Hierarchical Computation of 3-D Interconnect Capacitance using Direct Boundary Element Method*

Jiangchun Gu, Zeyi Wang and Xianlong Hong

Department of Computer Science and Technology
Tsinghua University, Beijing 100084, P. R. China
Tel: +86-10-62785564
Fax: +86-10-62781489
e-mail: {gujc, hongxl, wangzy}@tiger.cs.tsinghua.edu.cn

Abstract— The idea of Appel's hierarchical algorithm handling the many-body problem is implemented in the direct boundary element method (BEM) for computation of 3-D VLSI parasitic capacitance. Both the electric potential and normal electric field intensity on the boundary are involved, so it can be much easier to handle problems with multiple dielectrics and finite dielectric structure than the indirect BEM. Three kinds of boundaries (forced boundary, natural boundary and dielectric interface) are treated. Two integral kernels with different singularity ($1/r$, $1/r^3$) are involved while computing the interaction between the boundary elements. These features make it significantly distinct from the hierarchical algorithm based on the indirect BEM, which only handles single dielectric, one integral kernel and one forced boundary. The coefficient matrix is generated and stored hierarchically in this paper. As a result, computation cost of the matrix is reduced, and the matrix-vector multiplication in the GMRES iteration is accelerated, so computation speed is improved significantly.

Key Words: VLSI, Parasitic Capacitance, Direct Boundary Element Method, Hierarchical Computation

I. Introduction

In VLSI circuits, with the rapid increase of device density and working frequency, parasitic interconnect influence becomes the key factor affecting the circuit performance. Therefor, fast and accurate parasitic capacitance extraction is very important in designing the integrated circuits with high performance.

Compared with the finite difference method (FDM) and finite element method (FEM), the boundary element method (BEM) drew more attention in recent years due to its higher accuracy, fewer variables and strong ability to handle complex structure [1-9]. There are two kinds of the BEM, that is the indirect and direct BEM [10]. The indirect BEM solves single-layer potential integral equation with the density function as unknown variable. The equation contains only one integral kernel, which represents the electric potential generated by a unit charge located at the source point with a distance r. The kernel is $1/r$ in three dimensions and $\ln(1/r)$ in two dimensions. The direct BEM solves the direct boundary integral equation with the electric potential and normal electric field intensity as unknown variables. The direct BEM is convenient to handle problems with the multiple dielectrics and finite dielectric structures that are described by the mixed boundary condition. Both the indirect BEM [3-6] and direct BEM [2,7-9] are widely used in the extraction of the parasitic capacitance.

The coefficient matrix of the system of linear equations generated by BEM is dense and non-symmetric, which brings much cost to the solution of the system of equations. In 1985, Appel presented a hierarchical algorithm for the many-body simulation in astrophysics [11]. In the same year, Rokhlin presented the multipole accelerated algorithm for rapid solution of the boundary integral equation of 2-D classical potential theory [12], which was extended to 3-D situation in 1988 [13]. After that time, the hierarchical and multipole accelerated algorithms were applied to extraction of the 3-D parasitic capacitance. In 1991, Nabors and White applied the multipole accelerated algorithm to extraction of capacitance [3], and then extended to the problems with multiple dielectrics [4]. Last year, Appel's idea of the hierarchical algorithm was successfully applied to 3-D capacitance extraction [6]. Besides, another fast solution of the indirect boundary integral equation was proposed based on the singular value decomposition [5], in 1997. But, it should be pointed out that all of algorithms mentioned above were almost implemented in the indirect boundary integral equation with only one integral kernel and will encounter a lot of difficulties while extending them to the multiple dielectrics and finite domain problems.

In this paper, the hierarchical algorithm proposed by Appel [11] and Barnes [14] is implemented in the direct BEM, and the multiple and finite dielectric structure can be easily handled. Three kinds of boundaries (forced boundary, natural boundary and dielectric interface) are treated. Two integral kernels with different singularity ($1/r$, $1/r^3$) are involved while computing the interaction between the boundary elements. These features make it significantly distinct from the hierarchical algorithm presented by Shi et al in [6], which only handles single dielectric, one integral kernel and one forced boundary. The coefficient matrix is

* This work is supported by the National Science Foundation 69876024 and 973 National Project G1998030413.

generated and stored hierarchically. As a result, the computation cost of matrix is reduced, and the matrix-vector multiplication in the GMRES iteration is accelerated, so the computation speed is improved significantly.

