

11

LA-UR- 97-4080

CONF-9609245--

Approved for public release;
distribution is unlimited.

Title:

CAVITY DESIGN PROGRAMS

RECEIVED

FEB 02 1998

OSTI

Author(s):

Eric Nelson, LANSCE-9

DTIC QUALITY INSPECTED 2

Submitted to:

Joint US/CERN/Japan Particle Accelerator
School

September 9-17, 1996

19980507 010

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

ph

MASTER

Los Alamos
NATIONAL LABORATORY

Los Alamos National Laboratory, an affirmative action/equal opportunity employer, is operated by the University of California for the U.S. Department of Energy under contract W-7405-ENG-36. By acceptance of this article, the publisher recognizes that the U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or to allow others to do so, for U.S. Government purposes. Los Alamos National Laboratory requests that the publisher identify this article as work performed under the auspices of the U.S. Department of Energy. The Los Alamos National Laboratory strongly supports academic freedom and a researcher's right to publish; as an institution, however, the Laboratory does not endorse the viewpoint of a publication or guarantee its technical correctness.

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

CAVITY DESIGN PROGRAMS

E.M. NELSON

Los Alamos National Laboratory, Los Alamos, NM 87545, USA

Numerous computer programs are available to help accelerator physicists and engineers model and design accelerator cavities and other microwave components. This article discusses the problems these programs solve and the principles upon which these programs are based. Some examples of how these programs are used in the design of accelerator cavities are also given.

1 Applications of Cavity Design Codes

There are many design and modeling problems concerning accelerator cavities and associated microwave and RF components. These problems are often solved with the help of cavity design codes. The calculations commonly performed by these codes can be assigned to one of three categories: time domain calculations, frequency domain calculations and eigenmode calculations. Each type of calculation is important to the design and modeling of accelerators. No single type of calculation conveniently solves all problems.

The characteristics and limitations of each type of calculation are described in the following subsections, starting with time domain calculations. Examples of cavity design problems which are conveniently treated by each type of calculation are given in the same subsections. The details concerning how the calculations are performed are deferred to the following section on discretization methods.

The cavity design problems described below are focused on computing electromagnetic fields. These fields may be affected by, or entirely due to, the presence of charged particle beams. However, in some problems the trajectories of these particles are significantly affected by the electromagnetic field. The output cavity of a klystron is one example of such a problem. A self-consistent solution is desired. This typically requires simultaneous calculation of fields and particles, and particle-in-cell (PIC) codes are often employed for this purpose. However, such problems are not considered in this article. In the examples below, the trajectory of particles in a beam which drives fields in a cavity will be known a priori. The effect of the computed fields on these particle trajectories is assumed to be insignificant.

1.1 Time Domain Calculations

A common task in the design and modeling of accelerators is to compute the wakefields in a structure. Wakefields are the electromagnetic fields left in a structure due to the passage of a charged particle beam. These wakefields can have a detrimental effect on the particle beam. Time domain calculations are often employed to compute wakefields because the wakefields are complicated functions of time. In many situations, computation of the wakefields via the other two types of calculations is not convenient or not practical. After the wakefields have been computed by a time domain calculation, then a separate beam dynamics calculation computes the effect of the wakefields on the particle beam. If the result is not satisfactory then the beam parameters or the accelerator structure are changed, and the computational process is repeated.

Time domain calculations compute electromagnetic fields as a function of time. The electromagnetic fields in these calculations are typically driven by a source. The source is an input waveguide or a charged particle beam. Some calculations start with an initial electromagnetic field, although the typical calculation starts with zero electromagnetic fields. Time domain calculations typically model the transient fields and usually stop when the steady state has been reached.

The basic scheme for a time domain calculation starts with the electromagnetic field at time t_i stored in the computer's memory. The subscript i indicates this is the i th time step. The program uses the fields and the specified sources at time t_i to compute the fields at the next time step, $t_{i+1} = t_i + \Delta t$. This process repeats to compute the field at time t_{i+2} , and then t_{i+3} , and so on. This process is described in more detail in the section on discretization methods. Typically the user instructs the computer program to compute the fields for the duration of the transient phenomena.

Time domain calculations can generate an enormous amount of physically meaningful information because they compute the field at every time step. There is more information than can fit in the computer's memory, or even on the computer's disk. There is often more information computed than one person can look at in a lifetime. This property of time domain calculations affects how they are employed compared to the other two types of calculations. In time domain calculations, one does not save all of the information and then decide what to look at later. Instead, one chooses a small fraction of the available information to save before performing the calculation. In other words, the computed field data must be reduced, and the way it is reduced must be decided before running the calculation.

One common way to reduce the output data is to save the field data at

only a few selected time steps. The field data at a single time step is called a snapshot. Another common way to reduce the output data is to save the field at a particular point in the cavity at all (or very many) time steps. This type of output is called a time history. Other quantities like field energy and mode amplitudes can be computed at each time step and saved as a time history.

Time domain calculations replace old field data (i.e., at a time step t_i) with new field data (e.g., at time step t_{i+1} or t_{i+2}) in order to avoid rapidly consuming all of the computer's memory. The computer memory occupied by the field data must be reused. Time domain calculations typically replace old field data with new field data every one or two time steps. Thus, much "post-processing" and reduction of the time-dependent field data takes place during the calculation while the fields are still available in memory.

