PISCESII: 2.5D RF CAVITY CODE WITH HIGH ACCURACY

Y. Iwashita
Accelerator Laboratory, Nuclear Science Research Facility
Institute for Chemical Research, Kyoto University
Gokanosho, Uji, Kyoto 611-0011, JAPAN

Abstract

PISCES II calculates any Eigensolutions including dipole, quadrupole and so on, in any axisymmetric cavity with 2.5D Finite Element Method. Dipole and higher multipole solutions are obtained by hybrid elements. A periodic boundary condition can be handled for a long periodic structure. The accuracy of the frequency in a solution obtained from PISCES II is improved by use of higher order elements. The improvement on the Eigenvalue solver reduces the computation time.

1 INTRODUCTION

An RF cavity code with high accuracy is necessary for design with high precision. There are still some needs of an axisymmetric cavity because of its simplicity.

PISCES II can calculate all Eigensolutions in a cavity with axisymmetric boundaries including multipole modes[1,2]. Periodic boundary conditions can also be handled in this code.

2 FORMULATION

The differential equation for electric field $\vec{E}$ or magnetic field $\vec{H}$ to be solved are [3,4],

$$\nabla\times\nabla\times\vec{E}+k^2\vec{E}=0,\quad \nabla\cdot\vec{E}=0\quad (\text{in } \Omega), \tag{1}$$

or, $$\nabla\times\nabla\times\vec{H}+k^2\vec{H}=0,\quad \nabla\cdot\vec{H}=0\quad (\text{in } \Omega), \tag{2}$$

where $k^2=\omega^2\varepsilon\mu$ and Ω is the entire volume. In vacuum space, $k^2=\omega^2/c^2$, where c is the speed of light. Boundary conditions are

$$\vec{E}\times\vec{n}=0 \quad \text{or} \quad \vec{H}\cdot\vec{n}=0 \tag{3}$$

on electric boundaries (Γe) for metal surfaces,

$$\vec{E}\cdot\vec{n}=0 \quad \text{or} \quad \vec{H}\times\vec{n}=0 \tag{4}$$

on magnetic boundaries (Γm) for symmetry plane, and

$$\boldsymbol{E}_{left}=e^{i\varphi}\boldsymbol{E}_{right} \quad \text{or} \quad \boldsymbol{H}_{left}=e^{i\varphi}\boldsymbol{H}_{right} \tag{5}$$

on periodic boundaries (Γp), where $\vec{n}$ denotes the outward normal on the boundary and φ is the phase advance in the cell.[5] The periodic boundary is not a real boundary but only for a convenience in defining a problem. Because either $\vec{E}$ or $\vec{H}$ can be used as the field variable, only the electric field will be shown hereafter. Integrating Eq.(1) over Ω after multiplying by $\delta\vec{E}$ (virtual electric field), we get

$$\int_\Omega \delta\vec{E}\cdot\nabla\times\nabla\times\vec{E}dv=-k^2\int_\Omega\delta\vec{E}\cdot\vec{E}dv. \tag{6}$$

using vector relation and applying Gauss's theorem, the following relations must hold for any $\delta\vec{E}$:

$$\int_\Gamma\left(\nabla\times\vec{E}\right)\times\delta\vec{E}\cdot d\vec{S}-\int_\Omega\left(\nabla\times\vec{E}\right)\cdot\left(\nabla\times\delta\vec{E}\right)\,dv =-k^2\int_\Omega\delta\vec{E}\cdot\vec{E}dv \tag{7}$$

$$\vec{E}\times\vec{n}=0 \text{ and } \delta\vec{E}\times\vec{n}=0 \text{ on } (\Gamma e), \tag{8}$$

$$\vec{E}\cdot\vec{n}=0 \text{ and } \delta\vec{E}\cdot\vec{n}=0 \text{ on } (\Gamma m), \tag{9}$$

The term in the surface integration of Eq. (7) becomes zero on either (Γe) or (Γm) because of the boundary condition of Equ's. (8) or (9).

3 FINITE ELEMENT MODEL

Because only the problems on axisymmetric domains are considered, we can assume sin $m\theta$ and cos $m\theta$ dependencies for E_r, E_Z and E_θ components, and then the problem can be reduced to two-dimensional problem:

$$\vec{E}=(E_\theta \sin m\theta, Er\cos m\theta, Ez\cos m\theta). \tag{10}$$

Then (E_θ, E_r, E_Z) are functions of r and z only. The field variables are (rE_θ, E_r, E_Z) for $m\geq1$ and (E_θ , H_θ) for m=0.

