An International Journal
computers & mathematics
with applications

PERGAMON Computers and Mathematics with Applications 43 (2002) 621–630
www.elsevier.com/locate/camwa

Poisson Equation Solver with Fourth-Order Accuracy by using Interpolated Differential Operator Scheme

K. SAKURAI
Department of Energy Science, Tokyo Inst. of Tech.
2-12-1 O-okayama, Tokyo, Japan

T. AOKI
Global Sci. Info. and Compting Center, Tokyo Inst. of Tech.
2-12-1 O-okayama, Tokyo, Japan

W.-H. LEE
Department of Energy Science, Tokyo Inst. of Tech.
2-12-1 O-okayama, Tokyo, Japan

K. KATO
Research Lab. of Nuclear Reactors, Tokyo Inst. of Tech.
2-12-1 O-okayama, Tokyo, Japan

(Received August 2000; revised and accepted July 2001)

Abstract—A Poisson equation solver with fourth-order spatial accuracy has been developed by using the interpolated differential operator (IDO) scheme. The number of grids required for obtaining the results of numerical computation with the same accuracy as that by the second-order center finite difference method is drastically reduced. The consistency is confirmed to the use of the multigrid method and the red-black algorithm. The decrease of the CPU time by the multigrid method and by the parallel computing indicates the applicability of the IDO Poisson solver to large scale problems.

Keywords—Interpolated differential operator (IDO) scheme, Hermite interpolation, Poisson equation.

1. INTRODUCTION

The Poisson equation plays an important role in fluid dynamics, electromagnetism, and other processes. In numerical simulations in these fields, solving the Poisson equation requires long CPU time and large-scale computer memory. The discretized expression of the Poisson equation is described as a series of linear equations, and the coefficient matrix becomes large and sparse. Great effort has been made to develop efficient solvers. The majority of these solvers are designed

Parallel computing was carried out by using resources in the Computer Center in the Tokyo Institute of Technology in Japan.
This computing was supported partly by a Grant-in-Aid for Scientific Research from the Japanese Ministry of Education and the Frontier Research System for Global Change.

PII: S0898-1221(01)00308-X

to find fast algorithm for large sparse matrix [1,2]. If more accurate results are required in computational simulation, there are two choices; the use of a larger number of grids or the use of discretized expression with higher accuracy. The former method requires long CPU time and large-capacity computational memory. The latter method derives the coefficient matrix with wider nonzero band and complex boundary conditions. Recently, the authors have developed an interpolated differential operator (IDO) scheme [3,4] for solving a variety of partial differential equations. The Poisson equation is one of the typical ellipsoidal equations which can be accurately solved by the IDO scheme. In order to solve the Poisson equation, the Hermite interpolation of fifth-order polynomial [5] in space x is applied.

$$F(x) = ax^5 + bx^4 + cx^3 + dx^2 + f_{x,i}x + f_i, \tag{1}$$

where the subscript x denotes the derivative of x, $f_x = \frac{\partial f}{\partial x}$ and i is the grid index number as shown in Figure 1.

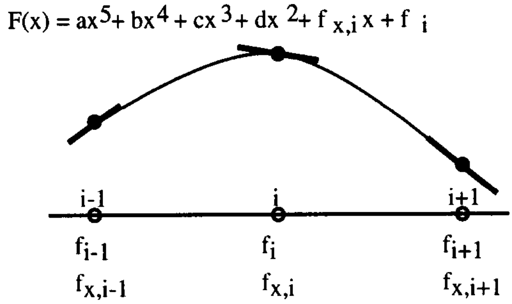

Figure 1. Illustration of fifth-order center interpolation of the IDO scheme.

The unknown coefficients included in equation (1) depend upon the matching conditions at the $i - 1^{\text{st}}$ and $i + 1^{\text{st}}$ grids; $F(-\Delta x) = f_{i-1}$, $F_x(-\Delta x) = f_{x,i-1}$, $F(\Delta x) = f_{i+1}$, $F_x(\Delta x) = f_{x,i+1}$, as follows,

$$a = -\frac{3}{4\Delta x^5}\left(f_{i+1} - f_{i-1}\right) + \frac{1}{4\Delta x^4}\left(f_{x,i+1} + 4f_{x,i} + f_{x,i-1}\right), \tag{2}$$

$$b = -\frac{1}{2\Delta x^4}\left(f_{i+1} - 2f_i + f_{i-1}\right) + \frac{1}{4\Delta x^3}\left(f_{x,i+1} - f_{x,i-1}\right), \tag{3}$$

$$c = \frac{5}{4\Delta x^3}\left(f_{i+1} - f_{i-1}\right) - \frac{1}{4\Delta x^2}\left(f_{x,i+1} + 8f_{x,i} + f_{x,i-1}\right), \tag{4}$$

$$d = \frac{1}{\Delta x^2}\left(f_{i+1} - 2f_i + f_{i-1}\right) - \frac{1}{4\Delta x}\left(f_{x,i+1} - f_{x,i-1}\right). \tag{5}$$