Some forms of data reduction, like the computation of wake potentials, require careful arrangement of the computational order so that the calculations can be performed efficiently. Other forms of data reduction may not be feasible. For example, convolving the fields with a response function (e.g., the response of a dispersive dielectric media) is in general not feasible because it would require too much field data in computer memory. If such convolutions are necessary, they must be recast as running sums which can only be performed for special forms of response functions.

The fields are typically driven by one of two sources: particle beams and waveguide ports. Particle beams are commonly used in wakefield calculations. A specified particle beam travels through an accelerator cavity, driving fields in the cavity. These fields are integrated along the trajectory of a witness particle to give the wake potential. Another example is an input coupler for a travelling wave accelerator structure. The fields in the structure are driven by a wave entering the structure at the waveguide port. One might look at maximum surface field near the input coupler during the rise of the RF pulse. This could have an effect on dark current in the structure.

Of the three types of calculations, time domain calculations seem to have the greatest flexibility. Time domain calculations can be substituted for the other two types of calculations. Applications of time domain codes to the other types of calculations are given in the following subsections.

1.2 Frequency Domain Calculations

Accelerator cavities in storage rings and accelerator structures in linear accelerators are typically driven at some specified frequency ω . There are transients when the RF or microwave source is turned on, but after the transients decay there is a steady-state driven field which accelerates the charged particle beam.

Many issues like field asymmetries experienced by the beam due to asymmetric input couplers, or peak electric and magnetic fields near input couplers, can be addressed by looking at steady-state driven fields.

Steady-state driven fields can be computed by frequency domain calculations. Frequency domain calculations compute the steady-state fields directly without computing any transient fields. The sources and fields are assumed to have a time dependence $e^{j\omega t}$ where the frequency ω is specified by the user. The results are complex (phasor) vector fields and often scattering parameters (e.g., transmission and reflection coefficients). Each component of the complex vector field has its own phase (or argument) which may be different from the phases of the other components. A complex vector field $\mathbf{E}(\mathbf{x})$ computed by a frequency domain calculation can be interpreted as a real time dependent vector field $\mathbf{E}(\mathbf{x}, t)$,

$$\mathbf{E}(\mathbf{x}, t) = \Re \{ \mathbf{E}(\mathbf{x}) e^{j\omega t} \}. \quad (1)$$

Field plots are often limited to real field quantities. An arrow plot of the real part of $\mathbf{E}(\mathbf{x})$ is a snapshot of the driven steady-state field at phase $\phi = \omega t = 0$, while a plot of the imaginary part of $\mathbf{E}(\mathbf{x})$ is a snapshot of the driven steady-state field one quarter of a cycle earlier, at phase $\phi = \omega t = -\pi/2$. These two plots give enough information to represent the steady-state field at all other times. Sometimes it is difficult to mentally keep track of three field components and their phases simultaneously, in which case additional field plots of quantities like $\Re \{ \mathbf{E}(\mathbf{x}) e^{\pm j\pi/4} \}$ can help one visualize the time evolution of the steady state fields.

Contour plots of quantities like $|\Re \mathbf{E}|^2 = (\Re E_x)^2 + (\Re E_y)^2 + (\Re E_z)^2$ give the field magnitude at a particular phase, in this case $\phi = 0$. A contour plot of $|\mathbf{E}|^2 = |E_x|^2 + |E_y|^2 + |E_z|^2$ gives twice the time averaged magnitude of the field \mathbf{E} , but care should be exercised when interpreting this result as the magnitude of the field at some particular phase. Different components can have different phases, so the time-dependent field $\mathbf{E}(\mathbf{x}, t)$ might not ever reach its “magnitude” $|\mathbf{E}|^2$. However, in some cases one may know that phases of all three field components are the same. For example, the electric field normal to a metal wall has a single phase at each point on the wall, so locating the peak $|\mathbf{E}|^2$ on the wall is the same as finding the location of the largest time dependent steady state electric field $\mathbf{E}(\mathbf{x}, t)$ on the wall.

Waveguide ports are the most common source in frequency domain calculations. One can run many simulations at different frequencies to get the reflection coefficient versus frequency. One can also look at the fields to find the peak steady-state field at an operating frequency f_o . Harmonic current sources are also used in frequency domain calculations. They might be used

to compute the impedance of klystron cavities or storage ring beamline components.

Time domain codes can be used for frequency domain calculations. To obtain the steady state fields, run the time domain simulation until the transients are negligible. Two snapshots 90° apart in phase (time) will give the real and imaginary components of the field. To find the scattering parameters versus frequency one can perform a time domain simulation with an input pulse of modest bandwidth centered on the frequency region of interest. Run the time domain simulation until the fields are negligible. Fourier transform the input mode amplitude and the response (transmitted or reflected) mode amplitude. Dividing these two amplitudes will give the scattering parameters as a function of frequency.

A word of caution is warranted for these calculations. Many waveguide port boundary conditions in time domain simulations are only good for a limited bandwidth. Hence the drive fields in the simulations should not have very fast rise or fall times.

Another feature of frequency domain calculations is that they can deal effectively with dispersive materials. These are materials whose permittivity or permeability depend on frequency. Time domain calculations can have some difficulty modeling dispersive materials. Only special forms of dispersion (i.e., dependence on frequency) can be handled effectively in time domain calculations¹.