II. DIRECT BEM COMPUTATION FOR CAPACITANCE

The parasitic capacitance can be obtained through the computation of the normal electric field intensity on the forced boundary (surfaces of conductors) [1,10]. In three-dimensional multiple dielectrics, the Laplacian's equation in domain $\Omega = \cup\Omega_k$ and its mixed boundary conditions are:

$$\begin{cases} \varepsilon_k \nabla^2 u = 0 & \text{In } \Omega_k\,(\mathrm{k}=1,\cdots,\mathrm{M}) \\ u = u_0 & \text{On } \Gamma_u \\ q = \partial u/\partial \mathbf{n} = q_0 = 0 & \text{On } \Gamma_q \\ \varepsilon_a \cdot \partial u/\partial \mathbf{n}_a = \varepsilon_b \cdot \partial u/\partial \mathbf{n}_a, & \\ u_a = u_b & \text{On } \Gamma_I \end{cases} \quad (1)$$

where u(x, y, z) is electric potential, boundary $\Gamma = \Gamma_u + \Gamma_q + \Gamma_I$. Γ_u is the forced boundary, where the electric potential is known, determined by bias voltage. Γ_q is the natural boundary (surfaces of dielectrics), where the normal electric field intensity $q{=}\partial u/\partial \mathbf{n}$ is zero, $\mathbf{n}$ is the unit out normal vector of the boundary. Γ_I is the interface between two adjacent dielectrics, where the electric potential and normal field intensity are unknown. ε_a、ε_b are the permittivity of the two dielectrics adjacent to the interface, $\mathbf{n}_a$ is the out normal vector of dielectric a. M is the number of dielectrics in the domain Ω.

With the fundamental solution u^* as weighting function, the Laplacian's equation (1) for one setting of the bias voltage is transformed into following direct boundary integral equation using the Green's formula [10]:

$$c_s u_s + \int_{\partial\Omega_k} q^* u d\Gamma = \int_{\partial\Omega_k} u^* q d\Gamma, \ (\mathrm{k}=1,\cdots,\mathrm{M}) \quad (2)$$

where u_s is the electric potential on source point s, c_s is a constant dependent on boundary geometry near point s. The fundamental solution of the Laplacian's equation is u^*=1/4πr, whose derivative along $\mathbf{n}$ is $q^*=\partial u^*/\partial \mathbf{n}$=-($\mathbf{r}$,$\mathbf{n}$)/4πr^3. The distance between the source point and the field point is r. $\partial\Omega_k$ is the boundary surrounding dielectric k. After discretizing the boundary $\partial\Omega_k$ into N_k boundary elements (elements on interface belong to two dielectrics), the discrete form of equation (2) is obtained:

$$c_i u_i + \sum_{j=1}^{N_k} \int_{\Gamma_j} q^* u d\Gamma = \sum_{j=1}^{N_k} \int_{\Gamma_j} u^* q d\Gamma,$$
$$(\mathrm{i}=1,\cdots,\mathrm{N_k}, \mathrm{k}=1,\ldots,\mathrm{M}) \quad (3)$$

where Γ_j represents the boundary element j on $\partial\Omega_k$. In equation (3), the integration is performed on field element Γ_j, and the source point is located on the source element Γ_i. When constant element is adopted, the number of unknown variables is N=N_1+⋯+N_M, and equation (4) can be obtained from equation (3):

$$\sum_{j=1}^{N} H_{ij} u_j = \sum_{j=1}^{N} G_{ij} q_j. \ (\mathrm{i}=1,\cdots,\mathrm{N}) \quad (4)$$

Substituting the boundary conditions listed in equation (1) into (4), a system of linear equations is obtained [7,10]:

$$\mathbf{Ax} = \mathbf{f}. \quad (5)$$

After solution of equation (5) using the generalized minimal residual (GMRES), the parasitic capacitance can be evaluated from the normal electric field intensity on the forced boundary.

III. HIERARCHICAL COMPUTATION OF DIRECT BEM

In the many-body simulation [11], the interaction between any two particles relies on one role – the gravitational attraction. Similarly, in solving the discrete indirect boundary integral equation, the interaction between any two panels (boundary elements) only relies on one integral kernel. But, it can be seen that the direct boundary integral equation (2) contains two integral kernels 1/r and 1/r^3 which have different singularity. This means that a panel acts on the boundary variables u and q of another panel through the two integral kernels. Because both physics variables u and q are involved in the direct integral equation, it can be much easier to handle the problems with the mixed boundary conditions such as multiple dielectrics and finite dielectric structures than the indirect BEM. On the other hand, it gives more difficulties to the hierarchical algorithms of the direct BEM.