Higher-order spatial derivatives are obtained by differentiating equation (1) at the i^{st} grid,

$$\begin{aligned} \left.\frac{\partial^5 f}{\partial x^5}\right|_i &= \frac{\partial^5 F}{\partial x^5}(0) = 120a, & \left.\frac{\partial^4 f}{\partial x^4}\right|_i &= \frac{\partial^4 F}{\partial x^4}(0) = 24b, \\ f\left.\frac{\partial^3 f}{\partial x^3}\right|_i &= \frac{\partial^3 F}{\partial x^3}(0) = 6c, & \left.\frac{\partial^2 f}{\partial x^2}\right|_i &= \frac{\partial^2 F}{\partial x^2}(0) = 2d. \end{aligned} \tag{6}$$

Equation (1) indicates the fourth-order spatial accuracy in solving a one-dimensional Poisson equation. This paper describes the advantage of the higher-order Poisson solver developed by the authors and the applicability of simple SOR algorithm with the red-black method and the multigrid method [6,7] to vector computers and parallel computers.

2. DISCRETIZATION AND NUMERICAL ACCURACY

The IDO scheme procedure for solving a two-dimensional Poisson equation is described as follows

$$\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = \rho. \tag{7}$$

In order to construct two-dimensional Hermite interpolation, spatial derivatives f_x, f_y, and f_{xy} are introduced as dependent variables in addition to the physical value f. The following discretized expression can be obtained by replacing the second derivatives of equation (7) by (6):

$$\begin{aligned}&\frac{2}{\Delta x^2}\left(f_{i+1,j} - 2f_{i,j} + f_{i-1,j}\right) - \frac{1}{2\Delta x}\left(f_{x,i+1,j} - f_{x,i-1,j}\right)\\ &+\frac{2}{\Delta y^2}\left(f_{i,j+1} - 2f_{i,j} + f_{i,j-1}\right) - \frac{1}{2\Delta y}\left(f_{y,i,j+1} - f_{y,i,j-1}\right) = \rho_{i,j},\end{aligned} \tag{8}$$

where the subscripts i and j are the grid indices in the x- and y-directions, respectively. The use of the additional dependent variables, f_x, f_y, and f_{xy}, requires the following three equations in addition to (7) to determine all the variables self-consistently:

$$\frac{\partial^3 f}{\partial x^3} + \frac{\partial^3 f}{\partial x \partial y^2} = \frac{\partial \rho}{\partial x}, \tag{9}$$

$$\frac{\partial^3 f}{\partial x^2 \partial y} + \frac{\partial^3 f}{\partial y^3} = \frac{\partial \rho}{\partial y}, \tag{10}$$

$$\frac{\partial^4 f}{\partial x^3 \partial y} + \frac{\partial^4 f}{\partial x \partial y^3} = \frac{\partial^2 \rho}{\partial x \partial y}. \tag{11}$$

The discretized expressions corresponding to equations (9)–(11) can be reduced as shown below,

$$\begin{aligned}&\frac{15}{2\Delta x^3}\left(f_{i+1,j} - f_{i-1,j}\right) - \frac{3}{2\Delta x^2}\left(f_{x,i+1,j} + 8f_{x,i,j} + f_{x,i-1,j}\right)\\ &+\frac{2}{\Delta y^2}\left(f_{x,i,j+1} - 2f_{x,i,j} + f_{x,i,j-1}\right) - \frac{1}{2\Delta y}\left(f_{xy,i,j+1} - f_{xy,i,j-1}\right) = \rho_{x,i,j},\end{aligned} \tag{12}$$

$$\begin{aligned}&\frac{2}{\Delta x^2}\left(f_{y,i+1,j} - 2f_{y,i,j} + f_{y,i-1,j}\right) - \frac{1}{2\Delta x}\left(f_{xy,i+1,j} - f_{xy,i-1,j}\right)\\ &+\frac{15}{2\Delta y^3}\left(f_{i,j+1} - f_{i,j-1}\right) - \frac{3}{2\Delta y^2}\left(f_{y,i,j+1} + 8f_{y,i,j} + f_{y,i,j-1}\right) = \rho_{y,i,j},\end{aligned} \tag{13}$$