1.3 Eigenmode Calculations

One of the first steps in designing an accelerator cavity is to choose the cavity parameters so that the accelerating mode has the desired operating frequency. Accelerator cavities have a high quality factor and a correspondingly small coupling (or high external quality factor). The input coupler has only a modest effect on the frequency of the accelerating mode. In such situations it is convenient to neglect the input coupler and simply compute the natural resonant frequencies and fields of modes in the accelerator cavity. These are eigenfrequencies and eigenmodes, respectively.

An eigenmode calculation computes the eigenfrequencies ω_i and corresponding eigenmodes (\mathbf{E}_i , for example) of a cavity or structure. The index i is a mode index. Cavities have more than one eigenmode and eigenfrequency, although in many cases only one mode, the accelerating mode, is desired. Typical eigenmode calculations compute a few of the lowest frequency modes of a cavity.

An eigenmode is a solution to the homogeneous, or source-free, set of

Maxwell's equations. An eigenmode field can exist in the cavity without any sources, and this eigenmode field oscillates at the eigenmode's resonant frequency. In reality there is a little bit of damping in the cavity, so the resonant frequency has a small imaginary part which indicates the field damps after many periods of oscillation. This damping is often neglected in the eigenmode calculation because it is a small effect on the frequency and fields of the mode.

The eigenmodes form an orthogonal basis which is sometimes employed in expansions of driven fields. For example, a steady-state field might be written as

$$\mathbf{E}(\mathbf{x}) = \sum_i \alpha_i \mathbf{E}_i(\mathbf{x}) \quad (2)$$

where the α_i are complex mode coefficients which depend on how the fields in the cavity are driven. In high Q cavities and single mode waveguides a field expansion with a single eigenmode is often employed.

The output of an eigenmode calculation are the frequencies and fields of the modes. If the cavity model is lossy then the frequencies may be complex. For lossy cavity models and periodic structures the fields may be complex (phasor) vector fields. Visualization of complex vector fields can be handled just like results from frequency domain calculations.

Some eigenmode calculations for waveguides or periodic structures will compute a propagation constant k_z instead of a resonant frequency ω . This is especially appropriate for waveguide with dispersive materials, where the material properties depend on frequency ω . If one were to compute ω given k_z , then the material properties would not be known a priori. By solving for k_z given ω , one knows the values of permittivity and permeability in the dispersive material at the beginning of the calculation. Similarly, cavities with dispersive material are a challenge for resonant frequency eigenmode calculations because one often does not know the frequencies (and thus the material properties) a priori.

Time domain calculations can be employed for computing eigenmodes. In the time domain calculation, initialize the field randomly and let it evolve over time for many cycles of the desired frequencies. Record the time history of a field component which is expected to be non-zero for the desired modes. Multiple time histories of all components at various locations may be required if multiple modes whose properties are not well known are desired. A fourier transform of the time histories yields the mode frequencies. One can drive the fields at a resonant frequency to obtain the eigenmode fields. Alternatively, one can perform a fourier transform of the source free fields at the resonant frequency in order to obtain the eigenmode fields.

2 Discretization Methods

2.1 Finite Difference Methods

As an introduction to finite difference methods, consider a two dimensional eigenmode problem for the cutoff frequency ω_c of the TM modes in a homogeneous waveguide. The partial differential equation to solve is

$$\nabla^2 E_z + \gamma^2 E_z = 0 \quad \text{in } \Omega \quad (3)$$

with boundary condition

$$E_z = 0 \quad \text{on } \partial\Omega. \quad (4)$$

The domain Ω is the waveguide cross section, and its boundary $\partial\Omega$ is assumed to be perfectly conducting metal. The eigenvalue is $\gamma^2 = \epsilon\mu\omega_c^2$. From γ^2 one can easily compute the cutoff frequency ω_c .

Now consider a uniform rectangular grid covering the domain Ω . This grid is shown in figure 1. Given a field E_z specified at the nodes of the grid, one can compute approximate spatial derivatives using finite difference formulas. The simplest approximation for a partial derivative is

$$\frac{\partial E_z}{\partial x}(i + \frac{1}{2}, j) = \frac{E_z(i + 1, j) - E_z(i, j)}{\Delta}. \quad (5)$$

The notation used in these finite difference equations is $E_z(i, j) = E_z(x_i, y_j)$, where x_i is the x -coordinate of the i th vertical grid line, and y_j is the y -coordinate of the j th horizontal grid line. The spacing between grid lines is Δ . Applying this finite difference once again yields an approximation for the second partial derivative,

$$\frac{\partial^2 E_z}{\partial x^2}(i, j) = \frac{E_z(i + 1, j) - 2E_z(i, j) + E_z(i - 1, j)}{\Delta^2}. \quad (6)$$

Replacing the laplacian in (3) with its finite difference approximation gives the finite difference equation,

$$\frac{E_z(i+1,j)+E_z(i,j+1)+E_z(i-1,j)+E_z(i,j-1)-4E_z(i,j)}{\Delta^2} + \gamma^2 E_z(i, j) = 0 \quad (7)$$

for each node (i, j) in the interior of Ω . The boundary $\partial\Omega$ is deformed to coincide with nearby nodes of the grid. A node (i, j) on the boundary will have $E_z(i, j) = 0$. This causes some terms in the equations for nodes next to the boundary to be eliminated.

