A virtual 3-D fast extractor for interconnect capacitance of multiple dielectrics

Zhaozhi Yang*, Zeyi Wang

Department of Computer Science and Technology, Tsinghua University, 100084 Beijing, China

Received 22 March 2002; accepted 13 May 2002

Abstract

As features in VLSI reach submicron sizes, the parasitic interconnect must be calculated even more quickly and accurately. In this paper, we present a virtual 3-D extractor of multiple dielectrics that is very fast. In the indirect boundary integral equations from classical potential theory, we replaced the plane charge distribution on the surfaces of conductors with metal mesh charge distribution to simplify the 3-D structure. We also adopted the multipole acceleration with improved non-uniform cube partitioning to further simplify the computational complexity. Numerical results show that the computational complexity of our algorithm is about $O(n)$.

Keywords: Interconnect capacitance; Boundary element method; Line boundary element; Virtual three dimension; Multipole acceleration

1. Introduction

In VLSI circuits, because the feature size and the spacing between metal lines have decreased to the deep submicron level and multilevel metal layer technology is being used widely, interconnect parasitical effects have become key factors for determining signal delay, energy dissipation, and circuit reliability. In circuits with 5–6 metal layers and 0.25-μm feature size, interconnect delay accounts for about 50% of the total delay. Therefore, it is critical to access interconnect parameters quickly and accurately [1–3] in order to improve circuit performance.

To obtain simultaneous improvement of accuracy, speed and scale, several virtual 3-D capacitance extraction methods have been proposed [4–8]. They are fast but cannot handle complex structures. The method using mesh line charge distribution to replace charge distribution on the surfaces of

*Corresponding author.
E-mail address: yzz00@mails.tsinghua.edu.cn (Z. Yang).
conductors was first introduced in Refs. [9,10]. It preserves the 3-D frame of conductors, which results in the ability to handle complex structures, but because the 2-D surfaces of conductors are reduced to 1-D mesh lines, the calculations required are greatly reduced.

In Ref. [11], the approach of using mesh lines to simulate the surfaces of conductors [9,10] was adopted; here we improved that approach as follows. First, line boundary elements were used on the surfaces of the conductors, while plane boundary elements were used on the Neumann boundary, i.e. mixed boundary elements were introduced into the 3-D capacitance computation with single dielectric. Second, in the indirect boundary integral equations from the classical potential theory, Green’s function for free space was used instead of Green’s function only suitable for stratified media [9,10]; therefore, media with complex geometry could be handled. Finally, a multipole-accelerated algorithm [1–3] was introduced for higher computational efficiency.

This paper extends the method of Ref. [11] from a single dielectric to multiple dielectrics in 3-D capacitance extraction. Three contributions are presented. First, we use plane boundary elements on the interfaces between the dielectrics, and extend the idea of mixed boundary elements presented in Ref. [11]. Second, we deduced four types of analytic integral formulae for near field calculation, which greatly enhances performance of the multipole acceleration algorithm. Third, we give an improved non-uniform cube partitioning method for multipole acceleration, which results in higher speed and precision. Numerical results show that a prototype extraction tool ‘Lcap’ using the algorithms presented here can run several to tens of times faster than Fastcap2 [1,2] and with comparable accuracy.

The rest of the paper is divided into five sections. Section 2 describes interconnect capacitance extraction using mixed boundary elements. Section 3 gives integral formulae of the near field effect. Section 4 illustrates improvement of non-uniform cube partitioning using our methods, Section 5 shows some numerical results and comparisons and Section 6 gives conclusions.

2. Interconnect capacitance extraction using mixed boundary elements

Interconnect capacitance extraction can be represented as an electrostatics problem. In inhomogeneous media with relative permittivity \( \varepsilon(x) \), the electrostatic potential \( u(x) \) satisfies the following Laplace’s equation and boundary conditions:

\[
\begin{align*}
\nabla^2 u & = 0, & \text{in } \mathbb{R}^3 \backslash \Gamma_c \backslash \Gamma_i, \\
u & = u_0 & \text{on } \Gamma_c, \\
\varepsilon_+ \frac{\partial u}{\partial n} & = \varepsilon_- \frac{\partial u}{\partial n}, & u_+ = u_-, & \text{on } \Gamma_i,
\end{align*}
\]

(1)

where \( \Gamma_c \) denotes the union of conductor surfaces, \( u_0 \) denotes the given potential on the conductors, \( \Gamma_i \) is the union of the dielectric interfaces, \( n \) is the normal to \( \Gamma_i \), and \( u_\pm = u(x \pm 0n) \), \( \varepsilon_\pm = \varepsilon(x \pm 0n) \).

