JOURNAL OF COMPUTATIONAL PHYSICS 67, 263–278 (1986)

Numerical Solution of Poisson's Equation with Arbitrarily Shaped Boundaries Using a Domain Decomposition and Overlapping Technique

KAZUYOSHI MIKI AND TOSHIYUKI TAKAGI

Energy Research Laboratory, Hitachi, Ltd., 1168 Moriyama-cho, Hitachi-shi, Ibaraki, 316 Japan

Received July 4, 1984; revised February 20, 1986

A direct-solution scheme for numerically solving the 3-dimensional Poisson's problem with arbitrarily shaped boundaries $\nabla \cdot (\lambda \nabla \phi) = S$ on Ω, $C_1\phi + C_2 \mathbf{n} \cdot (\lambda \nabla \phi) = C_3$ on $\partial\Omega$, has been developed by using a boundary-fitted coordinate transformation. The scheme also used the technique of decomposing the closed domain Ω into several hexahedron subdomains and then overlapping neighboring hexahedrons to deal with complicated geometries. A large system of linear equations derived from discretizing the Poisson's equation was solved by using a biconjugate gradient method with incomplete LU factorization of the nonsymmetric coefficient matrix as preconditioning. The convergence behavior of the different domain decompositions was demonstrated for a numerical experiment. Application to the electrostatic field problem in the electron gun of a color picture tube confirms that the present numerical scheme should provide an efficient and convenient tool for solving many important large-scale engineering problems.

1. INTRODUCTION

The numerical solution of the elliptic Poisson's equation is required for many important engineering problems including electrostatic field problems, thermal conduction problems, and flow problems for velocity potentials of pressure fields. Various numerical techniques have been developed; among these the finite difference method is the most straightforward and widely used. But this method needs some interpolation between grid points for complicated boundary shapes with strong curvature or slope discontinuities, which may introduce significant numerical errors. The finite element method, on the other hand, has geometrical advantages for matching complicated boundaries, but it requires considerable experience and time to divide the field into finite elements and concentrate the elements in some specified regions without using a preprocessor for grid generation.

Another approach to deal with geometrical complexities is the use of a boundary-fitted coordinate transformation technique [1]. This technique is based on an automated numerical generation of a curvilinear coordinate system having a coordinate line coincident with each boundary of the arbitrarily shaped domains. But when dealing with geometrically complicated 3-dimensional domains, the

0021-9991/86 $3.00

generation procedure becomes more complex and has less flexibility in adapting to a broad variety of geometries.

This paper presents a new direct-solution scheme for the 3-dimensional Poisson's equation with arbitrarily shaped boundaries. The scheme is based on the technique of decomposing the complicated closed domain into a set of overlapping hexahedron subdomains, each of which has six curved or plane surfaces. Each subdomain grid is generated independently by the boundary-fitted coordinate transformation technique [2]. It also uses an efficient solution technique for a large system of linear equations derived from discretizing the Poisson's equation. An outline of the solution scheme and some numerical results are described in the following sections.

2. METHOD

2.1. *Boundary-Fitted Coordinate Transformation*

The coordinate transformation technique used in the present study is based on the method developed by Mastin and Thompson [3]. The transformation from the physical space (x, y, z) to the transformed space (ξ, η, ζ) must be one of the solutions of the equations

$$\frac{\partial^2 \xi}{\partial x^2} + \frac{\partial^2 \xi}{\partial y^2} + \frac{\partial^2 \xi}{\partial z^2} = P, \tag{1a}$$

$$\frac{\partial^2 \eta}{\partial x^2} + \frac{\partial^2 \eta}{\partial y^2} + \frac{\partial^2 \eta}{\partial z^2} = Q, \tag{1b}$$

$$\frac{\partial^2 \zeta}{\partial x^2} + \frac{\partial^2 \zeta}{\partial y^2} + \frac{\partial^2 \zeta}{\partial z^2} = R, \tag{1c}$$

subject to Dirichlet boundary conditions. The source functions P, Q, and R are functions with control grid spacing.

Since all numerical computations are performed in the transformed space, the dependent and independent variables must be interchanged in Eqs. (1). This results in the elliptic system of quasilinear equations

$$\alpha_{11}\frac{\partial^2 x}{\partial \xi^2} + \alpha_{22}\frac{\partial^2 x}{\partial \eta^2} + \alpha_{33}\frac{\partial^2 x}{\partial \zeta^2} + 2\alpha_{12}\frac{\partial^2 x}{\partial \xi\, \partial \eta} + 2\alpha_{13}\frac{\partial^2 x}{\partial \xi\, \partial \zeta} + 2\alpha_{23}\frac{\partial^2 x}{\partial \eta\, \partial \zeta} + J^2\left(P\frac{\partial x}{\partial \xi} + Q\frac{\partial x}{\partial \eta} + R\frac{\partial x}{\partial \zeta}\right) = 0, \tag{2a}$$

$$\alpha_{11}\frac{\partial^2 y}{\partial \xi^2} + \alpha_{22}\frac{\partial^2 y}{\partial \eta^2} + \alpha_{33}\frac{\partial^2 y}{\partial \zeta^2} + 2\alpha_{12}\frac{\partial^2 y}{\partial \xi\, \partial \eta} + 2\alpha_{13}\frac{\partial^2 y}{\partial \xi\, \partial \zeta} + 2\alpha_{23}\frac{\partial^2 y}{\partial \eta\, \partial \zeta} + J^2\left(P\frac{\partial y}{\partial \xi} + Q\frac{\partial y}{\partial \eta} + R\frac{\partial y}{\partial \zeta}\right) = 0, \tag{2b}$$

$$\alpha_{11}\frac{\partial^2 z}{\partial \xi^2}+\alpha_{22}\frac{\partial^2 z}{\partial \eta^2}+\alpha_{33}\frac{\partial^2 z}{\partial \zeta^2}+2\alpha_{12}\frac{\partial^2 z}{\partial \xi\,\partial \eta}+2\alpha_{13}\frac{\partial^2 z}{\partial \xi\,\partial \zeta}+2\alpha_{23}\frac{\partial^2 z}{\partial \eta\,\partial \zeta}$$
$$+J^2\left(P\frac{\partial z}{\partial \xi}+Q\frac{\partial z}{\partial \eta}+R\frac{\partial z}{\partial \zeta}\right)=0. \tag{2c}$$