Only the triangular element is used in PISCES II. The shape functions used are the conventional simple linear one for the θ component and two dimensional Nedelec elements[3,6,7,8] for the E_r, E_Z components. (See Fig. 1) Only the tangential component of E_r, E_Z is assigned on the line, and is constant along the line. Then $\vec{E}$ and $\nabla\times\vec{E}$ can be written as

$$\vec{E}=\begin{bmatrix}\mathrm{E}_\theta\\ \mathrm{E}_\mathrm{r}\\ \mathrm{E}_\mathrm{z}\end{bmatrix}=\vec{N}\cdot\begin{bmatrix}r\vec{E}_\theta\\ \vec{E}_{rz}\end{bmatrix}=\begin{bmatrix}\frac{1}{r}\vec{N}_\theta & 0\\ 0 & \vec{N}_r\\ 0 & \vec{N}_z\end{bmatrix}\cdot\begin{bmatrix}r\vec{E}_\theta\\ \vec{E}_{rz}\end{bmatrix}, \text{ and} \tag{11}$$

$$\nabla\times\vec{E}=\vec{N}'\cdot\begin{bmatrix}r\vec{E}_\theta\\ \vec{E}_{rz}\end{bmatrix}=\begin{bmatrix}0 & \partial_z\vec{N}_r-\partial_r\vec{N}_z\\ \frac{-1}{r}\partial_z\vec{N}_\theta & \frac{1}{r}\partial_\theta\vec{N}_z\\ \frac{1}{r}\partial_r\vec{N}_\theta & \frac{-1}{r}\partial_\theta\vec{N}_r\end{bmatrix}\cdot\begin{bmatrix}r\vec{E}_\theta\\ \vec{E}_{rz}\end{bmatrix} =\begin{bmatrix}0 & \partial_z\vec{N}_r-\partial_r\vec{N}_z\\ \frac{-1}{r}\partial_z\vec{N}_\theta & \frac{-m}{r}\vec{N}_z\\ \frac{1}{r}\partial_r\vec{N}_\theta & \frac{m}{r}\vec{N}_r\end{bmatrix}\cdot\begin{bmatrix}r\vec{E}_\theta\\ \vec{E}_{rz}\end{bmatrix}, \tag{12}$$

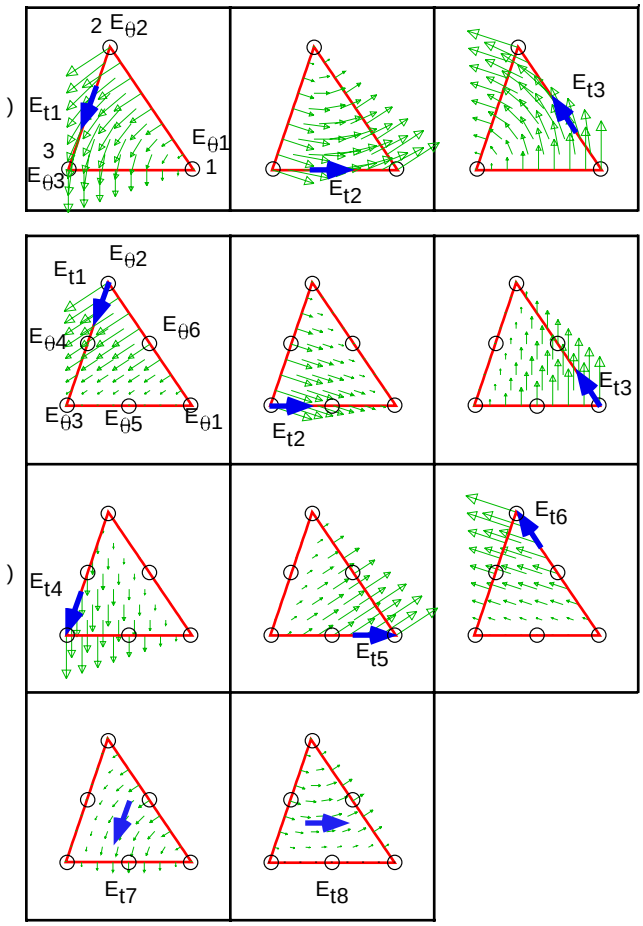

Fig. 1. Hybrid triangular elements.

a) Constant edge and linear nodal elements.

b) Linear edge and quadratic elements. The bottom two shape functions represent the internal freedoms.

