High-frequency electromagnetic field analysis using pseudo-quadruple precision in subdomain local solver

Amane Takei¹*, Hiroshi Kawai², Ryuji Shioya², Tomonori Yamada³

¹University of Miyazaki, Miyazaki, Japan
²Toyo University, Saitama, Japan
³University of Tokyo, Tokyo, Japan

¹takei@cc.miyazaki-u.ac.jp

Received: February 15, 2021; Accepted: July 19, 2021; Published: August 25, 2021

Abstract. The purpose of this study is an improvement of convergence properties of an interface problem in the high-frequency electromagnetic field finite element analysis based on the iterative domain decomposition method by introducing a pseudo-quadruple precision into the subdomain solver. In the high-frequency electromagnetic field analysis, it is well known that when the numerical model becomes huge scale, the convergence properties of the interface problem become extremely poor. To improve the deterioration of the convergence property in the interface problem, we propose to introduce a pseudo-quadruple precision into the iterative solver based on the conjugate gradient method in the subdomain problem. The pseudo-quadruple precision is constructed by ordinary double precisions. In this study, we apply the pseudo-quadruple precision for solving subdomain problems in the iterative domain decomposition method. As a result, we can obtain improvement of convergence properties in the iterative solver for the interface problem.

Keywords: Pseudo-quadruple precision, Domain decomposition method, Parallel finite element method, High-frequency electromagnetic field analysis

1. Introduction

The performance of the electromagnetic field analysis needs to be improved for reasons such as improving the accuracy of estimation of EMC and a cancer treatment using electromagnetism etc. [1, 2]. The authors have been investigating the parallel finite element method for high-frequency electromagnetic fields, which considers the displacement current in Maxwell’s
equations. The parallel full-wave electromagnetic field analysis solves the vector-wave equation taking an electric field as unknown functions. The solving matrix equation does not include uncertainty. However, the condition number of the matrix is large that becomes ill-conditioned for solving the matrix equation with iterative solvers, such as the conjugate gradient method [3, 4]. In the analysis, the iterative domain decomposition method (IDDM) is applied as a parallel technique. The IDDM is useful for large-scale problems, such as using mesh data with several millions to several tens of million elements. However, the convergence property of the iterative method in the interface problem is very poor in a large-scale analysis, as it is in an ordinary finite element analysis (FEA).

Therefore, it is necessary to improve the convergence of the interface problem. From the results of previous studies, it is known that improving the accuracy of the subdomain solution contributes to the improvement of the convergence of the interface problem. In the past, the direct method has been used for the subdomain solution in calculations up to several hundred million elements. However, since the amount of memory used increases in the order of the square with the increase in the number of degrees of freedom, it is desirable to consider an iterative solution method that can compute the solution with high accuracy while saving memory. In this study, we propose a method to improve the accuracy of the solution of subdomain problems by introducing quadruple precision to the floating point arithmetic in the iterative solution method of subdomain problems.

The parallel full-wave electromagnetic field analysis code being researched and developed in our research group is implemented in C and MPI/OpenMP for performance and portability. The __float128 type, provided in gcc-4.6 or later, is an implementation of 128-bit quadruple-precision arithmetic. However, at present, it is not sufficiently widespread in large-scale computing environments such as university information infrastructure centers. On the other hand, pseudo-quadruple precision is known to realize operations equivalent to quadruple precision by dividing quadruple precision into upper and lower digits and assigning double precision floating-point variables to each. The QD library [5] has been widely used for a calculation method by pseudo-quadruple precision. On the other hand, in this research, we will implement pseudo-quadruple precision using C for portability. By introducing pseudo-quadruple precision into the ICCOCG method for analyzing subdomain problems, it is expected to reduce the number of iterations until convergence is reached, and furthermore, it will be possible to improve the accuracy of approximate solutions to subdomain problems. This is expected to improve the convergence of the interface problem in IDDM. In addition, the convergence of the interface problem is expected to be improved by improving the accuracy of the subdomain solutions.

In past studies, some researches of computational science introduced pseudo-quadruple precision [6, 7]. These apply pseudo-quadruple precision to such as the element shape function and the iterative method to improve convergence. On the other hand, in this research, to improve the convergence of interface problems in the domain decomposition method, we apply
pseudo quadruple precision to improve the accuracy of subdomain calculation, with reference to these previous researches.