Coefficients α_{jk} are defined as

$$\alpha_{jk}=\sum_{m=1}^{3}\beta_{mj}\beta_{mk}, \tag{3}$$

where β_{mj} is the cofactor of the (m, j) elements in the matrix,

$$M=\begin{bmatrix}\dfrac{\partial x}{\partial \xi} & \dfrac{\partial x}{\partial \eta} & \dfrac{\partial x}{\partial \zeta}\\[2ex] \dfrac{\partial y}{\partial \xi} & \dfrac{\partial y}{\partial \eta} & \dfrac{\partial y}{\partial \zeta}\\[2ex] \dfrac{\partial z}{\partial \xi} & \dfrac{\partial z}{\partial \eta} & \dfrac{\partial z}{\partial \zeta}\end{bmatrix}. \tag{4}$$

And J denotes the Jacobian determinant of the inverse transformation,

$$J=\frac{\partial(x, y, z)}{\partial(\xi, \eta, \zeta)}=\det|M|. \tag{5}$$

2.2. *Transformation of Poisson's Equation*

The Poisson's equation in the physical space (x, y, z) is given by

$$\frac{\partial}{\partial x}\left(\lambda\frac{\partial \phi}{\partial x}\right)+\frac{\partial}{\partial y}\left(\lambda\frac{\partial \phi}{\partial y}\right)+\frac{\partial}{\partial z}\left(\lambda\frac{\partial \phi}{\partial z}\right)=S, \tag{6}$$

with a general form of the boundary conditions

$$C_1\phi+C_2\mathbf{n}\cdot(\lambda\nabla\phi)=C_3. \tag{7}$$

Here, λ is a given space-dependent material coefficient and $\mathbf{n}$ is the unit-outward-normal vector on the boundary. Coefficients C_1, C_2, and C_3 can be chosen to produce Dirichlet, Neumann, or Robbin boundary conditions.

In order to perform computations in the transformed space, Eq. (6) must be transformed such that (ξ, η, ζ) are the independent variables. The following integral form of Poisson's equation applied to the volume element V' shown in Fig. 1 is transformed to preserve the conservation properties of the original equation (6),

$$\iiint_{V'}\nabla\cdot(\lambda\nabla\phi)\,dV=\iiint_{V'}S\,dV. \tag{8}$$

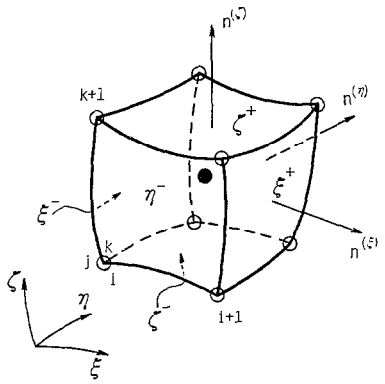

FIG. 1. Calculation cell defined by curvilinear coordinates in a physical space.

By using the Gauss divergence theorem, the left-hand side of Eq. (8) can be written as follows,

$$\begin{aligned}\iint (\lambda\nabla\phi)\cdot\mathbf{n}\,dS = {} & [(\lambda\nabla\phi)\cdot\mathbf{n}^{(\xi)}\,\Delta S^{(\xi)}]_{\xi^+} - [(\lambda\nabla\phi)\cdot\mathbf{n}^{(\xi)}\,\Delta S^{(\xi)}]_{\xi^-} \\ & + [(\lambda\nabla\phi)\cdot\mathbf{n}^{(\eta)}\,\Delta S^{(\eta)}]_{\eta^+} - [(\lambda\nabla\phi)\cdot\mathbf{n}^{(\eta)}\,\Delta S^{(\eta)}]_{\eta^-} \\ & + [(\lambda\nabla\phi)\cdot\mathbf{n}^{(\zeta)}\,\Delta S^{(\zeta)}]_{\zeta^+} - [(\lambda\nabla\phi)\cdot\mathbf{n}^{(\zeta)}\,\Delta S^{(\zeta)}]_{\zeta^-} \end{aligned} \tag{9}$$

where the subscripts ξ^+, etc., indicate evaluation on the surfaces of the volume element V', the area elements on the surfaces are

$$\Delta S^{(\xi)} = \sqrt{\alpha_{11}}\,\Delta\eta\,\Delta\zeta \tag{10a}$$

$$\Delta S^{(\eta)} = \sqrt{\alpha_{22}}\,\Delta\zeta\,\Delta\xi \tag{10b}$$

$$\Delta S^{(\zeta)} = \sqrt{\alpha_{33}}\,\Delta\xi\,\Delta\eta \tag{10c}$$

and the normal derivatives are

$$(\lambda\nabla\phi)\cdot\mathbf{n}^{(\xi)} = \frac{\lambda}{J\sqrt{\alpha_{11}}}\left(\alpha_{11}\frac{\partial\phi}{\partial\xi} + \alpha_{12}\frac{\partial\phi}{\partial\eta} + \alpha_{13}\frac{\partial\phi}{\partial\zeta}\right) \tag{11a}$$

$$(\lambda\nabla\phi)\cdot\mathbf{n}^{(\eta)} = \frac{\lambda}{J\sqrt{\alpha_{22}}}\left(\alpha_{21}\frac{\partial\phi}{\partial\xi} + \alpha_{22}\frac{\partial\phi}{\partial\eta} + \alpha_{23}\frac{\partial\phi}{\partial\zeta}\right) \tag{11b}$$

$$(\lambda\nabla\phi)\cdot\mathbf{n}^{(\zeta)} = \frac{\lambda}{J\sqrt{\alpha_{33}}}\left(\alpha_{31}\frac{\partial\phi}{\partial\xi} + \alpha_{32}\frac{\partial\phi}{\partial\eta} + \alpha_{33}\frac{\partial\phi}{\partial\zeta}\right). \tag{11c}$$