TABLE I
SHAPE FUNCTIONS

The lowest order (Constant edge and linear nodal):

$$\vec{N}_\theta = [u \quad v \quad w]$$

$$\vec{N}_r = \frac{1}{2A_e}\left[l_1(v\partial_r w - w\partial_r v) \quad l_2(w\partial_r u - u\partial_r w) \quad l_3(u\partial_r v - v\partial_r u)\right]$$

$$\vec{N}_z = \frac{1}{2A_e}\left[l_1(v\partial_z w - w\partial_z v) \quad l_2(w\partial_z u - u\partial_z w) \quad l_3(u\partial_z v - v\partial_z u)\right]$$

The second order (Linear edge and quadratic nodal):

$$\vec{N}_\theta = [u(2u-1) \quad v(2v-1) \quad w(2w-1) \quad 4vw \quad 4wu \quad 4uv]$$

$$\vec{N}_r = \frac{1}{2A_e}[l_1 v\partial_r w \quad l_2 w\partial_r u \quad l_3 u\partial_r v \quad -l_1 w\partial_r v \quad -l_2 u\partial_r w \quad -l_3 v\partial_r u \quad l_1 u(v\partial_r w - w\partial_r v) \quad l_2 v(w\partial_r u - u\partial_r w)]$$

$$\vec{N}_z = \frac{1}{2A_e}[l_1 v\partial_z w \quad l_2 w\partial_z u \quad l_3 u\partial_z v \quad -l_1 w\partial_z v \quad -l_2 u\partial_z w \quad -l_3 v\partial_z u \quad l_1 u(v\partial_z w - w\partial_z v) \quad l_2 v(w\partial_z u - u\partial_z w)]$$

where u, v and w are the area coordinates and l_i is the length of the edge facing to the vertex i.

where $\vec{N}$, $\vec{N}'$, $\vec{N}_\theta$, $\vec{N}_z$, and $\vec{N}_r$ are the shape functions, and $r\vec{E}_\theta$ and $\vec{E}_{rz}$ are the field variable. The base shape functions in (u-v) plane are given in Table I. The last two functions in $\vec{N}_z$, and $\vec{N}_r$ represent the internal freedoms and are omitted in twelve-parameter shape function. The element matrix equation is

$$\int_e \vec{N}'^{\mathrm{T}} \cdot \vec{N}' r dr dz = -k^2 \int_e \vec{N}^{\mathrm{T}} \cdot \vec{N} r dr dz, \tag{13}$$

where symbol e is the element volume and T denotes matrix transpose. The integrations are performed numerically up to 11th order precision. By assembling all element matrices and applying the boundary condition, finally we get the general Eigenvalue equation.

$$\vec{K} \cdot \vec{x} = k^2 \vec{M} \cdot \vec{x}, \tag{14}$$

where $\vec{K}$ and $\vec{M}$ are large sparse symmetric matrices, and $\vec{x}$ is an Eigenvector for the field variables. Usually several Eigensolutions starting from the smallest one but zero are of interest. Unfortunately, this Eigenvalue problem has many zero-Eigenvalue solutions, which correspond to the spurious modes, and thus special care should be taken. For axisymmetric solutions, such as TM_{0xx} or TE_{0xx}, the problem can be expressed by field variable of E_θ and/or H_θ, which has no zero Eigensolution.

3 PERIODIC BOUNDARY

The periodic boundary condition is implemented using Floquet's theorem[5]. Because $\vec{x}$ has to be a complex vector in this analysis, Eq.(14) can be rewritten splitting into real part and imaginary part:

$$\begin{pmatrix} \vec{x}_{right}^{Re} \\ \vec{x}_{right}^{Im} \end{pmatrix} = \begin{pmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{pmatrix} \cdot \begin{pmatrix} \vec{x}_{left}^{Re} \\ \vec{x}_{left}^{Im} \end{pmatrix} \text{ and} \tag{15}$$

$$\begin{pmatrix} \boldsymbol{K} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{K} \end{pmatrix} \cdot \begin{pmatrix} \vec{x}^{Re} \\ \vec{x}^{Im} \end{pmatrix} = k^2 \begin{pmatrix} \boldsymbol{M} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{M} \end{pmatrix} \cdot \begin{pmatrix} \vec{x}^{Re} \\ \vec{x}^{Im} \end{pmatrix}. \tag{16}$$