We use the indirect boundary element method, which converts the partial differential equation into a boundary integral equation, to solve the above problem. Given the conductor potential, charges on the surfaces of conductors can be computed using an equivalent charge formulation. In this formulation, surface charges are placed at conductor and dielectric interface surfaces, and the problem domain is replaced with free space. These surfaces charges produce the potential given by
\[
\begin{align*}
  u_0(x) &= \int_{\Gamma_+ + \Gamma_0} G(x, x') \sigma(x') \, d\Gamma_x, & x \in \Gamma_0, \\
  -\frac{1}{2\lambda} \sigma(x) &= \int_{\Gamma_+ + \Gamma_0} \frac{\partial}{\partial n} G(x, x') \sigma(x') \, d\Gamma_x, & x \in \Gamma_0,
\end{align*}
\]

where parameter \( \lambda \) is
\[
\lambda = \frac{\varepsilon_- - \varepsilon_+}{\varepsilon_- + \varepsilon_+}(x)
\]

where \( \sigma(x) \) denotes the charge density on the surfaces and \( G(x, x') \) is the Green’s function of the free-space:
\[
G(x, x') = \frac{1}{4\pi|x - x'|^\varepsilon_0}
\]

where \( |x - x'| \) is the Euclidean distance from \( x \) to \( x' \) and \( \varepsilon_0 \) is the permittivity of free space.

Solving Eq. (2), the free charges on the conductor \( j \) can be represented as
\[
Q_j = \int_{\Gamma_0} \sigma(x) \, d\Gamma_x
\]

Suppose that conductor \( i \) is set to be 1 V, while the others are 0 V; then the capacitance between conductor \( i \) and \( j \) \((i \neq j)\) is
\[
C_{ij} = -Q_j
\]

\( \Gamma_0 \) and \( \Gamma_i \), which are 2-D surfaces in the 3-D extraction, should be discretized when numerically solving Eq. (2). Generally, they are discretized into small patches to form plane boundary elements [1–3]. In Ref. [9], Green’s function suitable for the stratified media was used and only metal mesh lines instead of the conductor surfaces need to be discretized into line elements. Here for discretization of conductor surfaces \( \Gamma_i \) we adopt the method proposed by Ref. [9], as shown in Fig. 1. But, in consideration of differences in charge distribution, the plane boundary elements are used on the interfaces between adjacent dielectrics.

Our decision to replace charge distribution on a conductor surface with charge distribution on the metal mesh lines is based on the following three considerations. First, this approach adequately and conveniently represents the fact that the electrical charge on the surfaces of a conductor is mainly concentrated on its edges. The metal mesh lines serve as a Faraday’s cage that can shield internal space from the outside electrostatic field, just like the surfaces of a conductor. Numerical results have

![Fig. 1. The metal mesh lines on the surfaces of a conductor (black points are ends of elements).](image)
shown that sufficiently dense mesh lines can well describe the charge distribution on the surfaces of conductors. Second, adoption of the line distribution of charge reduces many of the boundary element integrals on surfaces from the 2-D to 1-D; this is the same as 1-D integral calculation in 2-D BEM, and greatly simplifies 3-D BEM computations. Third, the mesh lines on the surfaces of conductors can preserve the 3-D geometry frame, geometrical information and 3-D electrical effect of the interconnect wires, in detail. Hence, it can nicely overcome the shortcomings existing in many virtual methods that cannot adequately handle 3-D effects.

3. Integral formulae of the near field

We adopted constant elements for both the plane and line boundary elements. Because mixed boundary elements are used, there are four types of boundary element integrals after discretization. They are: the potentials of the line and plane charges, and electrical field intensities of the line and plane charges. We deduce the analytic formulae of all four integral types. Compared with the numerical integral, semi-analytic integral and divided-difference approaches [2,3], they are faster and more accurate.

Fig. 2 depicts the base model that calculates the potential and the electric field intensity for plane charges.