One can visualize this equation using the stencil shown in figure 1. The stencil shows how the laplacian operator ∇^2 is applied to the grid. One overlays

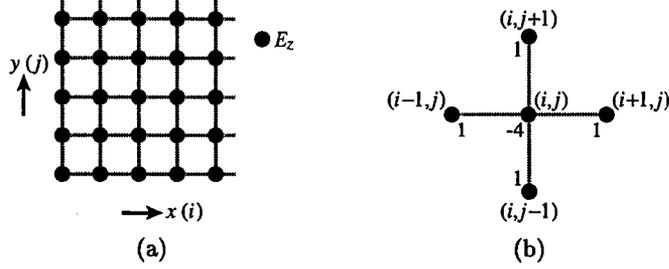


Figure 1: (a) Grid for finite difference method to compute E_z for TM modes in homogeneous waveguide. The domain Ω or its boundary is not shown. The vertical (fixed x) grid lines are labeled by an index i and the horizontal (fixed y) grid lines are labeled by an index j . Each node (filled circles) is labeled by a pair of indices (i, j) . (b) The stencil of the ∇^2 operator acting on the field grid.

the stencil on the grid at the node (i, j) and reads the coefficients in the ∇^2 finite difference operator. Higher order approximations to the laplacian will have larger stencils.

This set of equations (7) is an algebraic eigenvalue problem which can be solved by any of a number of techniques developed by applied mathematicians. One can visualize each equation as the row of a large sparse matrix,

$$\frac{-1}{\Delta^2} \begin{pmatrix} \ddots & & & & & & \\ & \ddots & & & & & \\ & & \ddots & & & & \\ \dots & 1 & \dots & 1 & -4 & 1 & \dots & 1 & \dots \\ & & & & \ddots & & & & \\ & & & & & \ddots & & & \\ & & & & & & E_x(i+1, j) & & \\ & & & & & & \vdots & & \end{pmatrix} \begin{pmatrix} \vdots \\ E_x(i-1, j) \\ \vdots \\ E_x(i, j-1) \\ \vdots \\ E_x(i, j) \\ \vdots \\ E_x(i, j+1) \\ \vdots \\ E_x(i+1, j) \\ \vdots \end{pmatrix} = \gamma^2 \begin{pmatrix} \vdots \\ E_x(i-1, j) \\ \vdots \\ E_x(i, j-1) \\ \vdots \\ E_x(i, j) \\ \vdots \\ E_x(i, j+1) \\ \vdots \\ E_x(i+1, j) \\ \vdots \end{pmatrix} \quad (8)$$

This can be written as the matrix equation $\mathbf{Ax} = \gamma^2\mathbf{x}$ where \mathbf{A} is the matrix on the left hand side representing the (negative) laplacian operator, and \mathbf{x} is the vector of field values at the nodes.

Now consider a two dimensional time domain problem with TE fields. This will be an example of the Yee finite difference algorithm². There are also eigenmode variations of the Yee algorithm³. The set of Maxwell's equations to solve is

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \quad (9)$$

$$\nabla \times \mathbf{H} = \frac{\partial \mathbf{D}}{\partial t} + \mathbf{J}. \quad (10)$$

For TE fields, the electric field \mathbf{E} is in the x - y plane and the magnetic field H_z is out of the plane. One places a grid like the one shown in figure 2(a) over the domain Ω .

The unknowns are the fields E_x , E_y and H_z . Notice that the points on the grid where one field is computed, say E_x , does not coincide with the points on the grid where the other fields are computed, in this case E_y and H_z . The motivation for this arrangement of field quantities is to conveniently compute accurate approximations of the curl of the fields.

Applying the basic finite difference formulas, the time derivative of the magnetic field,

$$\frac{\partial B_z}{\partial t} = -\frac{\partial E_y}{\partial x} + \frac{\partial E_x}{\partial y} \quad (11)$$

is approximated by

$$\begin{aligned} \frac{\partial B_z}{\partial t}(i + \frac{1}{2}, j + \frac{1}{2}) = & -\frac{E_y(i + 1, j + \frac{1}{2}) - E_y(i, j + \frac{1}{2})}{\Delta} \\ & + \frac{E_x(i + \frac{1}{2}, j + 1) - E_x(i + \frac{1}{2}, j)}{\Delta}. \end{aligned} \quad (12)$$

Notice how the field quantities on the grid are arranged for convenient and accurate calculation of the time derivative of the magnetic field. Likewise, the time derivatives of the electric field,

$$\frac{\partial D_x}{\partial t} = \frac{\partial H_z}{\partial y} - J_x \quad (13)$$

$$\frac{\partial D_y}{\partial t} = -\frac{\partial H_z}{\partial x} - J_y \quad (14)$$

are approximated by

$$\frac{\partial D_x}{\partial t}(i + \frac{1}{2}, j) = \frac{H_z(i + \frac{1}{2}, j + \frac{1}{2}) - H_z(i + \frac{1}{2}, j - \frac{1}{2})}{\Delta} - J_x(i + \frac{1}{2}, j) \quad (15)$$

$$\frac{\partial D_y}{\partial t}(i, j + \frac{1}{2}) = -\frac{H_z(i + \frac{1}{2}, j + \frac{1}{2}) - H_z(i - \frac{1}{2}, j + \frac{1}{2})}{\Delta} - J_y(i, j + \frac{1}{2}) \quad (16)$$

Once again, the location of the field quantities on the grid provides for convenient and accurate calculations of the time derivative of the electric field.