$$\begin{aligned}&\frac{15}{2\Delta x^3}\left(f_{y,i+1,j} - f_{y,i-1,j}\right) - \frac{3}{2\Delta x^2}\left(f_{xy,i+1,j} + 8f_{xy,i,j} + f_{xy,i-1,j}\right)\\ &+\frac{15}{2\Delta y^3}\left(f_{x,i,j+1} - f_{x,i,j-1}\right) - \frac{3}{2\Delta y^2}\left(f_{xy,i,j+1} + 8f_{xy,i,j} + f_{xy,i,j-1}\right) = \rho_{xy,i,j}.\end{aligned} \tag{14}$$

The detailed procedure for discretizing a two-dimensional Poisson equation by the IDO scheme is described below. For example, a discretized equation for higher-order derivative $\frac{\partial^3 f}{\partial^2 x \partial y}$ included in (9) can be obtained by constructing an interpolation of f_y in the x direction and differentiating it. The interpolation function $F_y(x)$ is obtained by replacing f and f_x in equation (1) by f_y and f_{xy}. The discretized expression of $\frac{\partial^2 F_y(0)}{\partial x^2}$ can be derived by using the relationship between the interpolation and the higher-order derivative expressed by (6).

Equations (8), (12)–(14) can be solved by relaxation procedures such as SOR (successive over relaxation) method. It is, however, difficult to apply these equations to CG (conjugate gradient) type solvers because of the asymmetry of the coefficient matrix.

In a three-dimensional case, eight dependent variables, f, f_x, f_y, f_z, f_{xy}, f_{xz}, f_{yz}, and f_{xyz} are used to construct interpolations self-consistently. The discretized expressions for a three-dimensional Poisson equation are obtained by using a procedure similar to that used for the two-dimensional case.

$$\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} + \frac{\partial^2 f}{\partial z^2} = \rho. \tag{15}$$

The following equations are obtained:

$$\begin{aligned} &\frac{2}{\Delta x^2}\left(f_{i+1,j,k} - 2f_{i,j,k} + f_{i-1,j,k}\right) - \frac{1}{2\Delta x}\left(f_{x,i+1,j,k} - f_{x,i-1,j,k}\right) \\ &+\frac{2}{\Delta y^2}\left(f_{i,j+1,k} - 2f_{i,j,k} + f_{i,j-1,k}\right) - \frac{1}{2\Delta y}\left(f_{y,i,j+1,k} - f_{y,i,j-1,k}\right) \\ &+\frac{2}{\Delta z^2}\left(f_{i,j,k+1} - 2f_{i,j,k} + f_{i,j,k-1}\right) - \frac{1}{2\Delta z}\left(f_{z,i,j,k+1} - f_{z,i,j,k-1}\right) = \rho_{i,j,k}, \end{aligned} \tag{16}$$

where subscripts x, y, and z denote the spatial differentiations in each direction, such as $f_x = \frac{\partial f}{\partial x}$ and the subscripts i, j, and k represent the grid index numbers in the x-, y-, and z-directions, respectively.

It is necessary to solve the following governing equations to complete the consistency between the number of the dependent variables and the equations employed:

$$\frac{\partial^3 f}{\partial x^3} + \frac{\partial^3 f}{\partial x \partial y^2} + \frac{\partial^3 f}{\partial x \partial z^2} = \frac{\partial \rho}{\partial x}, \tag{17}$$

$$\frac{\partial^3 f}{\partial x^2 \partial y} + \frac{\partial^3 f}{\partial y^3} + \frac{\partial^3 f}{\partial y \partial z^2} = \frac{\partial \rho}{\partial y}, \tag{18}$$

$$\frac{\partial^3 f}{\partial x^2 \partial z} + \frac{\partial^3 f}{\partial y^2 \partial z} + \frac{\partial^3 f}{\partial z^3} = \frac{\partial \rho}{\partial z}, \tag{19}$$

$$\frac{\partial^4 f}{\partial x^3 \partial y} + \frac{\partial^4 f}{\partial x \partial y^3} + \frac{\partial^4 f}{\partial x \partial y \partial z^2} = \frac{\partial^2 \rho}{\partial x \partial y}, \tag{20}$$

