operations (FLOPs) required for the five stages of the generic hierarchical method as described by Katzenelson [9] is listed in Table 1. The *optimal hierarchy depth* minimizes the total number of FLOPs.

Round–off error Finite machine precision leads to round–off error. The evaluation of near–field potential/force via the direct method is subject to the accumulated round–off error, while the error in the evaluation of the far–field potential/force via hierarchy traversal comes from both the approximation method itself and the round–off error from finite machine precision. For a fixed degree of separation of the near–field, varying the hierarchy depth changes the proportion of particles in the near–field and in the far–field, and therefore may affect the error of the complete method. The error bounds given in [5, 4, 14] for multipole–based methods apply to individual computational elements. The error for the complete methods remain to be verified.

The main contribution of this paper is a scheme for choosing parameters of a nonadaptive version of Anderson's $O(N)$ N–body method, so that the code performs potential or force evaluation in the shortest running time for a prescribed error requirement.

TABLE 2
Some previous implementations of $O(N)$ N–body methods.

Author	Method	Degree of separation	Use of supernodes	Hierarchy depth
Greengard–Rokhlin [5]	GR	2	No	$\lfloor \log_8^N \rfloor$
Zhao [14]	Zhao	2	Yes	$\lfloor \log_8^N \rfloor$
Schmidt–Lee [11]	GR	1	No	3,4,5
Shimada et al. [12]	GR	1,2	No	$\log_8^N$
Leathrum [10]	GR	1,2,mixed 1 and 2	Yes and No	$\lfloor \log_8^N \rfloor$
Esselink [3]	GR	2	Yes	2,3,4

The scheme is developed and verified for uniform distributions of particles. The impact on the error of non–uniform particle distributions is studied through some model distributions. Our study shows that defining the near–field to contain only nearest neighbor boxes, *one–separation*, and using a hierarchy depth that minimizes the number of arithmetic operations for one–separation yields the shortest running time for a given error.

We also study the impact on the accuracy of Anderson's method by varying the radii of the outer and inner sphere approximations. We numerically verify the optimum choices of the radii for uniform particle distributions, and investigate the impact of varying sphere radii on the error for some model non–uniform distributions. Our experiments show that clustering the particles in the center of the leaf–level boxes yields a slight improvement in the error compared to a uniform distribution of particles in the leaf–level boxes. However, clustering particles at other locations in the leaf–level boxes may increase the error significantly. Enlarging the outer spheres, which intuitively will smooth out the integrations, improves the accuracy. Due to the page limitation, the experimental results for this study are omitted in this paper, but can be found in [8].

Previous work Previously reported work on implementations of $O(N)$ N–body methods, as listed in Table 2, have used a variety of parameters (hierarchy depth, degree of separation of the near–field, and the error decay rate). The issue of how to choose the parameters in order to minimize the running time for a prescribed accuracy remains unresolved.

2 Anderson's method

In Anderson's method [1], Poisson's formula is used for representing solutions of the underlying Laplace equation. Given a numerical integration formula, the discretized integration becomes a summation of expansions with Legendre functions as the base functions and relies only on the potential values at the integration points. The far–field and local–field potentials are called *outer–sphere approximation* and *inner–sphere approximation*, respectively. The three translation operations used in $O(N)$ methods, namely, shifting far–field potential T_{F2F}, converting far–field to local–field potential T_{F2L}, and shifting local–field potential T_{L2L} become evaluations of the outer–sphere and inner–sphere approximations of the source boxes at the destination boxes of the translation, as shown in Figure 1.

The error decay rate of Anderson's method is determined by the integration order of the sphere approximations, which in turn determines the choices of other parameters associated with the integration. Table 3 [1] lists the relationship among the parameters and the expected error decay rate of Anderson's method.