Compared with the Shi's algorithm [6] implemented in the indirect BEM, three achievements have been made and will be presented in this paper. The first one is how to partition boundary elements on three different boundaries and organize them hierarchically. The second one is the hierarchical organization of the coefficient matrix for the structures with multiple boundaries. The third one is the hierarchical matrix-vector multiplication when multiple boundaries and integral kernels are involved.

A. Hierarchical Organization of Boundary Elements

In Shi's algorithm [6], the hierarchical partition of boundary element is performed during procedure of constructing the hierarchical coefficient matrix. But, it is difficult to extend the Shi's approach to the direct BEM with treatment of three different boundaries and two different integral kernels. In fact, the partition criterions for different kinks of boundaries and kernels should not be the same. So, in the algorithm presented here, the boundary element is partitioned prior to the hierarchical construction of the coefficient matrix.

Three different boundaries are classified as two kinds of type. The first one is the surfaces of conductors, and the other is the dielectric surfaces including the natural boundary surfaces and interfaces of dielectrics. Generally, shape of the conductor surfaces is simpler and most of them can be treated as a surface without holes. Here we suppose that only conductor surfaces without holes are discussed and the trapezoid is the most complicated shape we can meet. But, the dielectric boundaries that are touched by conductors should be treated as surfaces with some polygon holes. These dielectric boundaries with holes can be further treated as composition of smaller trapezoids by using the scan line method. Hence, both the conductor and dielectric boundary surfaces consist of the trapezoids which are called mother elements and need further partition of boundary element, generally. A surface with three holes is illustrated in Fig. 1. It was partitioned into nine mother elements.

For each mother element to be partitioned, the mesh number along the two directions is determined according to its type, position, and size etc. Then every mother element is partitioned recursively and organized as a complete binary tree. Each internal node of the tree is called a node element, and a leaf node is a leaf element. The node element always consists of its children node elements or leaf elements. The organization of complete binary tree is convenient for representing hierarchical relationship of different boundary elements. It is also convenient for the hierarchical construction of the coefficient matrix.

Besides geometry information, each element consists of some other items used in successive computation. RightChild and LeftChild point to its two children respectively. RightVector stores the right hand vector of the system of linear equations. Vector stores the operand vector for matrix-vector multiplication. Result stores the product of the matrix-vector multiplication. InterActionList stores the interaction coefficients with current element as source element.

The partition process of a mother element is illustrated in Fig. 2. S_{X0} and S_{X1} are children of S_X.

B. Construction of the Coefficient Matrix

After partitioning the boundary elements, the hierarchical algorithm processes each pair of mother elements recursively, and a hierarchical representation of the coefficient matrix is constructed during the process. The procedure is given below, the relevant data structure is omitted.

```
FormMatrix(Element * p, Element * q)
 {
   R = Distance(p, q);
   L1 = Size(p);    L2 = Size(q);
   if(L1/R < PEPS && L2/R <PEPS)
      FormInteraction(p, q);
   else
      if(L1 > L2 && p is not leaf element ||
         L1 ≤ L2 && q is leaf element)
        if(p is not leaf element)
         {
           if(the self interaction of p is not processed)
             FormMatrix (p->LeftChild, p->RightChild);
           FormMatrix(p->LeftChild, q);
           FormMatrix (p->RightChild, q);
         }
        else
           FormInteraction (p, q);
      else
        if(q is not leaf element)
         {
           if(the self interaction of q is not processed)
              FormMatrix (q->LeftChild, q->RightChild);
           FormMatrix (p, q->LeftChild);
           FormMatrix (p, q->RightChild);
         }
        else
           FormInteraction (p, q);
 }
```

The procedure FormMatrix starts from two mother elements. When both the two elements are leaf elements or the distance between them is far enough compared with their size, the recursion is terminated. The procedure FormInteraction computes the interaction coefficients between two elements and stores them into InterActionList of their corresponding source elements. The contribution to the right hand vector is calculated and accumulated into RightVector of the source element. The field element corresponding to the interaction coefficient is also recorded. During the successive matrix-vector multiplication, Vector in the field element is used to multiply this coefficient.