$$\frac{\partial^4 f}{\partial x^3 \partial z} + \frac{\partial^4 f}{\partial x \partial y^2 \partial z} + \frac{\partial^4 f}{\partial x \partial z^3} = \frac{\partial^2 \rho}{\partial x \partial z}, \tag{21}$$

$$\frac{\partial^4 f}{\partial x^2 \partial y \partial z} + \frac{\partial^4 f}{\partial y^3 \partial z} + \frac{\partial^4 f}{\partial y \partial z^3} = \frac{\partial^2 \rho}{\partial y \partial z}, \tag{22}$$

$$\frac{\partial^5 f}{\partial x^3 \partial y \partial z} + \frac{\partial^5 f}{\partial x \partial y^3 \partial z} + \frac{\partial^5 f}{\partial x \partial y \partial z^3} = \frac{\partial^3 \rho}{\partial x \partial y \partial z}. \tag{23}$$

The basic numerical accuracy of the IDO Poisson solver is estimated below. The two-dimensional Poisson equation for $0 \le x \le 1$, $0 \le y \le 1$ is solved using the source term of $\rho(x, y) = \cos(k_x x)\cos(k_y y)$ and $k_x = 5$, $k_y = 5$. For this estimation, the Dirichlet type boundary condition given by the analytic solution is used. The numerical accuracy is estimated by

$$\sum_{i=1}^{N} \sum_{j=1}^{M} \frac{\left|f_{i,j} - f_{\text{analytic},i,j}\right|}{\left|f_{\text{analytic},i,j}\right|} / \text{mesh number}, \tag{24}$$

where f_{analytic} stands for the analytic solution, and N and M represent the numbers of grids in the x- and y-directions, respectively. Equations (8), (12)–(14) are numerically calculated for the cases of the mesh number $N \times M = 9 \times 9$, 17×17, 33×33, 65×65, 129×129, and 257×257. For reference, the results obtained by the second-order center finite-difference method are shown in Figure 2.

In Figure 2, the gradient of the line representing the results of the IDO scheme indicates fourth-order accuracy. For the error level of 2×10^{-7}, the required memory of the finite center difference

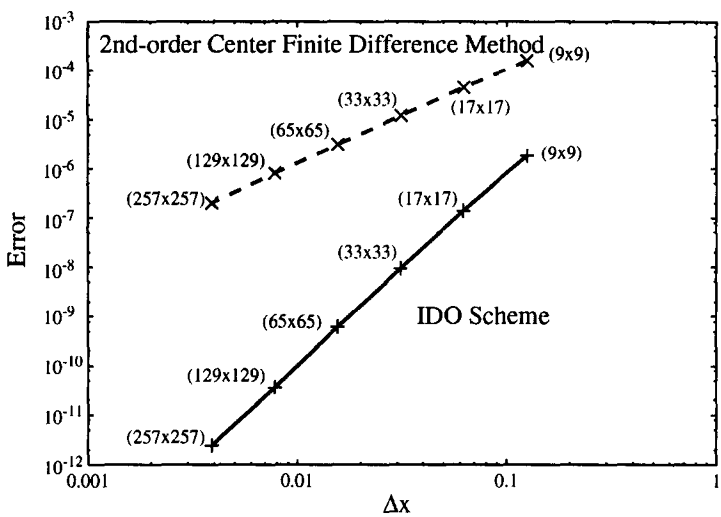

Figure 2. Accuracy of the results of numerical calculation by the two-dimensional Poisson equation. The accuracy of the results obtained by the IDO Poisson solver is on the order of Δx^4.

method and of the IDO scheme are estimated below. The finite center difference method uses grid number $M \times N = 257 \times 257$ and two variables, such as physical value f and source term ρ with double accuracy 8(Bytes). The required computational memory is estimated as follows,

$$\frac{257(\text{meshes}) \times 257(\text{meshes}) \times 8(\text{Bytes}) \times 2(\text{variables})}{1024} \simeq 1032(k\ \text{Bytes}).$$

The IDO scheme uses grid number $M \times N = 17 \times 17$ and two variables with double accuracy 8(Bytes). Each variable uses four dependent variables, such as f, f_x, f_y, and f_{xy}. The required computational memory of the IDO scheme is estimated as follows

$$\frac{17(\text{meshes}) \times 17(\text{meshes}) \times 8(\text{Bytes}) \times 2(\text{variables}) \times 4(\text{dependent variables})}{1024} \simeq 18\ (k\ \text{Bytes}).$$

The IDO Poisson solver requires less computer memory than the finite center difference method to obtain the fixed accuracy. When the number of grids required for computation is reduced, the CPU time is decreased from 85.869 sec (center FDM) to 0.008 sec (IDO) on the same computer.