The potential of rectangle ABCD with plane charge density $\sigma$ at particle $P$ is

$$\psi_p = \int_{y_1}^{y_2} \int_{x_1}^{x_2} \frac{\sigma}{\sqrt{x^2 + y^2 + d^2}} dx$$

$$= \sigma \left[ y_2 \ln\left(\frac{K(x_2, y_2, d)/K(x_1, y_2, d)}{K(x_1, y_1, d)/K(x_2, y_1, d)}\right) + x_2 \ln\left(\frac{K(y_2, x_2, d)/K(y_2, x_1, d)}{K(y_1, x_1, d)/K(y_2, x_1, d)}\right) - 0.5d K \tan(x_2, y_2) + 0.5d K \tan(x_1, y_2) + 0.5d K \tan(x_1, y_1) - 0.5d K \tan(x_1, y_1) \right]$$

where

Fig. 2. Plane charge integral model.
\[ K(x, y, d) = x + \sqrt{x^2 + y^2 + d^2} \]  
(8)

\[ K \tan(x, y) = a \tan \left( \frac{2xyd\sqrt{x^2 + y^2 + d^2}}{d^2(x^2 + y^2 + d^2) - x^2y^2} \right) \]  
(9)

Rectangle ABCD is regarded as a plane boundary element, and point P is regarded as an evaluation point. In our method, if P is located only on the rectangle, the rectangle’s potential need not be considered.

If P is not on the rectangle, the gradients of the rectangle’s potential at point P according to the three coordinate axes are:

\[ \frac{\partial \psi_p}{\partial z} = \sigma \left[ y_2((K_2(x_2, y_2, d) - K_2(x_1, y_2, d)) + x_2((K_2(y_2, x_2, d) - K_2(y_1, x_2, d)) - y_1((K_2(x_2, y_1, d) - K_2(x_1, y_1, d)) - x_1((K_2(y_2, x_1, d) - K_2(y_1, x_1, d)) - 0.5K \tan(x_2, y_2, d) + 0.5K \tan(x_2, y_1, d) + 0.5K \tan(x_1, y_2, d) - 0.5K \tan(x_1, y_1, d) \right] \]  
(10)

\[ \frac{\partial \psi_p}{\partial x} = \sigma \int_{x_1}^{x_2} \int_{y_1}^{y_2} \frac{\partial}{\partial x} \frac{\sigma}{\sqrt{x^2 + y^2 + d^2}} \, dx \, dy \]  
= \sigma \ln \left( \frac{K(x_2, y_2, d)K(x_1, y_1, d)}{K(x_1, y_2, d)K(x_2, y_1, d)} \right)  
(11)

\[ \frac{\partial \psi_p}{\partial y} = \sigma \int_{x_1}^{x_2} \int_{y_1}^{y_2} \frac{\partial}{\partial y} \frac{\sigma}{\sqrt{x^2 + y^2 + d^2}} \, dx \, dy \]  
= \sigma \ln \left( \frac{K(x_2, y_2, d)K(x_1, y_1, d)}{K(x_1, y_2, d)K(x_2, y_1, d)} \right)  
(12)

where

\[ K_d(x, y, d) = \frac{d}{\sqrt{x^2 + y^2 + d^2(x + \sqrt{x^2 + y^2 + d^2})}} \]  
(13)

Function \( K() \) and function \( K \tan() \) are given by Eqs. (8) and (9). The electric field intensity along direction \( (a, b, c) \) at point P is:

\[ E_p(a, b, c) = -a \frac{\partial \psi_p}{\partial x} - b \frac{\partial \psi_p}{\partial y} - c \frac{\partial \psi_p}{\partial z} \]  
(14)

If P is just on the rectangle ABCD, the intensity is given by [2]:

\[ E_p = \frac{\sigma}{2\lambda} \]  
(15)
The integral formulae for line boundary elements are much simpler, and are specified in Appendix A.

4. The modified non-uniform cube partitioning

The advantage of replacing the surface of conductors by 1-D mesh lines is that many of the boundary element integrals are reduced from 2-D to 1-D and much of CPU time usually spent in constructing the linear system is saved. However, the discretized linear system has a dense nonsymmetrical coefficient matrix. So, even though GMRES [12] is used to solve it, the computational complexity is still \( O(n^2) \), where \( n \) is the number of boundary elements. In order to improve computational efficiency, we select and modify the multipole-accelerated algorithm based on a non-uniform cube partitioning [3].

The multipole-accelerated algorithm separates the calculation of the electric field into two parts, near field calculation and far field calculation. Commonly, the direct integral approach is used in the near field calculation, and two approximate computations, i.e. the multipole expansion and local expansion, are used in the far field calculation. The mechanism and implementation of the multipole accelerated algorithm are described in Refs. [1,3,13,14], in detail.