The right-hand side of Eq. (8) becomes

$$\iiint_{V'} S\,dV = SJ\,\Delta\xi\,\Delta\eta\,\Delta\zeta. \tag{12}$$

Using Eqs. (8) through (12) with a grid spacing of $\Delta\xi = \Delta\eta = \Delta\zeta = 1$, the equation applicable in the transformed space can be obtained as follows,

$$\left[\frac{\lambda}{J}\left(\alpha_{11}\frac{\partial\phi}{\partial\xi}+\alpha_{12}\frac{\partial\phi}{\partial\eta}+\alpha_{13}\frac{\partial\phi}{\partial\zeta}\right)\right]_{\xi^+} - \left[\frac{\lambda}{J}\left(\alpha_{11}\frac{\partial\phi}{\partial\xi}+\alpha_{12}\frac{\partial\phi}{\partial\eta}+\alpha_{13}\frac{\partial\phi}{\partial\zeta}\right)\right]_{\xi^-}$$
$$+\left[\frac{\lambda}{J}\left(\alpha_{21}\frac{\partial\phi}{\partial\xi}+\alpha_{22}\frac{\partial\phi}{\partial\eta}+\alpha_{23}\frac{\partial\phi}{\partial\zeta}\right)\right]_{\eta^+} - \left[\frac{\lambda}{J}\left(\alpha_{21}\frac{\partial\phi}{\partial\xi}+\alpha_{22}\frac{\partial\phi}{\partial\eta}+\alpha_{23}\frac{\partial\phi}{\partial\zeta}\right)\right]_{\eta^-}$$
$$+\left[\frac{\lambda}{J}\left(\alpha_{31}\frac{\partial\phi}{\partial\xi}+\alpha_{32}\frac{\partial\phi}{\partial\eta}+\alpha_{33}\frac{\partial\phi}{\partial\zeta}\right)\right]_{\zeta^+} - \left[\frac{\lambda}{J}\left(\alpha_{31}\frac{\partial\phi}{\partial\xi}+\alpha_{32}\frac{\partial\phi}{\partial\eta}+\alpha_{33}\frac{\partial\phi}{\partial\zeta}\right)\right]_{\zeta^-} = SJ \tag{13}$$

where all of the coefficients are calculated as part of the coordinate transformation and are known quantities.

The boundary conditions of Eq. (7) must also be transformed to the (ξ, η, ζ) space. Since all the boundaries coincide with the ξ-, η-, or ζ-constant plane, the transformed boundary conditions can be represented using Eqs. (11) as follows.

$$C_1\phi \mp C_2\frac{\lambda}{J\sqrt{\alpha_{11}}}\left(\alpha_{11}\frac{\partial\phi}{\partial\xi}+\alpha_{12}\frac{\partial\phi}{\partial\eta}+\alpha_{13}\frac{\partial\phi}{\partial\zeta}\right) = C_3 \tag{14a}$$

for ξ-constant Γ_1 or Γ_2 plane,

$$C_1\phi \mp C_2\frac{\lambda}{J\sqrt{\alpha_{22}}}\left(\alpha_{21}\frac{\partial\phi}{\partial\xi}+\alpha_{22}\frac{\partial\phi}{\partial\eta}+\alpha_{23}\frac{\partial\phi}{\partial\zeta}\right) = C_3 \tag{14b}$$

for η-constant Γ_3 or Γ_4 plane,

$$C_1\phi \mp C_2\frac{\lambda}{J\sqrt{\alpha_{33}}}\left(\alpha_{31}\frac{\partial\phi}{\partial\xi}+\alpha_{32}\frac{\partial\phi}{\partial\eta}+\alpha_{33}\frac{\partial\phi}{\partial\zeta}\right) = C_3 \tag{14c}$$

for ζ-constant Γ_5 or Γ_6 plane, where Γ_i $(i = 1, 2, \ldots, 6)$ are the six boundary surfaces as shown in Fig. 2.

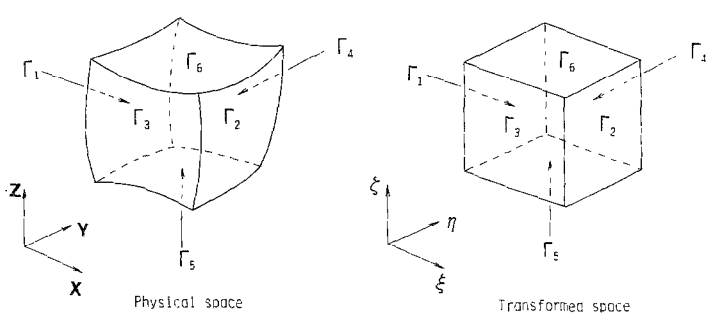

FIG. 2. Boundary surfaces of a hexahedron.

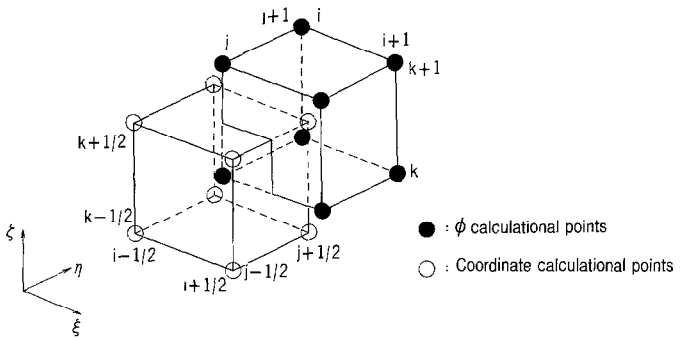

FIG. 3. Computational grid structure in a transformed space.