The introduction of pseudo-quadruple precision arithmetic to the iterative method of subdomain problems in the parallel full-wave electromagnetic field analysis based on IDDM proposed in this study has been shown to improve the convergence of the subdomain solution method, such that convergent solutions can be obtained for problems with poor convergence that do not converge with conventional double precision arithmetic. In addition, we have succeeded in reducing the number of iterations for the interface problem, which is reported in this paper.

2. Parallel high-frequency electromagnetic field analysis

2.1. Governing equations

Let \( \Omega \) with a boundary \( \partial \Omega \). The vector wave equations [8, 9] that describe an electromagnetic field with a single angular frequency \( \omega \) [rad/s] are derived from Maxwell’s equations containing the displacement current. The vector wave equations describing an electric field \( \mathbf{E} \) [V/m] are given by (1a) and (1b) below, for a current density \( J \) [A/m\(^2\)], and \( j \) is assigned as an imaginary unit:

\[
\text{curl}\left\{ \frac{1}{\mu} \text{curl}\mathbf{E} \right\} - \omega^2 \varepsilon \mathbf{E} = j \omega \mathbf{J} \quad \text{in} \quad \Omega, \tag{1a}
\]
\[
\mathbf{E} \times \mathbf{n} = 0 \quad \text{on} \quad \partial \Omega, \tag{1b}
\]
\[
\mathbf{J} = \sigma \mathbf{E}. \tag{1c}
\]

The permittivity and permeability are given by \( \varepsilon \) [F/m] and \( \mu \) [H/m], respectively. In this formulation, the permittivity has a complex value \( \varepsilon = \varepsilon' - \sigma / j \omega \). The electric field \( \mathbf{E} \) at known points is then substituted into (1a) from (1c), where the electrical conductivity is denoted as \( \sigma \) [S/m]. By solving (1a) under the boundary condition in (1b), we calculate the electric field \( \mathbf{E} \). For an absorbing boundary condition (ABC),

\[
(\text{curl}\mathbf{E}) \times \mathbf{n} - j \omega \sqrt{\varepsilon_0 \mu_0} (\mathbf{E} \times \mathbf{n}) \times \mathbf{n} = 0 \quad \text{on} \quad \partial \Omega_{\text{ABC}}. \tag{2}
\]

The first-order ABC to account for the Sommerfeld radiation condition is given by (2) [1]. Magnetic fields are calculated by the post-processing using Faraday’s law of induction (3).

\[
\text{curl}\mathbf{E} - j \omega \mu \mathbf{H} = \mathbf{0}. \tag{3}
\]
2.2. Finite element formulation

Next, we describe the finite element discretization. The electric field $\mathbf{E}$ is approximated with Nédélec elements (edge elements). The finite element approximation is performed as follows.

Find $\mathbf{E}_h$ such that

$$
\iint_{\Omega} \left(\frac{1}{\mu}\right) \nabla \times \mathbf{E}_h \cdot \nabla \times \mathbf{E}_h' \, dv - \omega^2 \iiint_{\Omega} \varepsilon \mathbf{E}_h \cdot \mathbf{E}_h' \, dv
+ j\omega \sqrt{\varepsilon_0 / \mu_0} \iint_{\partial\Omega_{ABC}} (\mathbf{E}_h \times \mathbf{n}) \cdot (\mathbf{E}_h' \times \mathbf{n}) \, dS = j\omega \iiint_{\Omega} \mathbf{J}_h \cdot \mathbf{E}_h' \, dv,
$$

(4)

where $\mathbf{J}_h$ is the electric current density approximated by the conventional piecewise linear tetrahedral elements.

2.3. Iterative domain decomposition method

We introduced the IDDM to high-frequency problems using the E method [6]. We convert the finite element equations of (4) to a matrix form as follows:

$$
Ku = f,
$$

(5)

where $K$ is the coefficient matrix, $u$ is an unknown vector, and $f$ is the known right-hand side vector. As shown in the following equation, the domain $\Omega$ is decomposed into $N$ pieces so that there is no overlap in the boundary between subdomains, namely

$$
\Omega = \bigcup_{i=1}^{N} \Omega^{(i)}.
$$

(6)