Now perform the integration in time, again using the Yee algorithm. This time integration scheme is also called the leapfrog algorithm. Suppose one

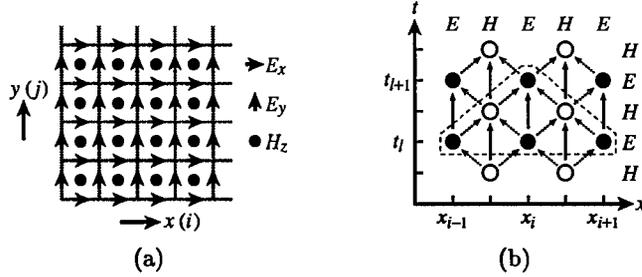


Figure 2: (a) Grid for Yee algorithm to compute TE fields (E_x , E_y and H_z) in a time domain problem. (b) The leapfrog algorithm for advancing fields in time. The electric (solid circles) and magnetic (open circles) fields are displaced by one half of a time step as well as one half of the grid spacing. Arrows show the flow of information, and the dashed line encloses field quantities involved in the calculation of $E(x_i, t_{l+1})$ from the electric fields at time t_l .

knows the fields at time steps separated by time Δt . The basic finite difference formula for the magnetic field time derivative at time step t_l gives

$$\frac{\partial B_z}{\partial t}(i + \frac{1}{2}, j + \frac{1}{2}, t_l) = \frac{B_z(i + \frac{1}{2}, j + \frac{1}{2}, t_l + \frac{1}{2}\Delta t) - B_z(i + \frac{1}{2}, j + \frac{1}{2}, t_l - \frac{1}{2}\Delta t)}{\Delta t}. \quad (17)$$

This equation can be used to advance the magnetic field one time step,

$$B_z(i + \frac{1}{2}, j + \frac{1}{2}, t_l + \frac{1}{2}\Delta t) = B_z(i + \frac{1}{2}, j + \frac{1}{2}, t_l - \frac{1}{2}\Delta t) + \Delta t \frac{\partial B_z}{\partial t}(i + \frac{1}{2}, j + \frac{1}{2}, t_l) \quad (18)$$

where the time derivative of B_z is taken from equation (12). Likewise, the electric field can be advanced a time step via the pair of equations,

$$D_x(i + \frac{1}{2}, j, t_l + \Delta t) = D_x(i + \frac{1}{2}, j, t_l) + \Delta t \frac{\partial D_x}{\partial t}(i + \frac{1}{2}, j, t_l + \frac{1}{2}\Delta t) \quad (19)$$

$$D_y(i, j + \frac{1}{2}, t_l + \Delta t) = D_y(i, j + \frac{1}{2}, t_l) + \Delta t \frac{\partial D_y}{\partial t}(i, j + \frac{1}{2}, t_l + \frac{1}{2}\Delta t) \quad (20)$$

where the time derivatives of D_x and D_y are taken from equations (15) and (16), respectively. Notice that the electric and magnetic fields are displaced one half time step just like the fields are displaced in space. This spacing is shown in figure 2(b).

The leapfrog algorithm is one example of an explicit algorithm for advancing the fields in time. Explicit algorithms advance the field at a particular point on the grid using only nearby fields. Most time domain calculations use such techniques because they can update the fields very quickly. The main drawback to explicit algorithms is that the time step must be small in order to avoid instabilities in the calculation. If the time step is too large then the fields will unphysically and spontaneously grow starting from numerical noise. The corresponding limit to the step size is the Courant condition⁴, which is roughly $c\Delta t < \Delta x$, where Δx is the minimum grid spacing.

There is another class of algorithms for advancing fields in time, called implicit algorithms. Implicit algorithms advance the fields by solving a large matrix equation involving all of the fields at the current and next time step. The field at a particular point at the next time step depends on the entire field at the current time step, not just nearby fields at the current time step. Updating the fields using an implicit algorithm is slow. However, the time step can be much larger because it is not limited by the Courant condition. One still chooses a time step small compared to the time scales of the phenomena being modeled (e.g., ten or more time steps per RF period).

The Yee algorithm works well in three dimensions just like it works in two dimensions. As in two dimensions, the field quantities E_x , E_y , E_z , H_x , H_y and H_z in the three dimensional calculation are known at distinct locations on the grid. This is shown in figure 3(a). One often speaks of dual grids in these calculations. The grid for the magnetic field \mathbf{H} is offset from the grid for the electric field \mathbf{E} by one half of a grid spacing along each coordinate axis. The same leapfrog algorithm is applied to time domain calculations in three dimensions, so the computed \mathbf{E} and \mathbf{H} fields are interleaved in time and separated from one another by one half time step.