In a three-dimensional or higher-dimensional case, there may be such cases where the computational memory or CPU time is limited. Judging from the results of the two-dimensional simulation, it is more effective to solve the three-dimensional Poisson equation with a coarse grid by using the IDO scheme. The numerical accuracy for a three-dimensional Poisson equation is estimated by using (25) with the calculation region $-1 \le x \le 1$, $-1 \le y \le 1$, $-1 \le z \le 1$, and source term $\rho(x,y,z) = \cos(k_x x)\cos(k_y y)\cos(k_z z)$, where $k_x = \pi$, $k_y = \pi$, $k_z = \pi$.

$$\sum_i \sum_j \sum_k \frac{|f(i,j,k) - f_{\text{real}}(i,j,k)|}{|f_{\text{real}}(i,j,k)|} / \text{mesh number}. \tag{25}$$

The solid line in Figure 3 indicates that the numerical accuracy is Δx^4 instead of Δx^2 which is the accuracy obtained by the second-order center finite-difference method.

3. MULTIGRID METHOD FOR IDO POISSON SOLVER

The multigrid (MG) method [6,7] is known as an effective algorithm to accelerate the convergence for relaxation methods. The MG method effectively reduces the error of longer wavelength

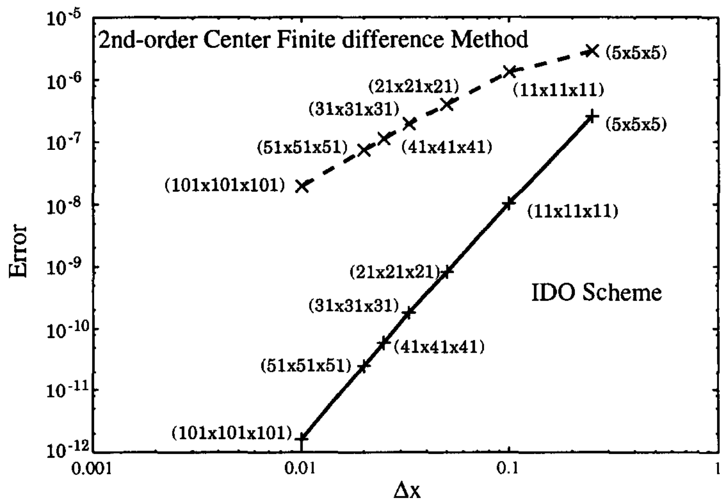

Figure 3. Accuracy of the results of numerical calculation by the three-dimensional Poisson equation. The accuracy estimated from the results obtained by the IDO Poisson solver is on the order of about Δx^4. The accuracy of numerical calculation by the IDO scheme with $11 \times 11 \times 11$ grids is on the same order as that obtained by the second-order center finite-difference method with $101 \times 101 \times 101$ grids.

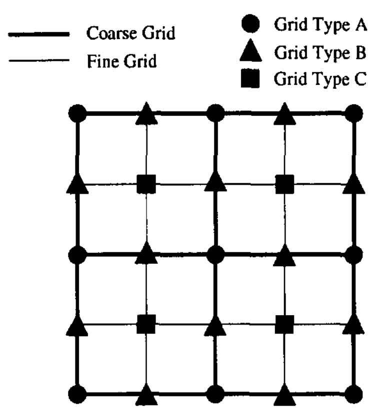

Figure 4. Restriction and prolongation in the multigrid procedure.

than the grid interval. Applying the MG method to the IDO Poisson solver, the effectiveness of the the SOR relaxation process is shown.

The restriction function for multigrid method is written as,

$$r = \frac{(r_{i+1,j+1} + r_{i,j+1} + r_{i-1,j+1} + r_{i+1,j} + 8 \times r_{i,j} + r_{i-1,j} + r_{i+1,j-1} + r_{i,j-1} + r_{i-1,j-1})}{16}, \quad (26)$$

and the prolongation functions are examined by using the following three types of functions corresponding to the grid types shown in Figure 4:

$$r = \begin{cases} r_{i,j}, & \text{(Grid type A)}, \\ \dfrac{r_{i+1,j} + r_{i,j}}{2} \text{ or } \frac{r_{i,j+1}+r_{i,j}}{2}, & \text{(Grid type B)}, \\ \dfrac{r_{i+1,j+1} + r_{i+1,j} + r_{i,j+1} + r_{i,j}}{4}, & \text{(Grid type C)}. \end{cases} \quad (27)$$

The restriction function and prolongation functions are used for all the dependent variables f, f_x, f_y, and f_{xy}.