Only the boundary condition Eq. (3) and (4) are included in Eq(16) at this point. Reordering $\vec{x}$, and using Eq.(15), we get:

$$\vec{x} = \boldsymbol{P} \cdot \vec{x}', \tag{17}$$

where

$$\boldsymbol{P} \equiv \left(\begin{array}{cc:cc} \boldsymbol{1} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{1} & \boldsymbol{0} & \boldsymbol{0} \\ \cos\varphi & \boldsymbol{0} & -\sin\varphi & \boldsymbol{0} \\ \hdashline \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{1} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{1} \\ \sin\varphi & \boldsymbol{0} & \cos\varphi & \boldsymbol{0} \end{array}\right), \quad \vec{x} \equiv \begin{pmatrix} \vec{x}_{left}^{Re} \\ \vec{x}_{inner}^{Re} \\ \vec{x}_{right}^{Re} \\ \hdashline \vec{x}_{left}^{Im} \\ \vec{x}_{inner}^{Im} \\ \vec{x}_{right}^{Im} \end{pmatrix} \text{ and } \vec{x}' \equiv \begin{pmatrix} \vec{x}_{left}^{Re} \\ \vec{x}_{inner}^{Re} \\ \hdashline \vec{x}_{left}^{Im} \\ \vec{x}_{inner}^{Im} \end{pmatrix}. \tag{18}$$

Applying P^{T} from left side and using Eq. (18), Eq.(16) becomes:

$$\boldsymbol{P}^{\mathrm{T}}\cdot\begin{pmatrix}\boldsymbol{K} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{K}\end{pmatrix}\cdot\boldsymbol{P}\cdot\vec{x}'=k^2\boldsymbol{P}^{\mathrm{T}}\cdot\begin{pmatrix}\boldsymbol{M} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{M}\end{pmatrix}\cdot\boldsymbol{P}\cdot\vec{x}'. \qquad (19)$$

This Eigenvalue problem is real and symmetric, the same Eigenvalue solver can be used.

4 GENERAL EIGENVALUE SOLVER

Because Eqs. (14) and (19) are general Eigenvalue problems for large sparse symmetric matrix with many zero Eigenvalue solutions, special care had to be taken. The solver is based on the subspace method[9] and uses zero and upper filtering technique[10].

The zero filtering technique requires a rough estimate of the lowest Eigenfrequency(FLO), which usually can be obtained by the physical dimension of the problem. If FLO is not given by a user and solution with the second order element or multiple solutions are required, PISCES II evaluates FLO from a single mode solution with lowest order element starting with the "guess" value from the physical dimension.

As similar technique, upper filter technique is also applied. If the upper frequency (FUP) is not given by a user, PISCES II starts to use the highest Eigenvalue in the subspace as FUP after the value settled. In this case the FUP value is adjusted adaptively.

Because the method is based on the iterative method, initial vectors should be given at the start. For a problem with the lowest order element, the initial vectors are set from random numbers. Before starting with the second order element, the solution vectors are obtained from the lowest order element and "prolonged" to the second order solutions by linear interpolation. Because fairly good initial solutions are obtained by this way and FUP is also available from the beginning, the solution time is reduced about half comparing with direct start with random vectors.

5 INPUTS TO PISCES II

A mesh data example for a sphere as shown in Fig. 2 is listed in Fig. 3. All the internal units are in SI. The first line contains a title with less than 80 characters. The second line is problem options where only the unit scale for cm is specified in the example. The options are explained later. The third line has three numbers, which are the number of nodes, the number of elements and the number of boundary points plus one for closed line. After this line, node coordinates are specified being enclosed by brackets. Followings are the element data, which specifies the three coordinates of the vertices by the sequential numbers of the coordinate sets. The fourth position specifies material number other than vacuum. Then the boundary points are specified in the same way as above. Each boundary condition for a segment between n and n+1 boundary points is specified by a character constant. The encodings of the boundary conditions are listed in Table 1. The last group is the curvature data which specifies the radii of the segments in the same sequence as the boundary conditions. Zero in radius data means the straight line. Each group is read by free format READ statement. The mid-line points will be generated in the code if the second order elements are needed.