The multipole expansion and local expansion for rapid evaluation of the potentials and electrostatic fields are due to distributions of the charges. The multipole algorithms based on uniform cube partitioning [13] and non-uniform cube partitioning [14] are proposed by Greengard et al. to distinguish the far and near relationship between particles. Refs. [1,2] adopted the uniform cube partitioning and Ref. [3] adopted the non-uniform cube partitioning for the capacitance extraction. Results show that they are very efficient methods.

Our approach improves the non-uniform cube partitioning proposed by Refs. [14,3] for reducing the number of the empty cubes and near field calculations as follows. For simplicity, the 2-D box is used to illustrate how to improve the non-uniform cube partitioning.

We consider a domain depicted in Fig. 3a as a computational box of level 0. \( N \) particles are non-uniformly distributed in the computational box.

Fig. 3b shows the uniform box partitioning [13]. When particles are uniformly distributed, every box at the same level contains an approximately equal number of particles. However, in the capacitance extraction, particles are not uniformly distributed, and many boxes have no particles. This reduces computational efficiency [3].

Fig. 3c shows the situation of non-uniform box partitioning [3]. In this case, an integer \( s \geq 0 \) is fixed, and only those boxes that contain more than \( s \) particles are subdivided at every level of refinement. This strategy avoids a large number of empty boxes; however, it has two disadvantages. First, some empty boxes still exist, such as boxes A and B in Fig. 3c. Second, near field computation for some particles is much more than that of the uniform box partitioning. For example, for a particle \( a \) as shown in Fig. 3a, we can find, from Fig. 3c, that all the other particles belong to its near field and their effects on the particle \( a \) should be directly calculated. However, all the other particles are outside the near field of particle \( a \) and their effects on particle \( a \) can be accelerated by multipole acceleration with the uniform box partitioning shown in Fig. 3b.

To increase computational efficiency of the multipole accelerated algorithm, two improvements of the non-uniform box partitioning are implemented as follows:
1. In the process of subdividing computational boxes, we not only consider whether a box should be subdivided, but also eliminate those empty boxes generated after subdivision.

2. To decide whether to subdivide a box or not, we consider not only the number of particles in it, but also to which level each particle belongs. An integer $l_0 > 0$ is fixed. If a box’s level $l < l_0$, it must be subdivided, no matter how many particles it contains.

When $l_0$ is small, the modified non-uniform box partitioning give results close to non-uniform box partitioning. When $l_0$ is large, this method is close to uniform partitioning. This method eliminates both empty boxes and much of unnecessary direct computations. Fig. 3d shows modified non-uniform box partitioning. Those squares with dotted lines are not computational boxes. In Fig. 3d, we can see that the empty boxes A and B do not exist and all the influences of the other particles on the particle $a$ need not be directly calculated. Selecting $l_0$ properly, we can obtain higher computational performance.

5. Numerical results

A code Lcap (Line Capacitance) implementing the algorithm presented here was written in C++ language and run on a Sun Enterprise450 workstation.
5.1. Comparison with Fastcap2

In this section, Lcap is compared with the 3-D capacitance extractor Fastcap2 [1,2], which is well-known for its computational performance and high extraction speed.

Exx are called x × x bus-crossing problems, as are Exxx. The height and width of all the rectangular parallelepipeds are 1 μm. The spacing between any two adjacent metal buses is 1 μm. The spacing between the buses and the boundary of the calculating domain is 1 μm and that between metals and their adjacent dielectric interfaces is 0.5 μm.

In Table 1, capacitance values of Lcap and Fastcap2 match well. The largest difference in the six examples is 7%. We can see that Lcap is much faster than Fastcap2. Lcap obtained capacitance values for E555 in only 5.3 s with 13 511 panels, while Fastcap2 took 202 s with 6768 panels. Though using about twice as many boundary elements as Fastcap2, Lcap can run much faster than Fastcap2.

With increase in the simulated size, the acceleration rate compared with Fastcap2 increases. This feature is demonstrated in Fig. 4, where more examples and partitions are considered. Both Lcap and Fastcap2 have computational complexity of about $O(n)$, but Lcap is still faster.

Both Fastcap2 and Lcap build boundary integral equations through the indirect element method and adopt multipole acceleration. The main differences between them are as follows: Fastcap2 uses plane elements while Lcap uses mixed elements; Fastcap2 uses uniform cube partitioning while Lcap uses the improved non-uniform cube partitioning; and Fastcap2 uses the divided-difference approach to calculate electric field intensities while Lcap uses analytical integrals. These differences, especially the utilization of mixed elements, mean that Lcap requires much less CPU time.