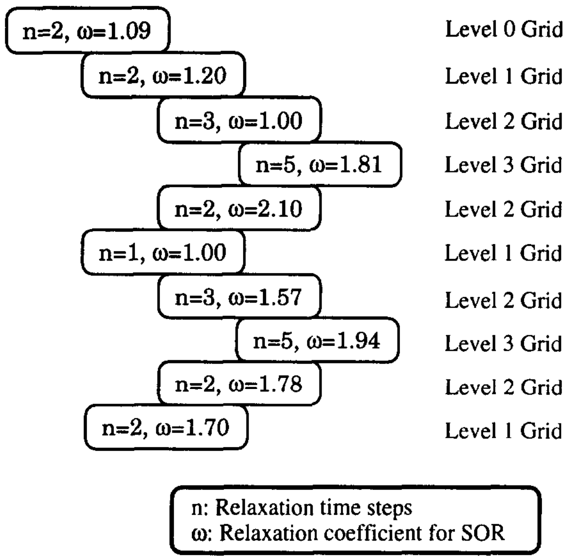

Figure 5. Illustration of the relaxation steps and coefficient for SOR.

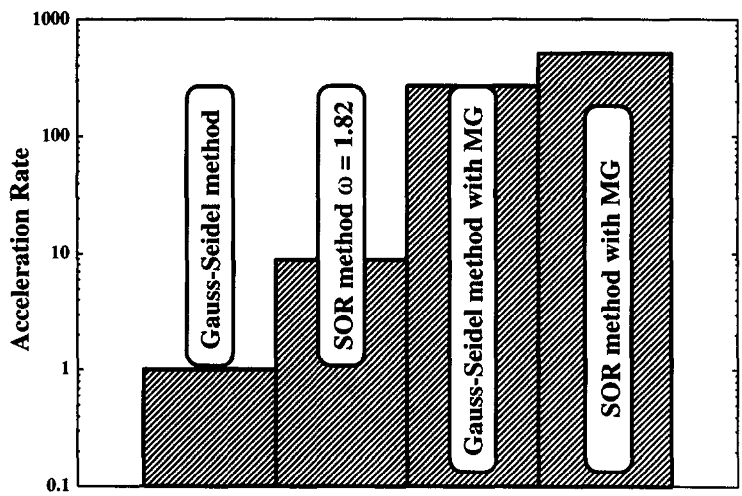

Figure 6. Acceleration rate of calculation time in comparison with the calculation time by the Gauss-Seidel method. The calculation time is faster by 8.9, 270, and 517 times compared with the Gauss-Seidel method with the SOR method, the Gauss-Seidel method using the multigrid method and the SOR method using the multigrid method.

For example, the calculation area of $-1 \leq x \leq 1$, $1 \leq y \leq 2$, $\Delta x = 2/350$, $\Delta y = 1/120$ is used and the source term is given by $\rho(x, y) = \sin(k_x x)\sin(k_y y)$ for $k_x = 6.5\pi$ and $k_y = 2.3\pi$. The boundary condition is imposed by the Dirichlet type. To solve this problem, a four-level and W-type cycle multigrid method is used. The relaxation steps n and relaxation coefficient ω for SOR method is shown in Figure 5.

The numerical error in the order of 10^{-13} is found as expected from Figure 2. To compare the acceleration of calculation time, the same problem was solved by using the Gauss-Seidel method, the SOR method with relaxation coefficient $\omega = 1.82$ and the Gauss-Seidel method with the multigrid method.

Figure 6 shows the acceleration rate in comparison with the calculation time by the Gauss-Seidel method. When the SOR method with relaxation coefficient $\omega = 1.82$ is used, the calculation time decreased 8.9 times compared with the time required by the Gauss-Seidel method.

When the multigrid method is implemented with the Gauss-Seidel method, the Poisson equation can be solved about 270 times faster. Moreover, the acceleration rate is improved 517 times by using the Poisson equation solver with the SOR method combined with the multigrid method. It is apparent that the MG method is effective for the improvement of the convergence of the IDO Poisson solver.