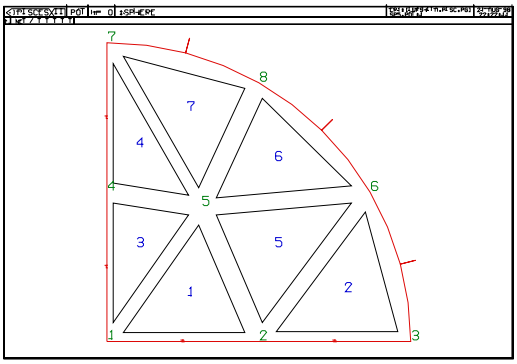

Fig. 2 Mesh example for a sphere.

```
:SPHERE
 &PROBLEM   UNIT=0.01 &END
      8      7      8
(0,0)  (5,0)  (10,0)  (0,5)  (3.1,4.5)  (8.66025,5)  (0,10)
(5,8.66025)
1,2,5,,2,3,6,,4,1,5,,4,5,7,,5,2,6,,5,6,8,,7,5,8,,
   1,   2,   3,   6,   8,   7,   4,   1,
'A' 'A' 'E' 'E' 'E' 'E' 'E' 'A'
0 0 10 10 10 0 0  0
```

Fig. 3 Mesh data for a sphere with radius of 10 cm.

Table 1: The encoding of the boundary conditions

Code	Boundary condition
'A'	on axis (r=0)
'E"	on electric boundary (metal surface)
'M'	on magnetic boundary (symmetry plane)
'P'	left side boundary for periodic boundary
'Q'	right side boundary for periodic boundary

MESHNET program[11] can convert a TAPE35 data that is generated by LATTICE[12] to an input file for PISCES II. Because the curvature information is not included in TAPE35 data, the curvature data has to be added by hand or by NETREF program. In order to add a curvature data for an arc, one should specify only starting and ending coordinates of the arc that may include edges of multiple elements.

Table 2 explains the problem constants. OPTNF can be used to change the problem constants without modifying the input data files that usually have very large sizes. The rotational symmetry parameter m and the phase shift parameter can be incremented repeatedly in the code by specifying appropriate values. BCLR overrides boundary conditions: 'ME' overrides left boundary to magnetic and right boundary to electric. If a blank appears, the corresponding boundary is not modified. All the unit in

the code is SI unit. If the unit in length is 'cm', used 0.01 for UNIT parameter. If CFV flag is false, all the curvature data are ignored. As mentioned above, either electric field 'E' or magnetic field 'H' can be used to obtain Eigenmodes except axisymmetric modes: FIELD parameter is used to choose 'E' or 'H'. The digit of one $\mathrm{LMTYP}_{(1)}$ in LMTYP specifies the order of elements used in the calculation if applicable: first and second order nodal elements, which may be selected by '1' and '2', respectively, are not used for $m>1$. If the first order elements are specified, both 'E' and 'H' components are calculated at the same time to give both TM0xx and TE0xx Eigenmodes, and the FIELD parameter is meaningless in the calculation. The digit of ten $\mathrm{LMTYP}_{(10)}$ in LMTYP is used to specify options for edge elements: it is ignored in nodal elements ($m=0$). If $\mathrm{LMTYP}_{(10)}$ is zero, simple constant edge and linear nodal hybrid elements are used. If $\mathrm{LMTYP}_{(10)}$ is 3 or 4, linear edge and quadratic nodal hybrid elements with 12 or 14 parameters are used, respectively. $\mathrm{LMTYP}_{(10)}$ of 7 or 8 gives the same results as 3 or 4 but they are hierarchical elements and faster in convergence. NMODE specifies the number of modes solved in a calculation. A few more modes are included in an Eigenvalue calculation to give better convergence. FLO or FUP may be specified if the lowest or highest frequencies are known, respectively. If higher order elements are selected, these values and the initial vectors are obtained from lower order calculation if MGF is true. Among many options for a solver for linear system, PCGM (Preconditioned Conjugate Gradient Method) with SOR (MCG=2) seems the best choice.