<table>
<thead>
<tr>
<th></th>
<th>Var.</th>
<th>Cap. (10$^{-19}$ F)</th>
<th>Time (s)</th>
<th>Difference (%)</th>
<th>Rate$^a$</th>
</tr>
</thead>
<tbody>
<tr>
<td>E33</td>
<td>Lcap</td>
<td>4058</td>
<td>993.6</td>
<td>1.5</td>
<td>2.5</td>
</tr>
<tr>
<td></td>
<td>Fastcap2</td>
<td>2197</td>
<td>968.9</td>
<td></td>
<td>1.5</td>
</tr>
<tr>
<td>E44</td>
<td>Lcap</td>
<td>6714</td>
<td>1232.0</td>
<td>2.6</td>
<td>−3.3</td>
</tr>
<tr>
<td></td>
<td>Fastcap2</td>
<td>3891</td>
<td>1273.9</td>
<td>212</td>
<td>1.9</td>
</tr>
<tr>
<td>E55</td>
<td>Lcap</td>
<td>6330</td>
<td>1647.5</td>
<td>2.6</td>
<td>4.5</td>
</tr>
<tr>
<td></td>
<td>Fastcap2</td>
<td>3187</td>
<td>1572.0</td>
<td>26</td>
<td>1.9</td>
</tr>
<tr>
<td>E333</td>
<td>Lcap</td>
<td>3447</td>
<td>930.3</td>
<td>1.5</td>
<td>−2.9</td>
</tr>
<tr>
<td></td>
<td>Fastcap2</td>
<td>1794</td>
<td>958.2</td>
<td>47</td>
<td></td>
</tr>
<tr>
<td>E444</td>
<td>Lcap</td>
<td>3872</td>
<td>1233.8</td>
<td>1.9</td>
<td>−3.0</td>
</tr>
<tr>
<td></td>
<td>Fastcap2</td>
<td>2004</td>
<td>1272.0</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>E555</td>
<td>Lcap</td>
<td>13 511</td>
<td>1517.8</td>
<td>5.3</td>
<td>−7.0</td>
</tr>
<tr>
<td></td>
<td>Fastcap2</td>
<td>6768</td>
<td>1631.6</td>
<td>202</td>
<td></td>
</tr>
</tbody>
</table>

$^a$Rate = Fastcap2 time/Lcap time.
5.2. Results of interconnect capacitances

Table 2 is the comparison of a $3 \times 3$ bus-crossing case in a single medium, shown as Fig. 5. Because of the symmetry of interconnect capacitances, the capacitance matrix should be symmetric. The result of Fastcap2 is strictly symmetric, because it only calculates once for each coupling capacitance. The result of Lcap is approximately symmetric, because it calculates each couple separately. This shows its reliability at one aspect.

Most results of these two programs match well, except the coupling capacitance between conductors 4 and 6. Another case that shows a significant difference is the coupling capacitance between conductors 2 and 3. If we consider only the symmetry of the case, the coupling capacitance should be equal to that between conductors 1 and 2. Hence, the result of Lcap is more reasonable.

Table 2
Interconnect capacitances of a $3 \times 3$ bus-crossing case ($10^{-18}$ F)

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fastcap2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1074</td>
<td>-212.9</td>
<td>-17.44</td>
<td>-110.5</td>
<td>-116.2</td>
<td>-112.1</td>
</tr>
<tr>
<td>2</td>
<td>-212.9</td>
<td>1154</td>
<td>-188.6</td>
<td>-108.4</td>
<td>-114.8</td>
<td>-109.9</td>
</tr>
<tr>
<td>3</td>
<td>-17.44</td>
<td>-188.6</td>
<td>1064</td>
<td>-110.6</td>
<td>-118.5</td>
<td>-112.4</td>
</tr>
<tr>
<td>4</td>
<td>-110.5</td>
<td>-108.4</td>
<td>-110.6</td>
<td>924.1</td>
<td>-273.8</td>
<td>-41.94</td>
</tr>
<tr>
<td>5</td>
<td>-116.2</td>
<td>-114.8</td>
<td>-118.5</td>
<td>-273.8</td>
<td>1086</td>
<td>-255.8</td>
</tr>
<tr>
<td>6</td>
<td>-112.1</td>
<td>-109.9</td>
<td>-112.4</td>
<td>-41.94</td>
<td>-255.8</td>
<td>914.7</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lcap</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1038</td>
<td>-227.79</td>
<td>-17.487</td>
<td>-112.71</td>
<td>-115.74</td>
<td>-118.49</td>
</tr>
<tr>
<td>2</td>
<td>-220.88</td>
<td>1134.7</td>
<td>-238.27</td>
<td>-106.41</td>
<td>-107.72</td>
<td>-112.81</td>
</tr>
<tr>
<td>3</td>
<td>-15.373</td>
<td>-233.75</td>
<td>1038.4</td>
<td>-111.75</td>
<td>-114.32</td>
<td>-117.21</td>
</tr>
<tr>
<td>4</td>
<td>-107.66</td>
<td>-98.26</td>
<td>-106.29</td>
<td>918.72</td>
<td>-290.47</td>
<td>-7.4235</td>
</tr>
<tr>
<td>6</td>
<td>-109.69</td>
<td>-101.8</td>
<td>-108.33</td>
<td>-5.2367</td>
<td>-282.25</td>
<td>916.76</td>
</tr>
</tbody>
</table>
5.3. Results of modified non-uniform cube partitioning