The fact that translations in Anderson's method are evaluations of sphere approximations leads to a different way of using supernodes. The error in evaluating an outer–sphere approximation with the expansion series truncated after M terms is $O(\frac{a}{r})^{M+2}$, where a is

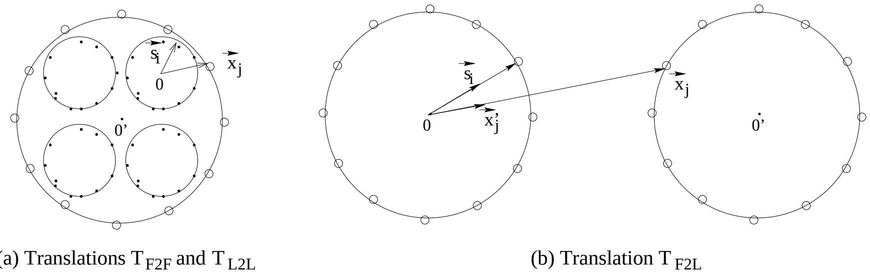

FIG. 1. *Translations (evaluations) of the approximations in Anderson's method.*

TABLE 3
Parameter selections and error decay rates of sphere approximations in Anderson's method.

Order of integration D	Number of integration points K	Expected error decay rate	Measured average error decay rate
5	12	4	3.95
7	24	5	5.68
9	32	6	8.85
11	50	7	7.32
14	72	9	8.90

the radius of the sphere containing the particles and r is the distance of the evaluation point from the center of the sphere [1]. Therefore, it is desirable to perform the conversion from a small sphere to a large sphere. Specifically, if a box in the interactive–field of a destination box is common to the interactive–fields of all sibling boxes of the destination box, then the far–field to local–field conversion is made with the parent box as a target instead of each of the eight destination sibling boxes, as shown in Figure 2. The converted local–field is added to the local–field of the parent. When all such conversions have been made, the local–field is passed to each of the children through the eight translation operations T_{L2L}, and the local–field conversions from the remaining interactive–field boxes are added. This way of using supernodes is dual to the use of supernodes in multipole–based methods, and the reduction in N_{int} remains the same.

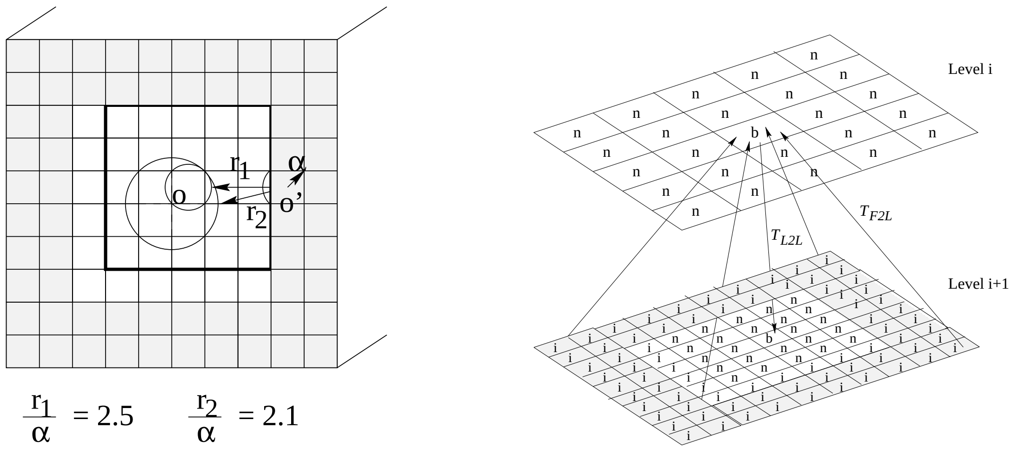

FIG. 2. *Supernodes in Anderson's method.*

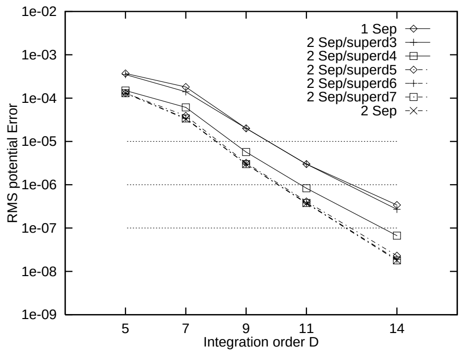

FIG. 3. *Delaying use of supernodes improves the accuracy, 1M particles with depth seven hierarchy on a 256 node CM-5E.*