4. PARALLEL PROCEDURE

The other method for shortening the computational time is vectorization or parallelization using a vector computer or parallel computer. To ensure the effective use of these functions, the data access dependency on the calculation of the governing equation must be removed. When the one-dimensional Poisson equation is solved by the SOR method, the renovation of the physical value on the grid i requires the renovation on the grid $i-1$. In order to remove this dependency, the red-black algorithm has been proposed. All the grid points are classified into two groups, i.e., red group and black group as shown in Figure 7.

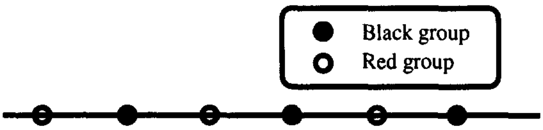

Figure 7. Red-black algorithm. The entire grid is divided into red and black groups. The values in the same group grid can be calculated without any dependency on calculation.

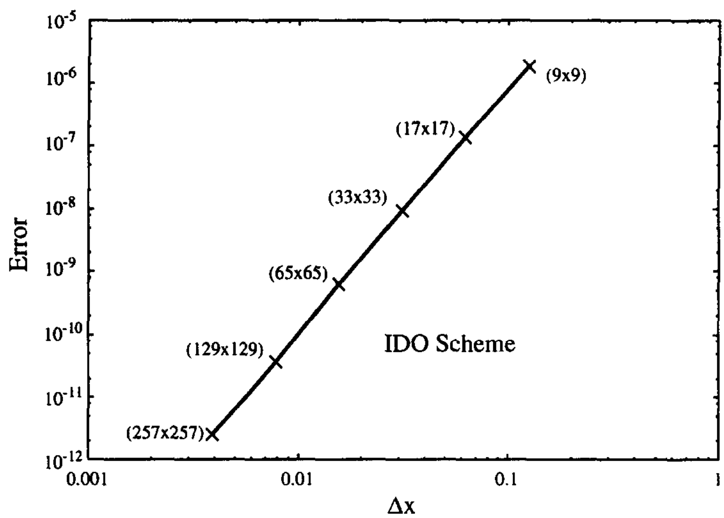

Figure 8. Errors in the results of numerical calculation by the two-dimensional Poisson equation solved by the IDO scheme implemented by the red-black algorithm. The errors estimated from the results obtained by the red-black algorithm are the same as those estimated by the normal procedure.

In the red-black algorithm, the values of the red group grid are calculated at the same time. After renovating these values, the values in the black group grid are calculated. The relaxation process advances, while repeating this procedure for the red and black groups alternately. The convergence by using this procedure is proved analytically.

First, let us estimate the numerical error by the IDO scheme in which the red-black algorithm is implemented. In the two-dimensional case, the numerical error estimated by (24) is exactly the same as that by the normal procedure shown in Figure 8. For the three-dimensional case, the source term $\rho(x,y,z) = \cos(k_x x)\cos(k_y y)\cos(k_z z)$, $k_x = \pi$, $k_y = \pi$, $k_z = \pi$ is given in

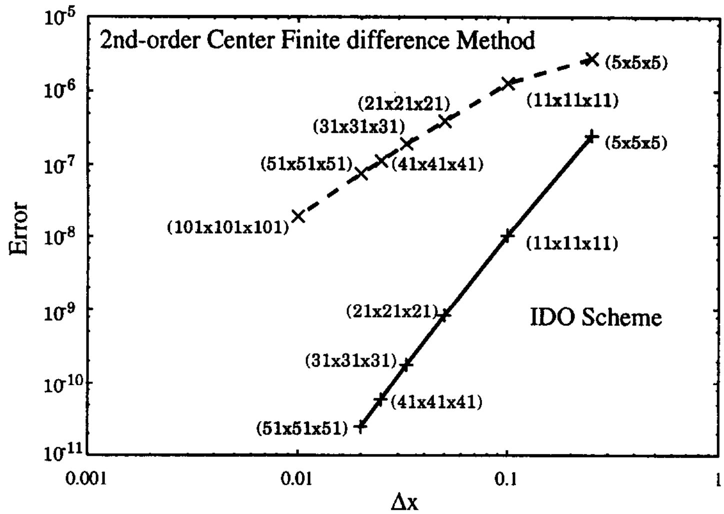

Figure 9. Errors in the results of numerical calculation by the three-dimensional Poisson equation solved by IDO scheme implemented by the red-black algorithm.