In Table 3, we can see that the improved non-uniform cube partitioning enhances the calculation efficiency from 30 to 150%.

6. Conclusions

In the process of extracting, we use the indirect boundary method with mixed boundary elements to simplify the 3-D structures and adopt the multipole acceleration with the improved non-uniform cube partitioning to further simplify the computational complexity. This algorithm gives high processing speed.

<table>
<thead>
<tr>
<th>Panel</th>
<th>Non-uniform cube partitioning</th>
<th>Modified approach</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Cap. (10^{-18} F)</td>
<td>Time (s)</td>
</tr>
<tr>
<td>E33.1</td>
<td>3014</td>
<td>1017</td>
</tr>
<tr>
<td>E44.1</td>
<td>7014</td>
<td>1236</td>
</tr>
<tr>
<td>E55.1</td>
<td>6710</td>
<td>1809</td>
</tr>
<tr>
<td>E333.1</td>
<td>3667</td>
<td>1068</td>
</tr>
<tr>
<td>E444.1</td>
<td>6568</td>
<td>1479</td>
</tr>
<tr>
<td>E555.1</td>
<td>14 695</td>
<td>1702.5</td>
</tr>
</tbody>
</table>
Acknowledgements

The work was supported by the National Science Foundation of China Grant no. 69876024, and also partly supported by National Foundation for Key Base Research Grant no. G1998030404.

Appendix A. The integral formulae of line boundary elements

Fig. 6 is the base model that calculates the potential of line charges. The potential of line AB with line charge density $\sigma$ at point C, which is called particle, is

$$\psi_c = \int_{x_1}^{x_2} \frac{\sigma}{\sqrt{x^2 + y_0^2 + z_0^2}} \, dx = \sigma \ln\left(x + \sqrt{x^2 + y_0^2 + z_0^2}\right) \bigg|_{x_1}^{x_2}$$  \hfill (A.1)

We call line AB a line boundary element, and point C an evaluation point, which is always the center of a boundary element. If C is on the line, we adopt the Gaussian integral formula to avoid the singular integral.

As shown in Fig. 6, the gradients of its potential at point C according to three coordinate axes are:

$$\frac{\partial \psi_c}{\partial x} = \sigma \left(\frac{y_0}{\sqrt{x^2 + y_0^2 + z_0^2}} \right) \bigg|_{x_1}^{x_2}$$  \hfill (A.2)

$$\frac{\partial \psi_c}{\partial y} = \sigma \left(\frac{y_0}{\sqrt{x^2 + y_0^2 + z_0^2}} \right) \bigg|_{x_1}^{x_2}$$  \hfill (A.3)

$$\frac{\partial \psi_c}{\partial z} = \sigma \left(\frac{y_0}{\sqrt{x^2 + y_0^2 + z_0^2}} \right) \bigg|_{x_1}^{x_2}$$  \hfill (A.4)

hence, the electric field intensity along direction $(a, b, c)$ at point C is:

$$E_c(a, b, c) = -a \frac{\partial \psi_c}{\partial x} - b \frac{\partial \psi_c}{\partial y} - c \frac{\partial \psi_c}{\partial z}$$  \hfill (A.5)

Fig. 6. Line charges integral model.
Because line elements are distributed on the surfaces of conductors and the electric field intensity need not be calculated here, point C is not just on line AB. No singular integral will appear.

References