After domain decomposition, (5) is rewritten as follows:

$$
\begin{bmatrix}
K^{(1)}_H & \cdots & 0 & K^{(1)}_B R^{(1)}_B^T \\
0 & \ddots & \vdots & \vdots \\
0 & \cdots & K^{(N)}_H & K^{(N)}_B R^{(N)}_B^T \\
R^{(1)}_B K^{(1)}_B & \cdots & R^{(N)}_B K^{(N)}_B & \sum_{i=1}^{N} R^{(i)}_B K^{(i)}_B R^{(i)}_B^T
\end{bmatrix}
\begin{bmatrix}
\mathbf{u}_I^{(1)} \\
\vdots \\
\mathbf{u}_I^{(N)} \\
\mathbf{u}_B
\end{bmatrix}
= \begin{bmatrix}
\mathbf{f}_I^{(1)} \\
\vdots \\
\mathbf{f}_I^{(N)} \\
\mathbf{f}_B
\end{bmatrix},
$$

(7)

where subscripts $I$ and $B$ correspond to nodal points within the subdomains and on the subdomain interfaces, respectively. Here, $R^{(i)}_B^T$ is a 0-1 matrix to restrict that relates the internal degrees of freedom (DOFs) $\mathbf{u}_B^{(i)}$ of subdomain $\Omega^{(i)}$ with $\mathbf{u}_B$. Equations (8) and (9) are obtained from (7).
\[ K_{ii}^{(i)} u_i^{(i)} = f_i^{(i)} - K_{iB}^{(i)} u_B^{(i)}, \quad i = 1, \ldots, N \quad (8) \]

\[
\left\{ \sum_{i=1}^{N} R_B^{(i)} \left( K_{BB}^{(i)} - K_{iB}^{(i)} (K_{ii}^{(i)})^{-1} K_{iB}^{(i)} \right) R_B^{(i)T} \right\} u_B \\
= \sum_{i=1}^{N} R_B^{(i)} \left( f_i^{(i)} - K_{iB}^{(i)} (K_{ii}^{(i)})^{-1} f_i^{(i)} \right) 
\] \quad (9)

where \( \sum_{i=1}^{N} R_B^{(i)} f_B^{(i)} \) is the right-hand side vector of the equation regarding \( u_B \).

Equation (9) represents the interface problem of enforcing the continuity between subdomains in the domain decomposition method. Here, \((K_{ii}^{(i)})^{-1}\) is the inverse matrix of \( K_{ii}^{(i)} \). Next, equation (9) is rewritten as (10):

\[ Su_B = g, \quad (10) \]

where \( S \) is described as follows:

\[ S = \sum_{i=1}^{N} R_B^{(i)} S^{(i)} R_B^{(i)T}, \quad (11) \]

\[ S^{(i)} = K_{BB}^{(i)} - K_{iB}^{(i)} (K_{ii}^{(i)})^{-1} K_{iB}^{(i)}, \quad (12) \]

where \( S \) and \( S^{(i)} \) are the Schur complement matrix and the local Schur complement matrix in subdomain \( \Omega^{(i)} \), respectively.

### 2.4 Hierarchical domain decomposition method

The original analysis domain is first divided into parts, which are further decomposed into smaller domains called subdomains. This is called the hierarchical domain decomposition method (HDDM)[10, 11].

In the hierarchical processor mode (H-mode), processors are classified into three groups: grandparent, parent, and child. One of the processors is assigned as a grandparent, a few as parents, and others as children. The number of parent processors is similar to that of the parts (Fig. 1). The number of child processors can be varied, and it affects the parallel processing performance. However, because most communication time is spent between parent processors and child processors, the communication speed is important. Although computer performance has been improved by advances in network technology in recent years, a high-speed network is still expensive. Generally, in
typically used PC clusters, network speed is a bottleneck that degrades the processing performance of the CPU. Moreover, when parallel processing performance is considered, it is important to reduce the amount of communications as much as possible. Therefore, the parent-only parallel processor mode (P-mode) is more useful than the H-mode that uses all three groups [4].

In the P-mode, only parent processors perform the FEA, while the H-mode is computed by child processors. In the H-mode, although parent processors store some of the subdomain analysis data and coordinate the conjugate orthogonal conjugate gradient (COCG) iterations as primary work, the idle time of the CPU increases because the Parent processors perform fewer computations. In contrast, in the P-mode, all processors in the P-mode perform the FEA, and every CPU can be used without idleness in an environment with 10-20 CPUs. Thus, the P-mode is considered superior to the H-mode in terms of performance.