Integrating Maxwell's equations over surfaces leads to the same set of difference equations that comprise the Yee algorithm. For example, to compute $\partial B_z / \partial t$ one starts with

$$\int_A \frac{\partial \mathbf{B}}{\partial t} \cdot d\mathbf{s} = - \int_A \nabla \times \mathbf{E} \cdot d\mathbf{s} = - \int_{\partial A} \mathbf{E} \cdot d\mathbf{l}, \quad (21)$$

where the area A is an element of the grid as shown in figure 3(b). The integral over each line segment in the boundary ∂A is approximately the field at the midpoint of the line segment multiplied by the length of the line segment. This is a one point numerical integration scheme. A similar one point integration scheme is applied to the surface integral on the left hand side of equation (21).

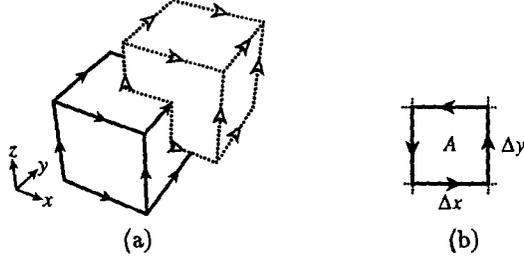


Figure 3: (a) Dual grids for the Yee algorithm in three dimensions. The solid cube is an element of the electric field grid and the dotted cube is an element of the magnetic field grid. Solid arrows indicate the location of electric field components on the grid. Open arrows indicate the location of magnetic field components on the grid. (b) Finite difference equations can be obtained using approximate numerical integration over grid element surfaces A and boundaries ∂A . The arrows indicate the direction of integration around the boundary ∂A .

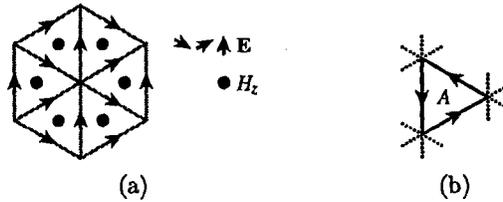


Figure 4: (a) A triangular grid for a generalized finite difference calculation of TE fields in the time domain. The electric field components are aligned with the edges of the triangular grid elements. (b) Integration over a triangular grid element A yields the generalized finite difference equations. Arrows indicate the direction of integration over the element boundary ∂A .

The result is

$$\begin{aligned} \frac{\partial B_z}{\partial t}(i + \frac{1}{2}, j + \frac{1}{2})\Delta x\Delta y = & -E_x(i + \frac{1}{2}, j)\Delta x - E_y(i + 1, j + \frac{1}{2})\Delta y \\ & + E_x(i + \frac{1}{2}, j + 1)\Delta x + E_y(i, j + \frac{1}{2})\Delta y \end{aligned} \quad (22)$$

Dividing by $\Delta x\Delta y$ gives an equation for $\partial B_z/\partial t$ equivalent to equation (12).

Such integration schemes for generating equations can be applied to other grids which are not rectangular. These schemes are called generalized finite difference calculations. The grid may be logically rectangular but warped in physical space. The grid might not be rectangular at all. It could be a triangular grid as shown in figure 4(a). The field components are not aligned with any coordinate axes. They are aligned with the edges of grid elements.

Calculation of a particular field component, say E_x , in the post-processing of a solution will require interpolation of the computed field components on the edges of an element. Integration over a grid element A as shown in figure 4(b) yields the finite difference equations for the fields on this triangular grid.

2.2 Finite Element Methods

The finite element method is conceptually different from the finite difference method. There are many good books on the basic finite element method⁵. Both methods start from a strong formulation of a problem. For example, consider again the computation of TM modes in homogeneous waveguide: find the eigenmodes E_z and corresponding eigenvalues γ^2 such that

$$\nabla^2 E_z + \gamma^2 E_z = 0 \quad \text{in } \Omega \quad (23)$$

and

$$E_z = 0 \quad \text{on } \partial\Omega. \quad (24)$$

This is called a strong formulation of the problem because a trial solution E_z must be twice differentiable in order to check equation (23). At this point the finite element method diverges from the finite difference method.

In contrast to the finite difference method, which would proceed by replacing the laplacian with a finite difference operator, the finite element method first converts the strong formulation of the problem to a weak formulation: find the eigenmodes E_z and corresponding eigenvalues γ^2 such that for all test functions F_z ,

$$\int_{\Omega} \nabla F_z \cdot \nabla E_z - \gamma^2 F_z E_z \, d\Omega = 0. \quad (25)$$

In this formulation, both E_z and F_z are zero on the boundary $\partial\Omega$. This formulation of the problem is called a weak formulation because a trial solution E_z need only have itself and its first derivative be integrable in order to check equation (25). This is a much weaker constraint on a trial solution than is demanded in the strong formulation. The weak formulation is also called the variational formulation.

The strong and weak formulations are equivalent. A solution of the strong formulation is a solution of the weak formulation and vice versa. One can obtain the weak formulation from the strong formulation by multiplying equation (23) by a test function F_z , integrating over the domain Ω , and subsequently integrating by parts. The boundary term in the integration by parts vanishes due to the boundary conditions on E_z and F_z .