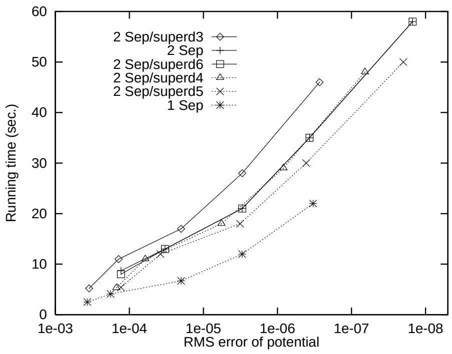

FIG. 4. *Running time vs. accuracy, 1M particles with optimal hierarchy depths on a 256 node CM–5E.*

3 Experiments

The empirical evaluations in this paper are performed on the CM–5E using our data–parallel implementation of the nonadaptive Anderson's method [7]. The accuracy is measured against an implementation of the direct $O(N^2)$ method on the CM–5E. Both codes perform floating–point operations in 64–bit precision. All the simulation results presented are collected on a 32–node CM–5E, unless otherwise specified. The default setups of the study are as follows:

- Uniform particle distribution: particles are distributed uniformly in a unit cubic domain, i.e., there is an equal number of particles in each leaf–level box, but the particle coordinates within the boxes are random.

- Uniform charges: particles are assigned equal charges of the same sign, and the total charge is 1, i.e. $q_i = 1.0/N, 1 \leq i \leq N$.

The root–mean–square (RMS) errors of the potential field are reported. The results for the force field, for random distribution of charges, and for small variations of uniform particle distributions can be found in [8].

3.1 Delaying use of supernodes improves the accuracy–cost tradeoffs

Our experiments show that during the downward traversal of the hierarchy, delaying the use of supernodes until further away from the root improves the accuracy of the code significantly, compared to using supernodes at all possible levels. Figure 3 shows that using supernodes starting at level four (with the root being level zero) improves the accuracy by up to a factor of five compared to starting applying supernodes at level three. Another factor of five is observed by delaying the use of supernodes until level five.

The reason that delaying the use of supernodes to levels four or five improve the accuracy of the method dramatically is that the contribution to the total potential from the interactive–fields from each level (except level two where the average number of interactive–field boxes is far less than normal) decreases exponentially away from the root of the hierarchy (see [8] for detail). For example, for the case above with 2–Sep, the contributions to the potential from interactive–fields at level two to seven account for 21%, 47%, 22%, 7%, 2%, and 0.5% of the total potential for a uniform distribution. Therefore, the contribution to the potentials from the interactive–fields at levels two, three, and four, carry large weights. Thus, their accuracies dominate the accuracy of the total potential. This explains the improvement of the accuracy offered by delaying the use of supernodes to levels four

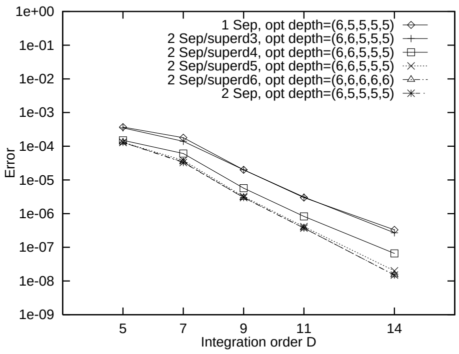

FIG. 5. *RMS error of the potential for 1M particles and optimal hierarchy depth for each of the five integration orders.*

and five, as seen in Figure 3.

Although delaying the use of supernodes in the downword traversal of the hierarchy improves the accuracy dramatically, it adds little to the total running time, compared with using supernodes at all the levels. Most of the computational effort for hierarchical methods occurs close to the leaf–level, while the contributions to the total potential at levels close to the root is far larger than that at levels close to the leaf–level. This suggests that for such charge distributions, one should choose the parameters such that the potential is evaluated with high accuracy at levels close to the root, e.g. by using a higher degree of approximation than for levels of the hierarchy close to the leaf–level, larger separateness for the interactive–fields, and not use supernodes.