![Diagram of HDDM](image)

Figure 1: Solving algorithm of HDDM

Parts and subdomains are assigned in each calculation node. Then, the iterative calculation of the COCG method of equation (10) is started. In each step of the COCG method, equation (8) is solved by the incomplete Cholesky COCG (ICCOCG) method changing the boundary condition for equation (8). The convergence criteria in the interface and subdomain problems are set as $\epsilon = 1.0 \times 10^{-7}$ and $\delta = 1.0 \times 10^{-9}$, respectively. These criteria are defined to obtain the balance between the accuracy of the solution and the calculation time.

In our past research, we showed that HDDM is effective in large-scale high-frequency
electromagnetic field analysis based on the finite element method [6, 8, 9]. In this research, we introduce the pseudo-quadruple precision to improve the convergence property of the subdomain problem. Also, we consider improving method of the convergence property of the interface problem by increasing accuracy of subdomains solutions.

3. Introduction of pseudo-quadruple precision

3.1. Conversion from double precision to pseudo-quadruple precision based on IEEE754

In this study, the pseudo-quadruple precision is introduced to the calculation of the iterative method of the subdomain problem as a way to realize a 106-bit mantissa by using two double precision floating-point numbers based on IEEE 754 [12].

The machine epsilon is defined as $\varepsilon_m = 2^{-53}$. When we consider that a real number “alpha” is represented by a pseudo-quadruple precision number A, alpha is presented by $A.hi + A.lo$. Here, $A.hi$ is a high-order digit and $A.lo$ is a low-order digit. This type of conversion is based on equations (13) below. The coefficient of equation (13a) is the calculation result of $\text{pow}(2, 27) + 1$, which is needed to separate $A.hi$ and $A.lo$ from the 53-bit mantissa.

$$\text{temp} = 134217729.0 \times \text{alpha}, \quad (13a)$$
$$A.hi = \text{temp} - (\text{temp} - \text{alpha}), \quad (13b)$$
$$A.lo = \text{alpha} - A.hi. \quad (13c)$$

Functions to calculate with the pseudo-quadruple precision are implemented according to the reference [12] in the following sections.

3.2. Calculation of pseudo-quadruple precision

In this study, calculations of the addition and the multiplication for pseudo-quadruple precision are implemented according to the reference [12]. In contrast, the division calculation is implemented according to the reference [13]. However, in the algorithm of the IDDM, the division by zero must be avoided when a very huge number is substituted as a calculation number. Hence, the threshold is introduced as $\text{DBL}_HUGE = 10^{300}$. When high-order $A.hi$ of a calculation number is bigger than $\text{DBL}_HUGE$, zero is returned as a calculation result by the IF statement.

Figure 2 shows an example of the implementation of the reciprocal operation in C language [12]. In the figure, $a_h$ is the upper digits of the input, $a_l$ is the lower digits of the input, and $x_h$ and $x_l$ are the results of the operation. The function $\text{two_product}$ used here is an error-free transformation of multiplication, and $\text{fast_two_sum}$ is a function to calculate the sum of
two values whose absolute values are known to be large or small.

The square root operation applied to the inner product and norm operations in the ICCOCG method requires the avoidance of zero division when zero is assigned to the number under operation, and a procedure for this purpose has been added to the algorithm in [11]. An example implementation of the square root operation in C is shown in Fig. 3 [12]. These are implemented based on [13].

```c
void DD_Sqrt(double *x_h, double *x_l,
             double a_h, double a_l)
{
    if ( a_h == 0.0 ) {
        (*x_h) = (*x_l) = 0.0;
        return;
    }
    double app = sqrt( a_h );
    double p1, p2, resl;
    two_product(&p1, &p2, app, app);
    resl = 0.5 * (((a_h - p1) - p2) + a_l) / app;
    fast_two_sum(x_h, x_l, &app, &resl);
}
```

Figure 2: Implementation of reciprocal operation using pseudo-quadruple precision in C

```c
void DD_Reciprocal(double *x_h, double *x_l,
                    double *a_h, double *a_l )
{
    if ( fabs(*a_h) > DBL_HUGE ) {
        *x_h = *x_l = 0.0;
        return ;
    }
    double r1, r2, resh, resl;
    resh = 1.0 / (*a_h);
    two_product(&r1, &r2, resh, *a_h);
    resl = ((1.0 - r1) - r2) * resh - resh * (*a_l) * resh;
    fast_two_sum(x_h, x_l, &resh, &resl);
}
```