2.3. *Discretization of Poisson's Equation*

The transformed Eq. (13) with the boundary conditions (Eqs. (14)) is solved numerically in the (ξ, η, ζ) space by using a finite difference approximation with a grid spacing of $\Delta\xi = \Delta\eta = \Delta\zeta = 1$. Figure 3 shows a computational grid structure in the (ξ, η, ζ) space. While the (x, y, z) coordinates are specified at the corners of each cell, the variable ϕ in Eq. (13) is computed at the cell center. Therefore, the coefficients α_{jk} and the Jacobian J in Eq. (13) must be evaluated at the surfaces of the cell by averaging the values calculated at the four neighboring corners.

The boundary conditions (Eqs. (14)) are discretized using the hypothetical cell center outside the boundaries. For example, the expression at the boundary surface Γ_1 shown in Fig. 4 becomes

$$C_1 \frac{\phi_{1,j,k} + \phi_{0,j,k}}{2} - C_2 \frac{\lambda}{J\sqrt{\alpha_{11}}} \left\{ \alpha_{11}(\phi_{1,j,k} - \phi_{0,j,k}) + \alpha_{12} \frac{\phi^0_{1/2,j+1,k} - \phi^0_{1/2,j-1,k}}{2} + \alpha_{13} \frac{\phi^0_{1/2,j,k+1} - \phi^0_{1/2,j,k-1}}{2} \right\} = C_3 \qquad (15)$$

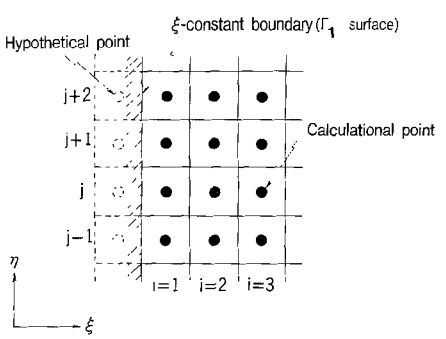

FIG. 4. Computational points on a ξ-constant plane for boundary conditions.

where ϕ_{ijk} is the value of ϕ at the cell center (i, j, k). Here the first-order partial derivatives with respect to η and ζ, $\partial\phi/\partial\eta$, and $\partial\phi/\partial\zeta$, are approximately calculated at the $i = 1/2$ plane using the values precalculated by the iteration process described in the following section. (The superscript 0 denotes the known value.) The boundary conditions at the other surfaces are expressed in the same way. These expressions are incorporated directly into the difference formula derived from Eq. (13) to eliminate ϕ at the hypothetical cell center.

Figure 5 represents the relevant calculational points in the finite difference formula. Instead of the familiar seven-point difference formula in the orthogonal coordinate system, the 19-point difference formula is derived in the boundary-fitted coordinate system. This is due to the cross-derivative terms of Eq. (13) such as

$$\left(\frac{\partial\phi}{\partial\eta}\right)_{\xi^+} = \frac{\phi_{i,j+1,k} + \phi_{i+1,j+1,k} - \phi_{i,j-1,k} - \phi_{i+1,j-1,k}}{4}. \tag{16}$$

The 19-point difference formula increases the number of non-zero elements in the coefficient matrix of a large system of linear equations, which causes increased computational time and storage capacity. Therefore, in the present study, the value of ϕ at the 12 computational points shown in Fig. 5 are treated as known values using the precalculated ones in the previous iteration, which results in the seven-point difference formula

$$a_{ijk}^{(6)}\phi_{i,j,k-1} + a_{ijk}^{(4)}\phi_{i,j-1,k} + a_{ijk}^{(2)}\phi_{i-1,j,k} + a_{ijk}^{(0)}\phi_{ijk} + a_{ijk}^{(1)}\phi_{i+1,j,k} + a_{ijk}^{(3)}\phi_{i,j+1,k} + a_{ijk}^{(5)}\phi_{i,j,k+1} = b_{ijk} \tag{17}$$

where the coefficients $a_{ijk}^{(n)}$ $(n = 0, 1, 2, \ldots, 6)$ and the known terms b_{ijk} are shown in the Appendix. The boundary conditions are taken into account by the coefficients $a_{ijk}^{(0)}$ and the known terms b_{ijk}.

The difference formula (17) is good approximation for near-orthogonal grid structure because the coefficients α_{ij} $(i \neq j)$ in the cross-derivative terms are much

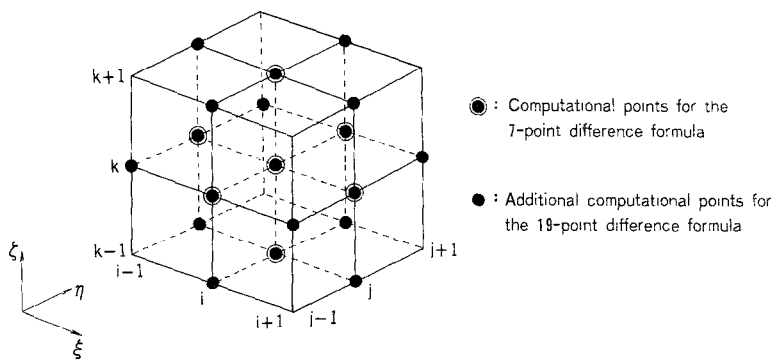

FIG. 5. Computational points used in difference formula.

smaller than α_{kk}, but it requires more iterations with the distortion of grid structure. In the present method, however, the know term in Eq. (17) is updated by the other kind of iteration needed for domain decomposition (see Sect. 3), so that the distorted grid structure does not always lead to large increase in computational time.