Since 2–Sep/superd3 gives similar errors as 1–Sep while performing more arithmetic operations, and similarly, 2–Sep/superd5 gives similar errors as 2–Sep and 2–Sep/superd$i$ for $i \geq 5$, but requires fewer arithmetic operations, we only consider 1–Sep, 2–Sep/superd4, and 2–Sep/superd5 in the following. Figure 4 plots the cost–accuracy tradeoffs for the six methods using optimal hierarchy depth on a 256 node CM–5E. We see that 1–Sep and 2–Sep/superd5 are more cost efficient than the others, while 2–Sep/superd4 is more cost efficient than 2–Sep and 2–Sep/superd6 for low accuracy and about as efficient as the latter two for high accuracy.

3.2 Error models

Our experiments show that the accuracy is independent of the number of particles and the hierarchy depth. Figures 5 plots the RMS error of the potential field as a function of D. The measured error does not conform well with the analytic error bound $c_1 \cdot c^{-p}$. Since the accuracy is independent of the number of particles and the hierarchy depth, for each fixed pair of separation and use of supernodes, it suffice to model the error as a function of the integration order D. Using second–order least–square–fit (LSF), let

$$log_{10}(\epsilon) = C_1 \cdot D^2 + C_2 \cdot D + C_3 \qquad (1)$$

Fitted constants are found in Table 4.

Given an error ϵ, we can compute the required D using equation (1). Since the integration orders are discrete and take values in $\{5,7,9,11,14\}$, the computed D is rounded up to the nearest available integration order. Predicting the integration order D using equation (1) is almost always accurate, as verified with simulations of upto one million particles [8].

Table 4

Second–order LSF of RMS potential error as a function of the integration order (equation (1).)

Method	C_1	C_2	C_3	LSE $= \sum_i \epsilon_i^2$
1–Sep	-4.42e-03	-.274	-1.84	8.8e-02
2–Sep/superd4	-4.69e-03	-.301	-2.10	7.6e-02
2–Sep/superd5	-5.88e-03	-.325	-2.04	4.7e-02

3.3 Timing models

For a parallel implementation of a hierarchical method, the optimal hierarchy depth has to take into account the communication needs of the implementation and the communication efficiency of the underlying parallel machine. For our implementation of Anderson's method on the CM–5/5E, the FLOP count gives quite accurate predictions of the relative running times. First, the optimal hierarchy depth balances the two computation–dominant stages – the downward pass and the direct evaluation in the near–field (see Table 1.) As reported in [7], for problems with reasonably large sizes relative to the machine size, the two time-dominant stages achieve similar efficiency, and therefore the FLOP count predicts the optimal hierarchy depth well for each method. Second, the efficiencies for different separations and use of supernodes, e.g., 1–Sep, 2–Sep/superd4 and 2–Sep/superd5, are almost the same for the same hierarchy depth. Thus, the FLOP count for Anderson's method using optimal hierarchy depth can be used to predict well the relative running times for different degrees of separation and use of supernodes.

Experiments show that the predicted optimal hierarchy depths based solely on FLOP counts conform well with measured optimal depths [8]. For the few miss–predicted cases, the running time using the miss–predicted depth and the measured optimal depth differ by at most 20%.

3.4 Accuracy–cost tradeoffs for Anderson's method

Based on the error models and timing models developed above for the potential field of uniformly distributed particles, the selection of the degree of separation for the near–field, the use of supernodes, and the hierarchy depth that minimizes the running time for a given accuracy can be achieved in the following two steps:

1. For each combination of degree of separation and use of supernodes,
   (a) Find the required integration order D_i using equation (1). Look up the number of integration points K required.
   (b) Use the FLOP count to predict the optimal hierarchy depth, and calculate the total FLOP count at the predicted optimal hierarchy depth.
2. Choose the combination that has the smallest total FLOP count.

Using the above procedure, we compare 1–Sep with 2–Sep/superd4 and 2–Sep/superd5. Figures 6 – 7 plot the ratios of the FLOP counts and that of the running time of the three candidate methods as a function of the desired error. The two plots conform reasonably well, and suggest that the proposed parameter selection procedure works well.

A surprising outcome of this study is that 1–Sep is almost always more cost–efficient than 2–Sep/super$i$ for any prespecified error for uniform particle distributions. The tradeoff between the two alternatives is mainly determined by the cost–accuracy tradeoff of the integration formula used in the sphere approximations in Anderson's method. A detailed explanation can be found in [8].