Figure 3: Implementation of square root operation using pseudo-quadruple precision in C

3.3. **Introduction into subdomain solver**

In this section, the introduction of pseudo-quadruple precision type into the ICCOCG solver in
the subdomain problem is described. The data type of the solving linear algebra equation is the complex type. Therefore, variables for matrices and vectors become the complex type. These complex type variables are expressed by using structure of the C language. When floating-point values in these variables are used in the ICCOCG method, values of the double precision type are converted the pseudo-quadruple types. The schematic of data I/O in the subdomain solver is shown in Fig. 4.

Variables of vectors and matrices are converted into real and imaginary parts. Conversely, the original double precision data are passed for the high-order digits in the pseudo-quadruple type, and zero is substituted for the low-order digits. After processing by the ICCOCG method, the pseudo-quadruple data are passed the COCG method in the interface problem rounding for the double precision data.

![Schematic of data I/O in subdomain solver](image)

**Figure 4:** Schematic of data I/O in subdomain solver

### 4. Numerical analysis

#### 4.1. Verification of subdomain solver

For verification of the pseudo-quadruple type in the ICCOCG method, calculations are performed with three kinds of linear equations. In the verification, the IC preconditioner is not used in the check of the effect for introducing of the pseudo-quadruple type into the COCG method. The testing linear equations, which have a complex symmetric system from high-frequency electromagnetic field problems, are qc324, dwg961b, and qc2534 in the Sparse Matrix Collection [14], published by University of Florida. In the high-frequency problem, there is the characteristic that eigenvalues gather around the origin point. Then, when the matrix size...
becomes large and the materials become complex, the convergence characteristic become worth [15, 16]. Properties of these matrices are shown Table 1 [14].

The computing environment is a workstation with Intel Core i7 950 (3.07 GHz/L2 8 MB) and 32 GB RAM. In the compile, gcc-4.5.1 with -O0 is used. Calculation results are shown in Table 2. In this table, ‘D’ means a calculation by the ordinary double precision type, and “DD” means a calculation by the pseudo-quadruple type. Convergence properties of the residual norm are shown Figure 5.

Iteration counts of qc324 and dwg961 are reduced into each 1/3 and 1/4 compared by the case of the ordinary double precision type. On the other side, in the case of qc324, elapsed time is increased 4.3 times of the case of double type precision. And in the case of qc961b, elapsed time is increased 5 times. Generally, the computation is increased from ten several times compared in the case of double precision type. In these cases, increasing ratio of elapsed time is saved around several times of double precision. It is necessary to reduce elapsed time by the contribution in implementations. As shown in Fig. 5 (a), in the case of qc324, the convergence characteristic is improved drastically after 550 times of iteration the pseudo-quadruple precision type. And as shown in Fig. 5(b), to solve dwg961b is easier than qc324 by using both precision types. Even with such easy-to-solve problems, the effectiveness is found. On the other hand, in the case of qc2534, the calculation is not completed by using double precision, however the calculation is completed by using pseudo-quadruple precision. As you can see Fig.5(c), this linear equation is very difficult to solve by using the double precision type. In the calculation by double precision, the residual norm oscillates around $10^8$, while pseudo-quadruple precision successfully solved. Thus, it can be seen that even with a problem of poor convergence that a solution cannot be obtained with double precision, it is possible to calculate by the pseudo-quadruple precision.

In addition, the memory requirement increases only the double precision type vector of the size of unknown number × 3 in the COCG algorithm by introducing the pseudo-quadruple precision. The dominant memory requirement is about the nonzero coefficient matrix to be input.