2.4. *Large System of Linear Equations*

The discrete approximation to Eq. (13), using finite difference formula (17), results in a large system of linear equations

$$\begin{bmatrix} a^{(0)}_{111} & a^{(1)}_{111} & & a^{(3)}_{111} & & a^{(5)}_{111} & 0 \\ a^{(2)}_{211} & \ddots & \ddots & & \ddots & & \\ & a^{(4)}_{121} & \ddots & \ddots & & \ddots & \\ & & a^{(6)}_{112} & \ddots & \ddots & & \\ & & & \ddots & \ddots & & \\ & & & & \ddots & & \\ 0 & & & & & & \end{bmatrix} \begin{bmatrix} \phi_{111} \\ \phi_{211} \\ \vdots \\ \phi_{NML} \end{bmatrix} = \begin{bmatrix} b_{111} \\ b_{211} \\ \vdots \\ b_{NML} \end{bmatrix}, \tag{18}$$

or, in brief, $A\phi = b$, where the unknown variables ϕ_{ijk} $(i = 1, 2,..., N;\ j = 1, 2,..., M;\ k = 1, 2,..., L)$ are arranged in the following order

$$\underbrace{\phi_{111}, \phi_{211},..., \phi_{N11}}, \underbrace{\phi_{121}, \phi_{221},..., \phi_{N21}}, ..., \underbrace{\phi_{1ML}, \phi_{2ML},..., \phi_{NML}}.$$

The coefficient matrix A is a regular sparse matrix and its order is $N \times M \times L$. Note here that the matrix A is also non-symmetric because of the relations

$$a^{(1)}_{i,j,k} \neq a^{(2)}_{i+1,j,k}, \qquad a^{(3)}_{i,j,k} \neq a^{(4)}_{i,j+1,k}, \qquad a^{(5)}_{i,j,k} \neq a^{(6)}_{i,j,k+1}, \tag{19}$$

and this property is due to both the non-orthogonality of the boundary-fitted coordinate systems and the approximated seven-point difference formula.

In the present study, the coefficient matrix A is approximated by an incomplete LU decomposition $K = LDU$, where L and U are lower and upper triangular matrices and D a diagonal matrix. The matrices L and U have the same sparsity pattern as the lower and upper triangular parts of A, respectively, except for the three diagonals adjacent to and between the two outermost diagonals. The matrix $K^{-1}A$ has eigenvalues all close to 1.0, so that Eq. (18) multiplied by K^{-1}, $K^{-1}A\phi = K^{-1}b$, can easily be solved by the biconjugate gradient method [4]. But this method requires about 1.7 times more computer storage capacity than the incomplete Cholesky conjugate gradient method [5, 6], which is widely used for a large symmetric matrix. Therefore, domain decomposition to reduce the order of a coefficient matrix is essential for the calculation of a large system of linear equations, as well as introduction of the seven-point difference formula (17) to decrease the number of non-zero elements in the matrix.

2.5. *Domain Decomposition and Overlapping Technique*

This technique is a variant of the well-known Schwarz alternating procedure [7]. A complicated 3-dimensional domain is divided into several hexahedron subdomains, each of which has six curved or plane surfaces. For each subdomain, the curvilinear coordinates (x, y, z) and the variable ϕ are calculated by solving Eqs. (2) and (13), respectively, and then they are joined with results for other subdomains to form composite solutions of the coordinates (x, y, z) and the variable ϕ for the original domain.

To ensure the composite solutions remain both continuous and smooth across the boundaries, an overlap of the solutions is adopted between any two adjacent hexahedrons. Figure 6 illustrates the procedure for overlapping the solutions. Since the staggered grid structure is used for the calculations of the coordinates and the variable ϕ, two kinds of overlapping are adopted in the present numerical scheme.

For the coordinate calculation, an overlap of two grid surfaces is used as shown in Fig. 6a. The inner grid coordinates are calculated using the Dirichlet boundary conditions on the six surfaces of a hexahedron. The calculated coordinates on the inner overlapped surface are transferred to the outer surface of the neighboring hexahedron, and the inner grid coordinates are calculated for the neighboring one. This procedure is repeated for all hexahedrons until the convergence requirement is met for all grid points.

For the calculation of the variable ϕ, on the other hand, three computational surfaces are overlapped as shown in Fig. 6b. The calculated values on the overlapped innermost surface are transferred to the hypothetical surface outside the neighboring hexahedron. Although this overlapping scheme needs double calculations on the midplane of the three overlapped surfaces, it does not change the calculational domain for each hexahedron, which makes it possible to apply the same treatment to all hexahedrons. (Contrast this with the alternative overlapping scheme shown in Fig. 6c.)

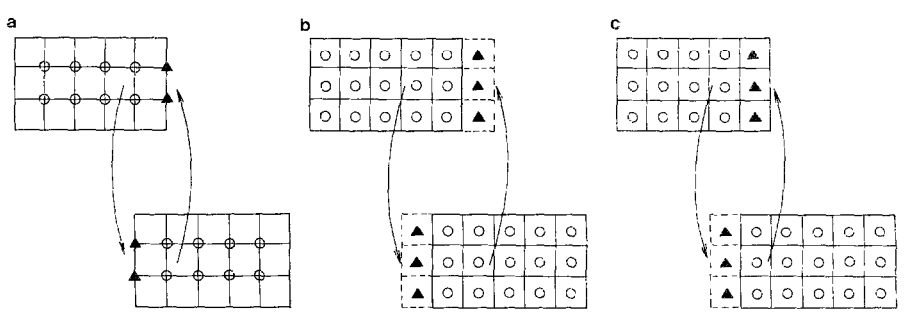

FIG. 6. Procedures for overlapping solutions among neighboring hexahedrons: (a) coordinate calculation process; (b) ϕ calculation process; (c) alternative calculation process.