Table 2: The program constants

name	Default	description
OPTNF	T	The problem constants will be read again
EM	1.0	(starting) *m* in Eq. (8)
MSTEP	1	increment of m
MNUM	1	iterations for m
PHASE	0.0	(starting) phase shift
PHSTEP	0.0	increment of phase
NUMPH	1	iterations for phase shift
BCLR	''	overrides boundary condition (2 chars.)
UNIT	1.	unit scale (metric)
CVF	T	enable curved boundary
FIELD	'E'	field variable E or H
LMTYP	1	element type
NMODE	5	modes to be solved
FUP	0.0	highest frequency
FLO	0.0	lowest frequency
EPS	1E-5	accuracy
ITMX	200	iterations for subspace
MGF	T	enable presolution
MCG	2	method of CGM
INFILE	''	file for initial value
OTFILE	''	output file

6 EXAMPLE

Figs. 4 and 5 show the relative frequency errors and CPU time as functions of the number of unknowns for the hemisphere problem shown in Fig.2. The lowest order solutions need less CPU time while they have less accuracy. Table 3 shows the calculated results and analytical solution of Q-values of the hemi-sphere with r=10cm. They are consistent with the analytical results of Qanalytical=40371.

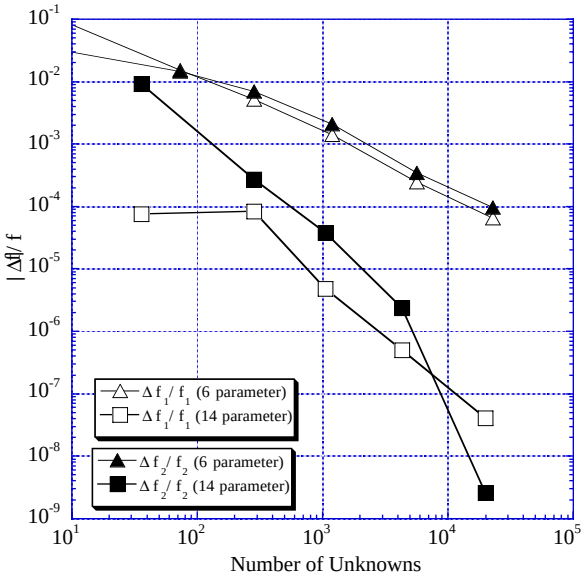

Fig. 4 Relative frequency errors of the second and the third lowest modes in a spherical cavity as a function of the number of unknowns.

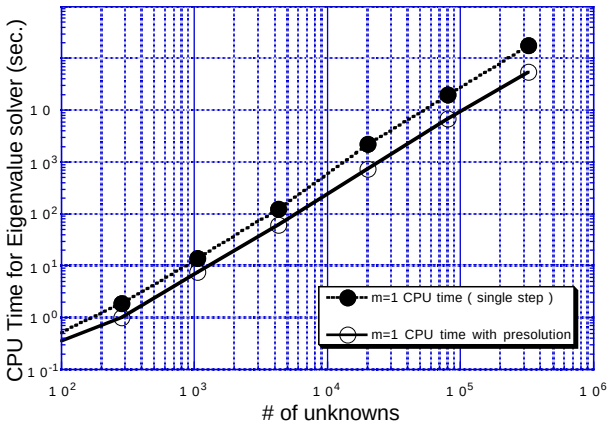

Fig. 5 CPU times as functions of the number of unknowns.

Table 3: Q-values for dipoles in hemi-sphere

mesh size	E as field variables	H as field variables
.05	40227	40230
.025	40263	40248
.0125	40261	40251

More practical example is shown in Fig. 6 [13]: contour plots for theta components and arrow plots for r-z components of first two dipole modes from lowest frequency. The measured values are listed in Table 4. The values are calibrated to those at 45 °C in vacuum. The effect of skin depth on the frequency measurement is included based on df/f=1/2Q formula to obtain a loss-less value as listed in the row "Meas.+Q.corr", where the Q values are obtained from the theoretical calculation. They have very good agreement

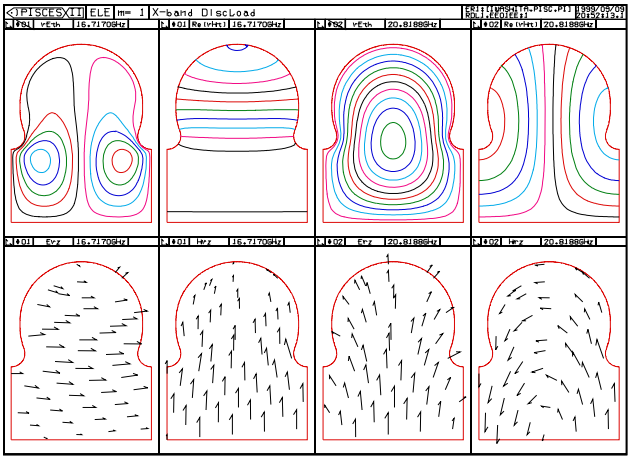

Fig. 6 Field plots for a DDS cell.