To fulfill the hierarchical matrix-vector multiplication, one-to-one relevancy between elements and unknown variables must be established. For the hierarchical algorithms based on indirect BEM [6], it is easy to do so because there is only one unknown variable in a element. But, for the direct BEM, there are two unknown variables (u and q) in one element located on dielectric interface. Considering that element on interface belongs to two adjacent dielectrics, let variable u relate with the element in one dielectric and q with the other one. With this arrangement, the interaction coefficients on matrix diagonal

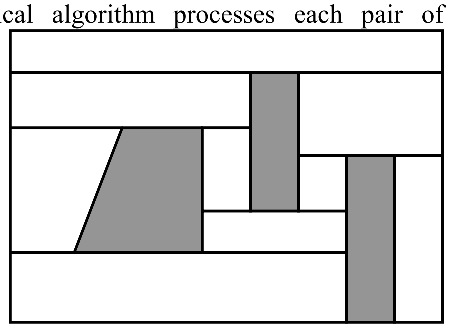

Fig. 1 Partitioning surface with three holes into nine mother elements.

are all generated from singular integration. The GMRES iteration can converge rapidly for the absolute values of the diagonal elements in the coefficient matrix are relatively larger.

The interaction coefficient matrix must be normalized to perform hierarchical matrix-vector multiplication. The normalization method is to divide each entry of the coefficient matrix by the area of its corresponding field element, and the right hand side remains unchanged. The normalization to the matrix is equivalent to multiply the solution by the area of its corresponding field element. So, the multiplication of electric field intensity with the area of its corresponding field element needs not to be done during evaluation of the capacitance.

After the procedure FormMarix is performed for each pair of mother elements, and the self-action of the leaf element is calculated, the coefficient matrix and the right hand side are generated. The hierarchical organization of three mother elements A, B and C is illustrated in Fig. 3. Mother element A belongs to dielectric 1, C belongs to dielectric 2, and B belongs to the interface between the dielectric 1 and dielectric 2. The hierarchical structure of the coefficient matrix concerned with the three mother elements is illustrated in Fig. 4. The source elements are on the left of the matrix. For element located on dielectric interface, the subscript denotes the dielectric it belongs to. The field elements are on the top. For element located on dielectric interface, the subscript denotes the variable it corresponds to.

C. Hierarchical Matrix-Vector Multiplication

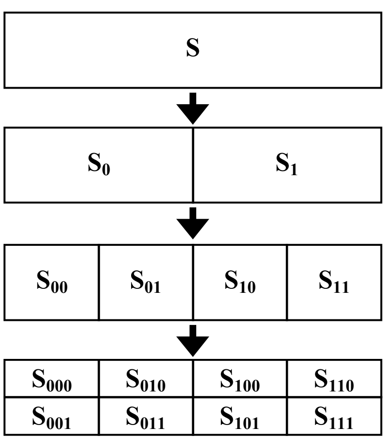

Fig. 2 Partition process of a mother element.

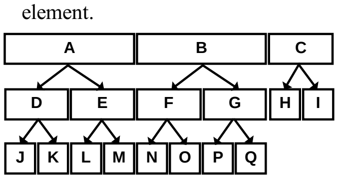

Fig. 3 Hierarchical organization of boundary elements on three mother elements A, B, C.

When the system of linear equations is solved by using the GMRES iteration, the essential operation in each iteration is a matrix-vector multiplication. Computation speed can be improved by exploiting the hierarchical matrix-vector multiplication algorithm. The multiplication proceeds in three steps.

1.Distribute Vector into Hierarchical Data Structure

The vector must be distributed into each element in the hierarchical structure before proceeding the multiplication. The values of leaf elements correspond to that in the vector. The value of a node element is the sum of its two child elements.

```
DistributeVector(Element * p, double * Vector)
{
   if(p is leaf element)
      p->Vector = Vector[Counter++];
   else
   {
      DistributeVector(p->LeftChild, Vector);
      DistributeVector(p->RightChild, Vector);
      p->Vector = p->LeftChild->Vector +
                  p->RightChild->Vector;
   }
}
```

2. Matrix-Vector Multiplication

The multiplication is performed on each element in the hierarchical data structure. For each element, the result is adding up the product of the coefficient and its corresponding term of the vector.

```
GenerateProduct(Element * p)
```