3. Numerical Results and Discussion

The numerical procedure presented in Section 2 have been incorporated into a 3-dimensional calculation program POISSON–3D. The program solves the transformed Poisson's equation (13) in the transformed coordinate system. The curvilinear coordinates (x, y, z), the Jacobian J, and the coefficients α_{jk} of the corners of all cells are calculated by a grid generation program GRID–3D [2] and given to the POISSON–3D program.

The program uses two iteration schemes: the first involves solving the system of linear equations (18) using the preconditioned biconjugate gradient method; and the second calculates the interaction among the hexahedrons. The known terms in the approximated difference formula (17) are updated by the second iteration scheme. In the following, these schemes are termed inner and outer iteration schemes, respectively.

In order to examine the convergence process in these iteration schemes, the program was applied to a cylinder as shown in Fig. 7. In this example, the rectangular-type coordinate system was used to include the influence caused by the non-orthogonality of the boundary-fitted coordinate system. The cylinder was divided into several hexahedrons overlapping each other to include the interaction between the hexahedrons. The calculational conditions are also shown in Fig. 7, which resulted in a 1-dimensional axisymmetric problem with a bilinear analytical solution. The initial values of ϕ were set to zero in the whole domain.

The distributions along the two radial directions are compared with the analytical solution in Fig. 8. The numerical results had axisymmetric distributions and agreed with the analytical distribution within an error of 0.5% in the two directions. Figure 9 shows the convergence behavior for one hexahedron in comparison with results of SOR (Successive Over-Relaxation) iterative method using the

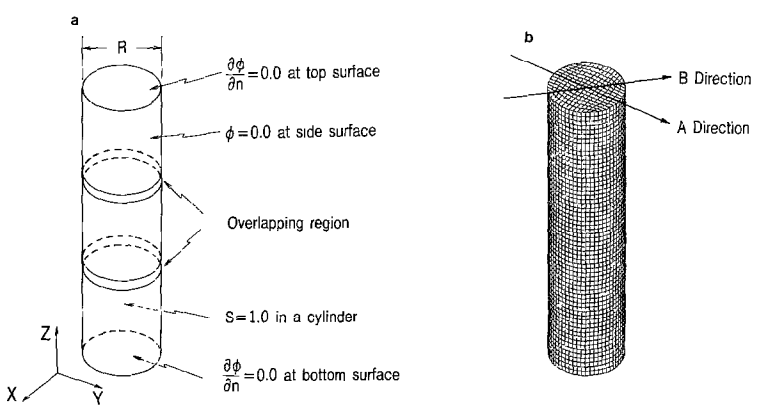

Fig. 7. Calculational conditions and grid structures for convergence test problem: (a) calculational conditions; (b) grid structure.

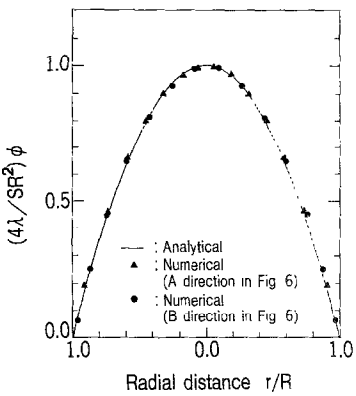

FIG. 8. Comparison between analytical and numerical solutions of ϕ.

optimum acceleration factor of 1.3. Each continuous line represents the convergence process of the inner iteration scheme, and the sequence of these lines results from the outer iteration scheme. It was assumed that the convergence criteria in each inner iteration was one tenth of the residue, $\|A\phi_i - b\|_2/\|b\|_2$, in the first iteration. This was due to the fact that the smaller convergence criteria offered no improvement for the next inner iteration. In the case of no domain decomposition ($N=1$), the discontinuity of the convergence process was caused by updating the known terms of Eq. (17) in the outer iteration. It should be noted that in the sequence of the solid lines ($N=2$), the residues were majorized by the terms of a

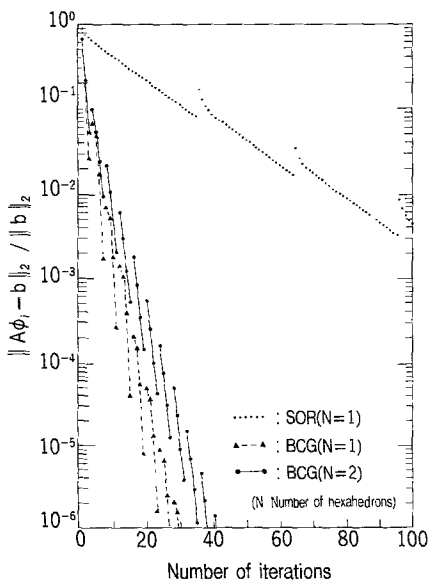

FIG. 9. Convergence behavior of POISSON-3D program.

geometric series with a ratio of about 0.3, which is characteristic of the Schwarz alternating procedure [7].

In Table I the convergence results are summarized. While the total number of iterations increased with the number of hexahedrons, the computational time required for each hexahedron decreased with the number of grid points in the hexahedron. This resulted in the relatively small increase in total computational time by about 65% even for the case of six hexahedrons. On the other hand, the core memory decreased in inverse proportion to the number of hexahedrons, which can overcome the difficulties related to dimensionality of 3-dimensional complicated geometries.

In order to demonstrate the applicability of the GRID-3D/POISSON-3D system, it was applied to the axisymmetric electron gun of a color picture tube. Figure 10a shows the electrode geometry and voltage of the prefocus and main lens sections. As shown in Fig. 10b, these sections were divided into 27 subdomains, which provided for a decrease of one-sixth in the needed CPU memory as compared to a case without domain decomposition. Figure 10c shows the perspective projection of the generated grid structure. The total number of grid points was 60400. Using P, Q, R functions (Eqs. (1)), the grid was concentrated toward the center of the axis and the cathode in order to increase accuracy along the electron beam trajectories. While the maximum grid size (400 μm) appears in the main lens section, the minimum one (10 μm) appears near the cathode.