From the above, the COCG method implemented with pseudo-quadruple precision requires several times of computation time in the double precision type. It also reduces the number of iterations considerably and converges even with problems that do not converge with double precision, such as poor convergence. The reason why the convergence is improved by applying the pseudo-quadruple precision is that the calculation accuracy of the residual norm in the COCG method is improved.
Table 1: Matrix equations for verification of subdomain solver

|       | Matrix size | Num. of nonzero |
|-------|-------------|-----------------|
| qc324 | 324 × 324   | 26,730          |
| dwg961b | 961 × 961   | 10,591          |
| qc2534 | 2,534 × 2,534 | 463,360        |

Table 2: Solving of each matrix equation

|       | Iteration count | Calc time [s] | Used max memory [kB] |
|-------|-----------------|---------------|----------------------|
|       | (D) | (DD) | (D) | (DD) | (D) | (DD) |
| qc324 | 2,986 | 736  | 1.559 | 6.687 | 559.1 | 585.0 |
| dwg961b | 188,197 | 58,141 | 11.32 | 56.29 | 279.8 | 356.7 |
| qc2534 | - | 22,824 | - | 3,438 | 9,446 | 9,648 |

(a) qc324  
(b) dwg961b  
(c) qc2534

Figure 5: Comparison of the convergent history of the residual norm

4.2. **Improving of convergence of interface problem**

In this section, the effect of reducing the number of iterations of interface problems in IDDM
is confirmed by introducing the pseudo-quadruple precision arithmetic in the iterative solution of subdomain problems. The model used for the analysis is the TEAM Workshop Problem 29, a reentrant cavity resonator model known as one of the standard problems for high-frequency electromagnetic field analysis shown in Fig. 6 [17]. The resonator is a cylindrical type with a diameter of 1.9 [m] and a height of 1.45 [m], and a disk-shaped dielectric phantom with a relative permittivity $\varepsilon_r = 80$ and a conductivity $\sigma = 0.52$ [S/m] is placed. The excitation source is an antenna installed on the lower surface and is driven at a frequency of $f = 60.0$ [MHz]. Figure 6 (a) shows the outline and dimensions [m] of the analysis model. The excitation source is an antenna installed on the lower surface and is driven at a frequency of $f = 60.0$ [MHz]. Figure 6 (a) shows the outline and dimensions [m] of the analysis model. The mesh is divided based on this model, and two types of mesh models (Mesh_A and Mesh_B) with different numbers of elements are prepared. Figure 6 (b) is an example of mesh division (Mesh_B). Table 3 shows the specifications of each mesh.

| Mesh   | Num. of elements | DOFs   | Num. of Subdomains |
|--------|------------------|--------|--------------------|
| Mesh_A | 979,660          | 1,217,613 | 8,960             |
| Mesh_B | 101,445          | 125,633 | 1,008              |

The computer environment is a PC cluster equipped with an Intel Core i7 2600K (3.40 GHz / L28 MB) multi-core CPU and 24 GB RAM, and 20 units (80 cores) are used in the calculation of Mesh_A. The compiler and compilation options used are gcc-4.5.1, -O0, and the parallelization method and implementation method for this PC cluster are as shown in Sections 2.3 and 2.4, respectively. In addition, the MPI (Message Passing Interface) is applied as a parallel library.
First, in the calculation using Mesh_A, we observe a change in the convergence history of the residual norm of the interface problem. Normally, in our research, the convergence test value $\delta = 1.0e-07$ for the interface problem and the convergence test value $\epsilon = 1.0e-09$ for the subdomain problem are set for calculation [5]. In the calculation of Mesh_A, the same convergence test values are set, and the subdomain problem is calculated and compared with two patterns, double precision, and pseudo-quadruple precision.

Figure 7 shows the convergence history of the interface problem, and Table 4 shows the number of iterations and the calculation time. From Fig. 7 and Table 4, the accuracy of the solution of the subdomain is improved and the number of iterations of the interface problem is reduced by about 45%. On the other hand, the calculation time increase 5.2 times.

![Figure 7: Convergence property of residual norm in interface problem (Mesh_A)](image)

| Iteration count | Elapsed time [s] |
|-----------------|------------------|
|     (D)         |      (DD)       |
| 8,022           |    4,402        |
| 2,412           |   12,757        |

Next, in order to confirm the improvement of the convergence of the interface problem in detail, in the analysis using Mesh_B, the calculation is performed without setting the convergence judgment value of the interface problem. In addition, the accuracy of the subdomain's solution gives to the number of iterations of the interface problem when the convergence criterion of the ICCOCG method for the subdomain problem is lowered from $\delta = 1.0e-09$ to $\delta = 1.0e-12$ and $\delta = 1.0e-15$. Regarding the effect and the improvement of convergence by using pseudo-quadruple precision, we investigate two patterns of calculation of double precision and pseudo-quadruple precision. In these numerical experiments, in order to suppress the vibration width of the residual norm and make it easier to understand the difference due to the introduction of pseudo-quadruple precision, the calculation is performed using a small-scale model with fewer elements than Mesh_A. The maximum number of iterations is set to 20,000. The computer used was 3 PC clusters (12 cores) used to calculate Mesh_A. Figure 8 shows the
convergence history of the residual norm of the interface problem. Table 5 shows the calculation results when the convergence criteria of the residual norm of the interface problem is set to $\delta = 1.0e-15$.