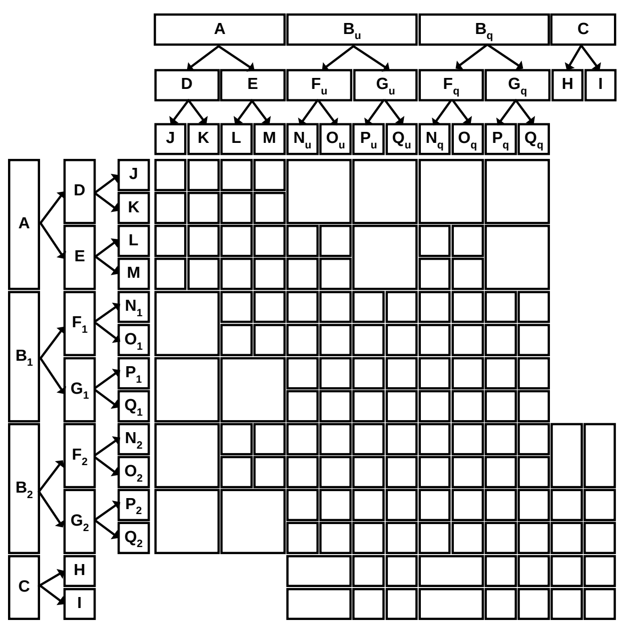

Fig. 4 Hierarchical organization of coefficient matrix concerned with the three mother elements A, B, C.

```
{
   p->Result = 0;
   for (each opInterAction in InterActionList of p)
      p->Result += opInterAction->Value *
                  opInterAction->Element->Vector;
   if(p is not leaf element)
   {
      GenerateProduct(p->LeftChild);
      GenerateProduct(p->RighChild);
   }
}
```

3. Collect the Product to the Vector

The product in each node element is added to its two children recursively, and the products in leaf elements are stored into the vector.

```
CollectResult(Element * p, double * Result)
{
   if(p is leaf element)
      Result[Counter++] = p->Result;
   else
   {
      p->LeftChild->Result += p->Result;
      p->RightChild->Result += p->Result;
      CollectResult(p->LeftChild, Result);
      CollectResult(p->RightChild, Result);
   }
}
```

The above procedures process one mother element at a time. After all the mother elements are processed by one procedure, successive procedure can start. One matrix-vector multiplication is finished when the three procedures performed. The global variable Counter should be initialized to 0 before procedures DistributeVector and CollectResult start.

IV. Numerical Results and Conclusion

To illustrate the efficiency of the hierarchical algorithm, the k×k (k = 2 ~ 6) bus crossing conductors embedded in two dielectrics are calculated using non-hierarchical and hierarchical algorithm, respectively. The substrate as a conductor is contained in the parasitic capacitors. The size of each bus is 1μm×1μm× (2k+1) μm. The space between two adjacent buses on the same layer is 1μm, and the space between two bus layers is 1μm. The distance between the outmost bus and the natural boundary of dielectric is 10μm in horizontal and 11μm in vertical direction. The height of the first dielectric layer is 13μm, and the other 12μm. A simulated structure with 6×6 cross buses is shown in Fig.5. The relative permittivity of the two dielectrics is set to 1. The threshold PEPS used in procedure FormMatrix is set to 1/2. The machine used in the experiments is Sun Ultra E450. The unit of time is second, and unit of capacitance is 10^{-18} farad. The interaction coefficient numbers of non-hierarchical and hierarchical algorithms are shown in table I. The accuracy and speed are shown in table II.

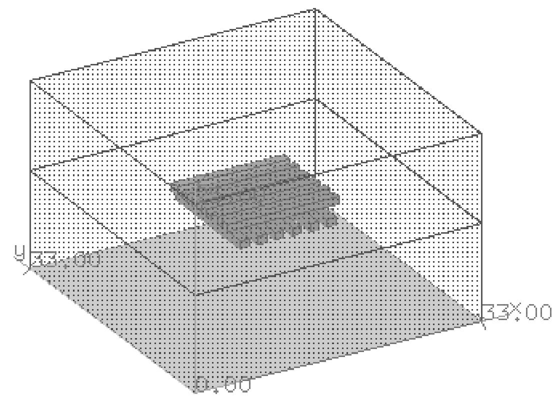

Fig. 5 A structure with 6×6 cross buses.

Data in table I show that the hierarchical algorithm generates much less interaction coefficients than the non-hierarchical one, so high speed up ratio can be expected.