The potential distribution and the electron beam trajectories for a case of cathode voltage of 100 V are shown in Fig. 10d. The beam trajectories were calulcated by

TABLE I

Convergence Behavior of POISSON-3D Program

Method	Number of hexahedrons	Grid structure of a hexahedron $I \times J \times K$[a]			Core memory (MB)	Number of iterations to reduce $\|A\phi_i - b\|_2/\|b\|_2 \leqslant 10^{-7}$	Computational time to reduce $\|A\phi_i - b\|_2/\|b\|_2 \leqslant 10^{-7}$ (relative)
SOR	1	15	15	62	3.87	334	2.91 (0.270)[c]
BCG	1	15	15	62	5.69	31	1.00[d] (1.00)
	2	15	15	32	3.18	85 (43)[b]	1.41 (0.513)
	3	15	15	22	2.34	131 (44)	1.46 (0.347)
	4	15	15	17	1.93	184 (46)	1.60 (0.269)
	5	15	15	14	1.69	234 (47)	1.63 (0.216)
	6	15	15	12	1.53	271 (45)	1.65 (0.189)

[a] I, J, and K are the maximum numbers of grid points in ξ, η, and ζ directions, respectively, for a hexahedron.

[b] Values in parentheses represent values per hexahedron.

[c] Values in parentheses represent values per iteration.

[d] Computational time was 55 s by HITAC M-200H.

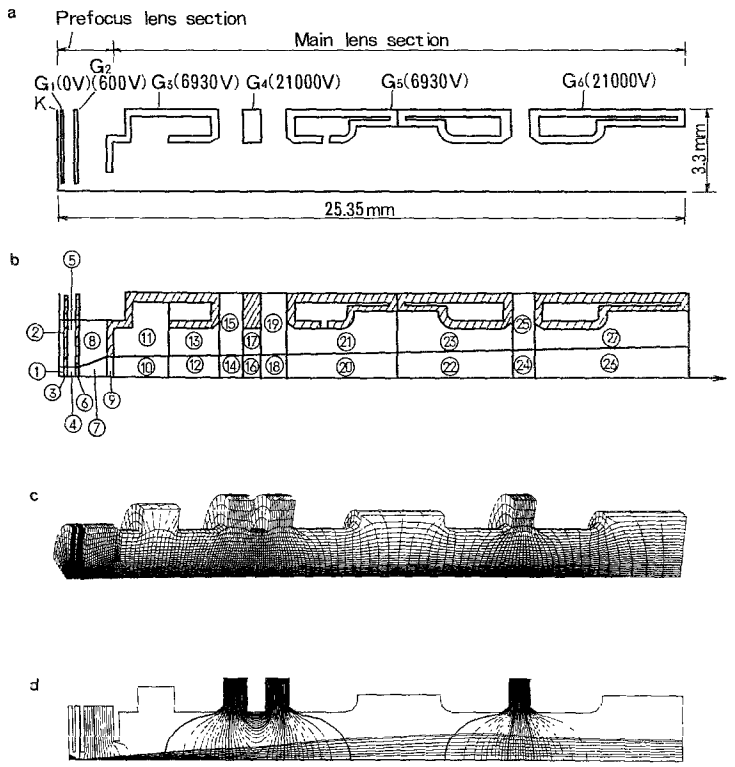

FIG. 10. Application to electrostatic field problem in electron gun of a color picture tube: (a) electrode configuration and voltage; (b) subdomain divisions: (c) generated grid; (d) potential and electron beam trajectories.

integrating the equation of motion for charged particles in the electric field. Note that the electron beams produce the space charge, which can change the potential distribution. Figure 10d shows the converged solution in a self-consistent field. Screen spot size was calculated by extrapolating the beam trajectories from the main lens exit to the screen. The calculated value agreed with the measured one using a photo-multiplier detector within 15% error [8].

4. CONCLUSION

A direct-solution scheme for numerically solving Poisson's equation in a 3-dimensional domain with arbitarily shaped boundaries has been developed by using a boundary-fitted coordinate transformation technique. The Poisson's equation and general boundary conditions were transformed in the simple domain and approximated by a 7-point finite difference formula. The 3-dimensional domain of

interest was decomposed into several hexahedrons, each of which had overlapping regions with each of its neighboring hexahedrons. This technique made it possible to deal with a complicated geometry consisting of many components. The large system of linear equations derived by discretizing the Poisson's equation has a non-symmetric coefficient matrix and was solved by using a biconjugate gradient method with incomplete LU factorization of the matrix.

On the basis of the scheme, the POISSON–3D computer program was developed and combined with the grid generation program GRID–3D. Numerical results showed the validity of the present scheme through comparisons with analytical results. Application to the electrostatic field problem in a electron gun verified the capability and feasibility of the GRID–3D/POISSON–3D system. Based on its automatic grid generation, geometric versatility, and small CPU memory, the present system provides significant progress towards improved productivity in the field of computer-aided engineering (CAE).