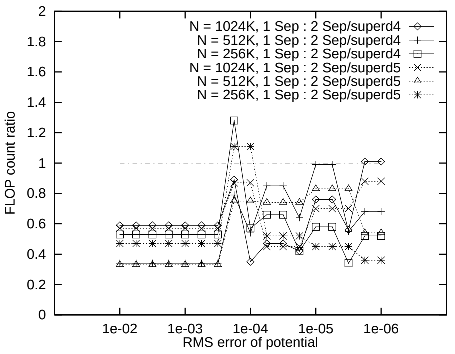

FIG. 6. *FLOP count ratio vs. error.*

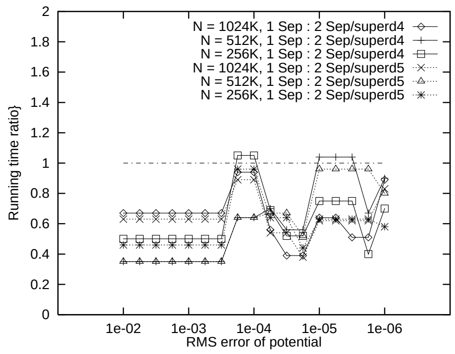

FIG. 7. *Running time ratio vs. error.*

4 Conclusions

Using our nonadaptive implementation of Anderson's method, we have studied its accuracy–cost tradeoffs. Our study shows that in three–dimensions, defining the near–field to encompass only nearest–neighbor boxes and using a hierarchy depth that minimizes the FLOP count almost always gives the shortest running time for a given accuracy

The study of accuracy–cost tradeoffs of $O(N)$ N–body methods in this paper focused on Anderson's method. We believe similar error and running time models can be developed to model the accuracy–cost tradeoffs of Greengard–Rokhlin's method and other methods as well. It is exceedingly valuable to develop a firm understanding of the accuracy–cost tradeoffs of the different hierarchical methods.

References

[1] C. R. Anderson. An implementation of the fast multipole method without multipoles. *SIAM J. Sci. Stat. Comput.*, 13(4):923–947, July 1992.

[2] A. Brandt. Multilevel computations: Review and recent developments. In S. McCormick, editor, *Proc. 3rd Copper Mountain Conference on Multigrid Method*, April 1987.

[3] K. Esselink. A comparison of algorithms for long range interactions. Technical Report 500.90.200GRA204-IN73916 AMER.94.006, Koninklijke/Shell-Lab, Amsterdam, August 1994.

[4] L. Greengard and W. D. Gropp. A parallel version of the fast multipole method. In *Parallel Processing for Scientific Computing*, pages 213–222. SIAM, Philadelphia, 1989.

[5] L. F. Greengard. *The rapid evaluation of potential fields in particle systems.* MIT Press, 1988.

[6] L. F. Greengard and V. Rokhlin. On the efficient implementation of the fast multipole algorithm. Technical Report YALEU/DCS/RR–602, Yale Univ., February 1988.

[7] Y. Hu and S. L. Johnsson. A data parallel implementation of hierarchical N–body methods. *Intl. J. of Supercomput. Appl. and High Perf. Comput.*, 10(1):3 – 40, 1996.

[8] ——–. On the accuracy of Anderson's fast N–body algorithm. Technical Report TR–06–96, Harvard University, Division of Applied Sciences, January 1996.

[9] J. Katzenelson. Computational structure of the N–body problem. *SIAM J. Sci. Statist. Comput.*, 10(4):787–815, 1989.

[10] J. F. Leathrum. *The parallel fast multipole method in three dimensions.* PhD thesis, Duke University, 1992.

[11] K. E. Schmidt and M. A. Lee. Implementing the fast multipole method in three dimensions. *J. Stat. Phy.*, 63(5/6), 1991.

[12] J. Shimada, H. Kaneko, and T. Takada. Performance of fast multipole methods for calculating electrostatic interactions in biomacromolecular simulations. *J. Comp. Chem.*, 15(1), 1994.

[13] M. S. Warren and J. K. Salmon. A portable parallel particle program. *Computer Physics Communications*, 87, 1995.

[14] F. Zhao. An $O(N)$ algorithm for 3–dimensional N–body simulations. Technical Report AI–TR–995, AI Lab, MIT, 1987.