As shown in table II, the hierarchical algorithm more than 20 times faster than the non-hierarchical algorithm while preserving high accuracy. It should be pointed out that the speed up ratio is relevant to the geometry of the parasitic capacitor being simulated. Higher speed up ratio can be obtained while computing parasitic capacitors containing large conductors.

TABLE I
Comparison between the Hierarchical and Non-hierarchical Computation in the Interaction Coefficient Number.

k×k Bus	Non-hierarchical algorithm	Hierarchical Algorithm	Ratio
2×2	15949376	425086	37.5
3×3	20459268	521484	39.2
4×4	33651472	649592	51.8
5×5	42684164	799472	53.4
6×6	52911360	964234	54.9

TABLE II
Comparison between the Hierarchical and Non-hierarchical Computation in Accuracy and Speed.

k×k	Non-hierarchical algorithm			Hierarchical algorithm				
Bus	Ele_Num	Capacitance	Time	Ele_Num	Capacitance	Error(%)	Time	Sp.
2×2	4036	238.6	154.1	4064	238.2	-0.2	6.7	23.0
3×3	4503	317.4	199.7	4432	317.4	0.0	8.2	24.4
4×4	5998	396.8	339.2	5824	396.4	-0.1	10.7	31.7
5×5	6729	474.6	436.1	6448	473.6	-0.2	13.5	32.3
6×6	7464	552.2	556.1	7072	551.2	-0.2	16.3	34.1

References

[1] S. Fukuda, N. Shigyo, K. Kate and S. Nakamara, “A ULSI 2-D Capacitance simulator for complex structures based on actual process,” *IEEE Trans. on CAD*, Vol. 9, No. 1, pp. 39-47, 1990.

[2] Zeyi Wang, Yanhong Yuan and Qiming Wu, “Parallel multipole accelerated 3-D capacitance simulator based on an improved model,” *IEEE Trans. on CAD*, Vol. 15, No. 12, pp. 1441-1550, 1996.

[3] K. Nabors and J. White, "FastCap: A multipole accelerated 3-D capacitance extraction program," *IEEE Trans. on CAD*, Vol. 10, No. 11, pp. 1447-1459, 1991.

[4] K. Nabors and J. White, “Multipole-Accelerated capacitance extraction algorithms for 3-D structures with multiple dielectrics,” *IEEE Trans. On CAS*, Vol.39, pp. 946-954, 1992.

[5] Sharad Kapur and David E. Long, “IES^3: A fast integral equation solver for efficient 3-Dimensional extraction,” *ICCAD’97*, pp. 448-455, 1997.

[6] Weiping Shi, Jianguo Liu, Naveen Kakani and Tiejun Yu, "A fast hierarchical algorithm for 3-D capacitance extraction," *Proc. DAC'98*, pp. 212-217, 1998.

[7] Qiming Wu and Zeyi Wang, “Application of the boundary element method in integrated circuit CAD,” *Computational Physics*, Vol. 9, No. 3, pp. 285-292, 1992 (in Chinese).

[8] E. A. Dengi and R. A. Rohrer, “Boundary element method macromodels for 2-D hierarchical capacitance extraction,” *35th DAC*, pp. 1-6, 1998.

[9] M. Bachtold, M. Emmenegger, J. G. Korvink and H. Baltes, “An error indicator and automatic adaptive meshing for electrostatic boundary element simulations,” *IEEE Trans. on CAD*, Vol. 16, No. 12, pp. 1439-1446, 1997.

[10] C. A. Brebbia, *The boundary element method for engineers*, London: Pentech Press, 1978.

[11] A. Appel, "An efficient program for many-body simulation," *SIAM J. Sci. and Stat. Comp.*, Vol. 6, No. 1, pp. 85-103, 1985.

[12] V. Rokhlin, “Rapid solution of integral equation of classical potential theory,” *J. Comput. Phys.*, Vol. 60, pp. 187-207, 1985.

[13] L. Greengard and V. Rokhlin, “The rapid evaluation of potential field in three dimensions,” *in Vortex Methods, Lecture Notes in Mathematics 1360*, C. Anderson and C. Greengard, Eds. New York: Springer-Verlag, 1988.

[14] J. Barnes and P. Hut, “ A hierarchical O(NlogN) force-Calculation algorithm,” *Nature*, Vol. 324, pp. 446-449, 1986.