APPENDIX: Coefficients of 7-Point Difference Formula [Eq. (17)]

The coefficients $\alpha_{ijk}^{(n)}$ $(n=0, 1, 2,..., 6)$ and the known terms b_{ijk} are given as

$$a^{(0)}=\left(\frac{\lambda\alpha_{11}}{J}\right)_{\xi+}(f_2-1)+\left(\frac{\lambda\alpha_{11}}{J}\right)_{\xi-}(f_1-1)$$
$$+\left(\frac{\lambda\alpha_{22}}{J}\right)_{\eta+}(f_4-1)+\left(\frac{\lambda\alpha_{22}}{J}\right)_{\eta-}(f_3-1)$$
$$+\left(\frac{\lambda\alpha_{33}}{J}\right)_{\zeta+}(f_6-1)+\left(\frac{\lambda\alpha_{33}}{J}\right)_{\zeta-}(f_5-1) \tag{A.1a}$$

$$a^{(1)}=\left(\frac{\lambda\alpha_{11}}{J}\right)_{\xi+}+\left(\frac{\lambda\alpha_{12}}{4J}\right)_{\eta+}-\left(\frac{\lambda\alpha_{12}}{4J}\right)_{\eta-}+\left(\frac{\lambda\alpha_{13}}{4J}\right)_{\zeta+}-\left(\frac{\lambda\alpha_{13}}{4J}\right)_{\zeta-} \tag{A.1b}$$

$$b=\left(\frac{\lambda\alpha_{12}}{4J}\right)_{\xi+}(\phi_{i+1,j+1}-\phi_{i+1,j-1})+\left(\frac{\lambda\alpha_{13}}{4J}\right)_{\xi+}(\phi_{i+1,k+1}-\phi_{i+1,k-1})$$
$$-\left(\frac{\lambda\alpha_{12}}{4J}\right)_{\xi-}(\phi_{i-1,j+1}-\phi_{i-1,j-1})-\left(\frac{\lambda\alpha_{13}}{4J}\right)_{\xi-}(\phi_{i-1,k+1}-\phi_{i-1,k-1})$$
$$+\left(\frac{\lambda\alpha_{12}}{4J}\right)_{\eta+}(\phi_{i+1,j+1}-\phi_{i-1,j+1})+\left(\frac{\lambda\alpha_{23}}{4J}\right)_{\eta+}(\phi_{j+1,k+1}-\phi_{j+1,k-1})$$
$$-\left(\frac{\lambda\alpha_{12}}{4J}\right)_{\eta-}(\phi_{i+1,j-1}\phi_{i-1,j-1})-\left(\frac{\lambda\alpha_{23}}{4J}\right)_{\eta-}(\phi_{j-1,k+1}-\phi_{j-1,k-1})$$
$$+\left(\frac{\lambda\alpha_{13}}{4J}\right)_{\zeta+}(\phi_{i+1,k+1}-\phi_{i-1,k+1})+\left(\frac{\lambda\alpha_{23}}{4J}\right)_{\zeta+}(\phi_{j+1,k+1}-\phi_{j-1,k+1})$$

$$-\left(\frac{\lambda\alpha_{13}}{4J}\right)_{\zeta^-}(\phi_{i+1,k-1}-\phi_{i-1,k-1})-\left(\frac{\lambda\alpha_{23}}{4J}\right)_{\zeta^-}(\phi_{j+1,k-1}-\phi_{j-1,k-1})$$
$$-\left(\frac{\lambda\alpha_{11}}{J}\right)_{\xi^+}d_2-\left(\frac{\lambda\alpha_{11}}{J}\right)_{\xi^-}d_1$$
$$-\left(\frac{\lambda\alpha_{22}}{J}\right)_{\eta^+}d_4-\left(\frac{\lambda\alpha_{22}}{J}\right)_{\eta^-}d_3$$
$$-\left(\frac{\lambda\alpha_{33}}{J}\right)_{\zeta^+}d_6-\left(\frac{\lambda\alpha_{33}}{J}\right)_{\zeta^-}d_5+JS \tag{A.1c}$$

where the subscripts are abbreviated such that $i+1$ means $i+1, j, k$. The coefficients $a^{(2)}$, etc., are expressed in the same way as $a^{(1)}$. Variables f_1, h_1, g_1, and d_1 are defined as

$$f_1=\begin{cases}1-C_1\Big/\left(\dfrac{1}{2}C_1+C_2\dfrac{\lambda\sqrt{\alpha_{11}}}{J}\right) & \text{for } i=1,\\ 0 & \text{for } i=1 \text{ or overlapping region,}\end{cases} \tag{A.2a}$$

$$d_1=\begin{cases}\dfrac{\dfrac{1}{2}C_2\dfrac{\lambda}{J\sqrt{\alpha_{11}}}\{\alpha_{12}(\phi^0_{j+1}-\phi^0_{j-1})+\alpha_{13}(\phi^0_{k+1}-\phi^0_{k-1})\}+C_3}{\dfrac{1}{2}C_1+C_2\dfrac{\lambda\sqrt{\alpha_{11}}}{J}} & \text{for } i=1,\\ 0 & \text{for } i\neq 1,\\ \phi \text{ at a hypothetical point,} & \text{for overlapping region.}\end{cases} \tag{A.2b}$$

Other variables in Eqs. (A.1) are defined in the same way.

ACKNOWLEDGMENTS

The authors would like to acknowledge Drs. S. Yamada and K. Inoue of Energy Research Laboratory, Hitachi, Ltd., for their sustained encouragement.

REFERENCES

1. J. F. THOMPSON AND Z. U. WARSI, *J. Comput. Phys.* **47**, 1 (1982).
2. K. MIKI, T. TAKAGI, B. C. CHEN, AND W. T. SHA, *J. Comput. Phys.* **53**, 319 (1984).
3. C. W. MASTIN AND J. F. THOMPSON, *Numer. Math.* **29**, 387 (1978).

4. R. FLETCHER, in *Proceedings of Dundee Biennial Conference on Numerical Analysis, New York, 1975*, edited by G. A. Watson (Springer-Verlag, New York/Berlin, 1976).
5. D. S. KERSHAW, *J. Comput. Phys.* **26**, 43 (1978).
6. J. A. MEIJERINK AND H. A. VAN DER VORST, *J. Comput. Phys.* **44**, 134 (1981).
7. R. COURANT AND D. HILBERT, "Methods of Mathematical Physics," Vol. II, (John Wiley & Sons, New York, 1962), 293.
8. T. TAGAKI, K. MIKI, AND H. SANO, *J. Numer. Math.*, in press.