7 CONCLUDING REMARKS

After the improvement of initial vectors and parameters, the dipole problem with 324388 node-points (including mid-line points generated in the code) takes 11.6 hours for seven Eigensolutions. The most time consuming portion is the matrix solver. PCGM (Preconditioned Conjugate Gradient Method) with SOR as a preconditioner is currently used. More efficient preconditioner will reduce the CPU time. Although PISCES-II is somewhat time consuming for very fine mesh calculation, reasonable precision results can be obtained in moderate CPU time without very fine mesh.

8 REFERENCES

[1] E. M. Nelson, "A finite element field solver for dipole modes", 1992 Linear Accelerator Conference Proceedings, Ottawa, Ontario, Canada, AECL-10728, Vol2, pp.814-816, August 1992

[2] M. Koshiba, S. Maruyama and K. Hirayama, "A vector finite element method with the high-order mixed-interpolation-type triangular elements for optical waveguiding problems", Journal of Lightwave Technology, Vol.12, No.3, March 1994, pp.495-502.

[3] M. Hara, T. Wada, T.Fukasawa, and F. Kukuchi, "A three Dimensional Analysis of RF Electro-magnetic Fields by Finite Element Method", IEEE Trans., **MAG-19** No. 6 Nov. 1983

[4] K. H. Huebner and E. A. Thornton, " The Finite Element Method for Engineers", (J.Weiley, New York); and A.R.Mitchell and R.Wait, "The Finite Element Method in Partial Differential Equations" (J.Weiley, New York, 1977);

[5] R. L. Gluckstern and E. N. Opp, "Calculation of dispersion curves in periodic structures", IEEE Trans. **MAG-21** No. 6 Nov. 85 pp. 2344-2346

[6] J. C. Nedelec, "Mixed Finite Elements in R^3", Num. Math., Vol. 35 pp.315-341, 1980

[7] J. C. Nedelec, "A New family of Mixed Finite Elements in R3", Num. Math., Vol. 50 pp. 57-81, 1986

[8] Iwashita Y: Accuracy of Eigenvalue Obtained with Hybrid Elements on Axisymmetric Domains, IEEE Trans. **MAG-34** No.5 pp.2543-2546 (1998)

[9] K.J. Bathe, "Solution Methods for Large Generalized Eigenvalue Problem in Structural Engineering", Doctoral thesis, University of California, Berkeley, 1971

[10] Y. Iwashita, "General Eigenvalue Solver with Zero and Upper Filters for Large Sparse Symmetric Matrix", Proc. of the Eighth Biennial IEEE Conference on Electromagnetic Field Computation CEFC'98, June-3,1998, Tucson, Arizona, p.253

[11] Y. Iwashita, "PISCES II:2.5D RF Cavity Code", Computational Accelerator Physics, Williamsburg, VA, AIP conference proceedings No. 361 Sept. 1996, pp.119-124

[12] "User's Guide for the POISSON/SUPERFISH Group of Codes", LA-UR-87-115, Los Alamos National Lab.

[13] "Confirmation of 2D contour of RDDS1 disks", T. Higo, T. Suzuki and N. Toge, Y. Funahashi, Y. Higashi, N. Hitomi, T. Takatomi, Y. Watanabe, J. Wang, Z. Li and R. Miller, to be published in LCC NOTE.

Table 4: Frequencies calculated and measured. The values in the line "Meas.-Cal." are the difference between measurement and calculation with the finite conductivity correction included. Units are in MHz.

	F0-0	F0-2π/3	F0-π	Fd1-π	Fd2-0
Measurement	10920.80 ±0.1	11519.40 ±0.1	11715.60 ±0.1	15253.30 ±0.2	16709.7 ±0.3
Meas.+Q.corr	10921.49	11520.13	11716.35	15254.48	16710.47
PISCES-II Q	7960	7870	7840	6490	10840
Cal.	10921.20	11519.66	11716.19	15254.12	16710.08
Meas. – Cal.	**0.29**	**0.47**	**0.16**	**0.36**	**0.39**