![Figure 8: Convergence property of residual norm in interface problem (Mesh_B)](image)

From Fig. 8 and Table 5, the convergency of the interface problem can be improved by stricter the convergence criteria of the subdomain problem in both cases with the double precision and the pseudo-quadruple precision. From this result, it can be seen that the improvement of the accuracy of the solution of subdomains contributes to the improvement of the convergence of the interface problem. In addition, it can be confirmed that the convergency of the interface problem is improved by the calculation with the pseudo-quadruple precision rather than the double precision in all the cases of the convergence criteria for the subdomain problem. Comparing the number of iterations when the residual norm of the interface problem reaches $\delta = 1.0e-15$, when the convergence criteria for the subdomain problem is $\epsilon = 1.0e-12$, the calculation with pseudo-quadruple precision converged with about 4.3% fewer iterations than the double precision. Also, the number of iterations is reduced by 8.1% in the case of $\epsilon = 1.0e-15$. Furthermore, in the case of $\epsilon = 1.0e-09$, even if the number of iterations of the interface problem is over 20,000 times, the residual norm of the interface problem does not converge to $\delta = 1.0e-15$ with double precision. On the other hand, by pseudo-quadruple precision, the residual norm of the interface problem converges.

In the case of the convergence criteria for the subdomain problem $\epsilon = 1.0e-12$, the elapsed time to reach the convergence criteria $\delta = 1.0e-15$ for the interface problem is about 6.5 times compared with the double precision. In addition, in the case of $\epsilon = 1.0e-12$, it is about 6.3 times.
4.3. Numerical experiment using another model

In this section, the effectiveness of the proposed method is verified using a different numerical model from the previous section. This numerical model is a coaxial cable model, which consists of a circularly cut cylinder with a rectangular column as the inner conductor in the center. The outer conductor is expressed by the boundary condition (1b).

A schematic of the numerical model is shown in Fig. 9. In this model, the dimensions of the inner conductor $xI$ and $zI$ are 0.1 [m], the height of the cylinder $d(=yI)$ is 0.2 [m], and the radius $r$ is 9.4 [m]. The number of elements is 938,256, and the analysis frequency is $f = 140$ [MHz]. The number of iterations when the convergence decision value of $1.0\times10^{-7}$ is reached for the interface problem and the computation time are shown in Table 5.

![Figure 9: Coaxial conductor model](image)

Table 5: Comparison of numerical results

| Iteration count: | Elapsed time [s] |
|-----------------|------------------|
| (D)             | (DD)             |
| 3,040           | 2,099            |
| 967             | 3,822            |

When the pseudo-quadruple precision is applied to the subdomain problem, the number of iterations is reduced by 31% compared to double precision. In addition, although the computation time increases, it is only about four times longer. As a result, the application of pseudo-quadruple precision to the subdomain problem improves the computational performance and shows the effectiveness of the proposed method. In the future, we will review the algorithm to further reduce the increased computation time.

5. Conclusion

In this research, pseudo-quadruple precision was introduced for the iterative method of the subdomain problem for the full-wave electromagnetic field analysis based on the iterative domain decomposition method. Purposes are to improve the convergence of the iterative method of the subdomain problem and to reduce the iteration count of the interface problem.
As a result, the iteration count was reduced in the subdomain problem. Also, we confirmed that the convergence property was stabilized. Moreover, we confirmed that the iteration count of the iterative method in the interface problem was reduced. The merit of the proposed method is that the improvement of the convergence of the ICCOCG method may make it possible to solve unsolvable problems. However, computation time was increased from several times to several ten times compared with using the double precision type. To overcome this shortcoming, we will improve it by reducing the amount of calculation and optimizing of implementations in the future work. Also, we will introduce the pseudo-quadruple precision type into the interface problem as one of the ways to optimize the analysis code.