The finite element method proceeds from the weak formulation by restricting the trial functions E_z and test functions F_z to a linear combination of M basis functions N_i ,

$$E_z(\mathbf{x}) = \sum_{i=1}^M e_i N_i(\mathbf{x}), \quad \text{and} \quad (26)$$

$$F_z(\mathbf{x}) = \sum_{i=1}^M f_i N_i(\mathbf{x}). \quad (27)$$

This is called the Galerkin formulation. The problem has been reduced from the infinite dimensional weak formulation to the finite dimensional Galerkin formulation. The Galerkin formulation can be solved on a computer. The linear combinations (26) and (27) can be installed in equation (25) and the summation factored out of the integration. Performing the integrations over the basis functions yields the matrix equation

$$\mathbf{A}\mathbf{e} - \gamma^2 \mathbf{B}\mathbf{e} = 0 \quad (28)$$

where \mathbf{e} is the vector of coefficients e_i . The components of the matrices \mathbf{A} and \mathbf{B} are

$$A_{ij} = \int_{\Omega} \nabla N_i \cdot \nabla N_j d\Omega, \quad \text{and} \quad (29)$$

$$B_{ij} = \int_{\Omega} N_i N_j d\Omega. \quad (30)$$

The matrix equation generated from the finite element method is a generalized eigenvalue problem. This is a modest drawback compared to the finite difference method, where the corresponding matrix equation (8) had only a matrix \mathbf{A} and no matrix \mathbf{B} .

There is one final step in the finite element method. The final step is the choice of basis functions. In the finite element method, the basis functions N_i are chosen to be simple functions (e.g., linear or quadratic polynomials) on simple elements (e.g. triangles or quadrilaterals). Each basis function is non-zero over only a few adjacent elements. This makes the matrices \mathbf{A} and \mathbf{B} sparse.

An example of a domain Ω partitioned into elements is shown in figure 5. This collection of elements is called a mesh. The waveguide cross-section is partitioned with quadratic triangular lagrange-type elements in this example.

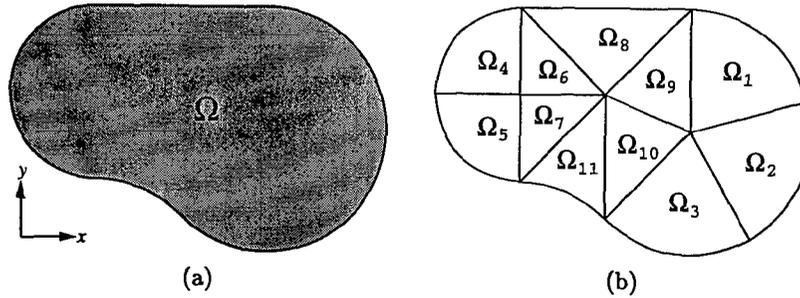


Figure 5: (a) A domain Ω to be partitioned into finite elements. (b) Quadratic triangular elements covering the domain Ω . The curved sides of the elements can closely follow the boundary of Ω with only a few elements.

Each element Ω_e is the image of the unit triangle $\hat{\Omega}$ under a quadratic coordinate transformation \mathbf{x}_e ,

$$(x, y) = \mathbf{x}_e(r, s) = \sum_{i=1}^6 \mathbf{x}_{ei} \hat{N}_i(r, s). \quad (31)$$

This map is illustrated in figure 6. The coordinates on $\hat{\Omega}$ are (r, s) . The basis functions \hat{N}_i of the coordinate transformation are the quadratic lagrange polynomials for the unit triangle. These basis functions are

$$\begin{aligned} \hat{N}_1 &= r(2r - 1) & \hat{N}_4 &= 4rs \\ \hat{N}_2 &= s(2s - 1) & \hat{N}_5 &= 4st \\ \hat{N}_3 &= t(2t - 1) & \hat{N}_6 &= 4rt, \end{aligned} \quad (32)$$

where $t = 1 - r - s$. The basis function \hat{N}_i is 1 at the i th node and 0 at all remaining nodes.

The curved sides of quadratic elements closely follow the curved boundaries of the waveguide. This is one aspect of the finite element method that provides for very accurate solutions. Boundaries in finite difference calculations are often restricted to grid lines or diagonals of the rectangular grid. This restriction introduces errors into the finite difference calculation. Finite element calculations have no such restriction. Linear elements can have their straight sides at any orientation with respect to the coordinates axes. Higher order elements have curved sides that can follow curved boundaries even more closely.

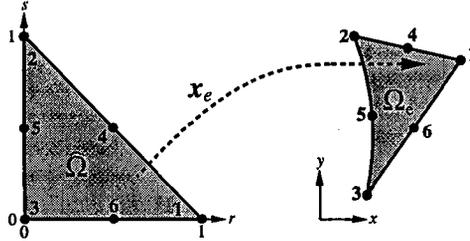


Figure 6: A quadratic lagrange-type triangular element Ω_e is the image of the unit triangle $\hat{\Omega}$ under a quadratic coordinate transformation \mathbf{x}_e . The bold numbers are local node indices.

The field E_z on each element Ω_e is also expressed as a linear combination of these basis functions,

$$E_z(x, y) = \sum_{i=1}^6 e_{ei} \hat{N}_i(\mathbf{x}_e^{-1}(x, y)). \quad (33)$$

The field in this case has been expressed in global coordinates (x, y) instead of local coordinates (r, s) . The field on each element is a quadratic function of the local coordinates. These elements are called isoparametric because the basis for the field E_z is the same as the basis for the coordinate transformation \mathbf{x}_e .