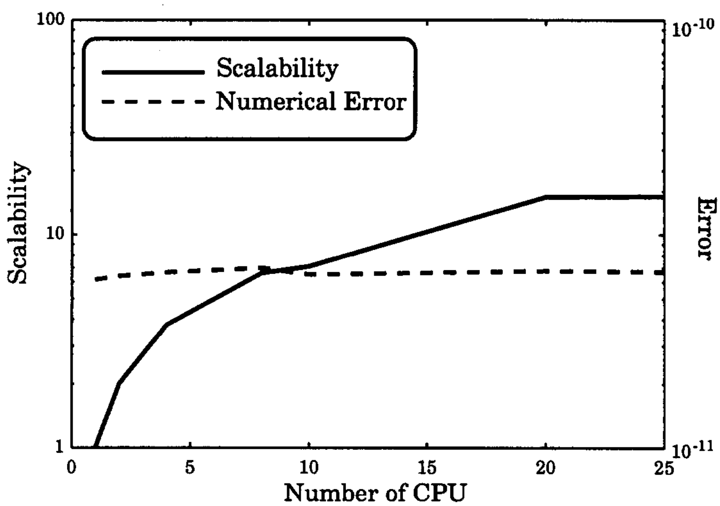

Figure 10. Scalability of the calculation speed for the Poisson equation solved by the IDO scheme implemented by the red-black algorithm. The calculation time is decreased to about 1/15 with 20 PEs.

$-1 \le x \le 1,\ -1 \le y \le 1,\ -1 \le z \le 1$. The numerical accuracy estimated by (25) is shown in Figure 9. The accuracy shown in Figure 9 coincides exactly with that shown in Figure 3 which is calculated by using the normal SOR algorithm.

Based on these results, it may be concluded that the IDO Poisson solver with red-black algorithm is applicable to vector and parallel computing. For the faster solution of the Poisson equation in the three-dimensional case, the calculation by parallel computing and the estimation of the scalability of the PE (processor element) number are attempted.

The time elapsed to solve the three-dimensional Poisson equation with $51 \times 51 \times 51$ grids with a source term $\rho(x, y, z) = \cos(k_x x)\cos(k_y y)\cos(k_z z)$ and $k_x = k_y = k_z = \pi$ in $-1 \le x \le 1,\ -1 \le y \le 1,\ -1 \le z \le 1$ is measured. The parallel computer used for estimating the scalability was a SGI Origin 2000 system with 128 CPUs of MIPS RISC R12000 300 MHz. To develop a parallel computing program, message passing interface (MPI) library was used. Figure 10 shows that the upper limit of scalability is about 15 with 20 PEs, although the problem has only 51 grids in each direction.

5. CONCLUSION

A multidimensional Poisson equation solver using the interpolated differential operator scheme has been presented. The Δx^4-order of spatial accuracy has been confirmed from the results of numerical calculation in the two-dimensional case and three-dimensional cases, respectively. Compared with the spatial accuracy of second-order center finite-difference method with 257×257 grids, the IDO scheme ensures the same spatial accuracy using only 17×17 calculation grids in the two-dimensional case described in the Section 2.

To decrease the CPU time in a relaxation process, the multigrid method and the red-black algorithm are implemented. Based on the results thus obtained, it can be concluded that the multigrid method and the red-black algorithm are applicable to the IDO scheme in a similar manner as the finite-difference method.

REFERENCES

1. M.H. Gutknecht, Variants of Bi-CGSTAB for matrices with complex spectrum, Tech. Report 91-14, IPS Research Report, (1991).
2. Y. Saad and M. Schultz, Conjugate gradient-like algorithms for solving nonsymmetric linear systems, *Math. Comp.* **44**, 417–424, (1985).
3. T. Aoki, Interpolated differential operator (IDO) scheme for solving partial differential equations, *Comp. Phys. Comm.* **102**, 132–146, (1996).
4. K. Sakurai and T. Aoki, Implicit IDO (interpolated differential operator) scheme, *Comp. Fluids Dynamics Journal* **8** (1), 6–12, (1999).
5. Y. Kondoh, Y. Hosaka and K. Ishii, Kernel optimum nearly-analytical discretization (KOND) algorithm applied to parabolic and hyperbolic equations, *Comput. Math. Applic.* **27** (3), 59–90, (1994).
6. A. Brandt, Multi-level adaptive solutions to boundary-value problems, *Math. Comput.* **31**, 333–390, (1977).
7. C. Qianshun, Use of the splitting scheme and multigrid method to compute flow separation, *Int. J. Num. Meth. Fluids* **7**, 719–731, (1987).