Acknowledgement

This work was supported by JSPS Grant-in-Aid for Scientific Research (Joint International Research (Fostering Joint International Research (B)), Grant Number 18KK0278, Japan, and JST CREST Grant Number JPMJCR1181, Japan.

References

[1] A. Takei, M. Ogino, S. Sugimoto: High-frequency electromagnetic field analysis by COCR method using anatomical human body models, IEEE Transaction on Magnetics, 54:3, 2018, DOI: 10.1109/TMAG.2017.2756098.

[2] A. Takei, K. Murotani, S. Sugimoto, M. Ogino, T. Yamada: S. Yoshimura, Performance evaluation of parallel finite element electromagnetic field analysis using numerical human models, Journal of Advanced Simulation in Science and Engineering, 1:1, 2014, 127-140.

[3] A. Takei, S. Sugimoto, M. Ogino, S. Yoshimura, H. Kanayama: EMC analysis in a living environment by parallel finite element method based on the iterative domain decomposition method, Theoretical and Applied Mechanics Japan, 62, 2014, 237-245.

[4] A. Takei, S. Yoshimura, H. Kanayama: Large-scale parallel finite element analyses of high frequency electromagnetic field in commuter trains, Computer Modeling in Engineering & Sciences, 31:1, 2008, 13-24.

[5] D.H. Bailey, A C++/Fortran-90 double-double and quad-double package, [Online], Available: http://crd-legacy.lbl.gov/~dhbailey/mpdist/, Accessed on: Jan. 1, 2021.

[6] S. J. Connolly, D. Mackenzie, Y. Gorash: Higher-order and higher floating-point precision numerical approximations of finite strain elasticity moduli, Int. J. Numer. Meth. Engng., 2019, 1–19, DOI: 10.1016/j.amc.2012.03.087

[7] K. Masui, M. Ogino: Research on the Convergence of Iterative Method Using Mixed Precision Calculation Solving Complex Symmetric Linear Equation, IEEE Transaction on Magnetics, 56:1, 2020, DOI: 10.1109/TMAG.2019.2951280.

[8] A. Takei, S. Sugimoto, M. Ogino, S. Yoshimura, H. Kanayama: Full wave analyses of electromagnetic fields with an iterative domain decomposition method, IEEE Transactions on Magnetics, 46:8, 2010, 2860-2863.
A. Takei, K. Murotani, S. Sugimoto, M. Ogino, H. Kawai: High-accuracy electromagnetic field simulation using numerical human body models, *IEEE Transactions on Magnetics*, 52:3, 2015, DOI: 10.1109/TMAG.2015.2479467.

S. Yoshimura, R. Shioya, H. Noguchi, and T. Miyamura: Advanced general-purpose computational mechanics system for large scale analysis and design, *Journal of Computational and Applied Mathematics*, 149:1, 2002, 279–296.

R. Shioya, H. Kanayama, A. M. M. Mukaddes, and M. Ogino: Heat conductive analysis with balancing domain decomposition method, *Theoretical and Applied Mechanics Japan*, 52, 2004, 43–53.

Yozo Hida, Xiaoye S. Li, David H. Bailey: Quad-Double Arithmetic: Algorithms, Implementation, and Application, Technical Report LBNL-46996, University of California, Berkeley, Lawrence Berkeley National Laboratory, 2000, 1-28.

Y. Yamanaka, S. Oishi: Quadruple-precision division and square root calculation methods, *Proc. of High Performance Computing Symposium 2012*, 2012. 66. (in Japanese)

The University of Florida Sparse Matrix Collection Home Page [Online]. Available: http://www.cise.ufl.edu/research/sparse/matrices, Accessed on: Jan. 01, 2021.

M. Ogino, A. Takei, H. Notsu, S. Sugimono, S. Yoshimura: Finite element analysis of high frequency electromagnetic fields using a domain decomposition method based on the COCR method, *Theoretical and Applied Mechanics Japan*, 61, 2013, 173-181.

T. Murayama, A. Muto, A. Takei: Convergence Properties of Iterative Full-Wave Electromagnetic FEM Analyses with Node Block Preconditioners, *IEICE Transactions on Electronics*, 101-C:8, 2018, 612-619, DOI: 10.1587/transele.E101.C.612.

International Compumag Society, Testing Electromagnetic Analysis Methods (T.E.A.M.) Home Page, [Online], Available: http://www.compumag.org/jsite/team.html, Accessed on: Jan. 01, 2021.