The availability of quadratic and higher order basis functions are another aspect of finite element calculations that provides for excellent accuracy. High order finite difference techniques are straightforward to implement in the interior of a regular grid, but accurate finite difference schemes for the grid points near the boundaries are fairly difficult to implement. In contrast, the finite element method is easily adapted to higher order field approximations inside each element. The integrals in equations (29) and (30) are just as easy to evaluate for elements next to the boundary as they are for elements in the center of the mesh.

The basis functions \hat{N}_i employed in equation (33) are local basis functions because they are defined on individual elements. The local basis functions need to be assembled into global basis functions N_i defined on the whole domain Ω . The global basis functions need to be continuous across element boundaries. The local basis functions are defined so that they can be matched easily across element boundaries. Figure 7 shows examples of global basis functions spanning a few adjacent elements. Some high order local basis functions comprise global basis functions by themselves because they can be zero at the element boundaries and thus satisfy interelement continuity conditions.

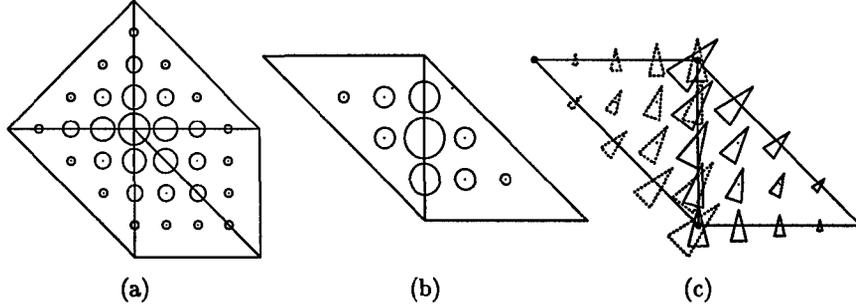


Figure 7: Examples of global basis functions. (a) A global basis function comprised of linear basis functions associated with a corner node. (b) A global basis function comprised of quadratic basis functions associated with a midside node. For (a) and (b) the field value is indicated by the size of the open circles. (c) A global vector basis function comprised of linear edge basis functions associated with a common edge. The direction and size of the arrowheads indicate the direction and size of the vector field. The dotted (solid) arrowheads are the basis function in the left (right) element.

The finite element method can also be applied to three dimensional electromagnetic calculations. Consider once again the eigenmode problem. Assuming the fields have an $e^{j\omega t}$ time dependence, the two curl equations

$$\nabla \times \mathbf{E} = -j\omega \mathbf{B} \quad \text{and} \quad (34)$$

$$\nabla \times \mathbf{H} = j\omega \mathbf{D}, \quad (35)$$

are combined with the linear constitutive relations to obtain the curl-curl equation

$$\nabla \times (\boldsymbol{\mu}^{-1} \nabla \times \mathbf{E}) - \omega^2 \boldsymbol{\epsilon} \mathbf{E} = 0. \quad (36)$$

This equation is solved for the eigenmode fields \mathbf{E} and corresponding eigenfrequencies ω . The fields at the perfectly conducting boundaries of the cavity satisfy $\hat{\mathbf{n}} \times \mathbf{E} = 0$. This is the strong formulation of the eigenmode problem.

As in the two dimensional example, the strong formulation is converted to an equivalent weak formulation by taking the dot product of equation (36) with a test field \mathbf{F} , integrating the dot product over the domain Ω , and then integrating by parts. The resulting problem is to find the eigenmode fields \mathbf{E} and corresponding eigenfrequencies ω such that for all test fields \mathbf{F} ,

$$\int_{\Omega} (\nabla \times \mathbf{F}) \cdot \boldsymbol{\mu}^{-1} (\nabla \times \mathbf{E}) - \omega^2 \mathbf{F} \cdot \boldsymbol{\epsilon} \mathbf{E} d\Omega = 0. \quad (37)$$

The field \mathbf{F} , like the field \mathbf{E} , is constrained by the boundary condition $\hat{\mathbf{n}} \times \mathbf{F} = 0$.

The fields \mathbf{E} and \mathbf{F} are written as linear combinations of vector basis functions $\mathbf{N}_i(\mathbf{x})$. The resulting Galerkin formulation leads to a generalized eigenvalue problem $\mathbf{A}\mathbf{e} - \omega^2\mathbf{B}\mathbf{e} = 0$ with matrix components

$$A_{ij} = \int_{\Omega} \nabla \times \mathbf{N}_i \cdot \boldsymbol{\mu}^{-1} (\nabla \times \mathbf{N}_j) d\Omega \quad \text{and} \quad (38)$$

$$B_{ij} = \int_{\Omega} \mathbf{N}_i \cdot \boldsymbol{\epsilon} \mathbf{N}_j d\Omega. \quad (39)$$

The domain Ω is partitioned into elements. Local basis functions are chosen, and global basis functions are assembled from local basis functions. The local basis functions in this case are mixed-order 1-form basis functions, which are also called edge elements⁶. These elements are not isoparametric because the basis functions for the field are different from the basis functions for the coordinate transformation from local to global coordinates.

Thinking of the basis functions as 1-forms, or covariant vectors, makes expression of the basis functions in local coordinates particularly simple. Given a 1-form basis function expressed in local coordinates, the corresponding vector basis function expressed global coordinates is obtained by transforming the 1-forms from local and global coordinates and then using the metric tensor to convert the 1-form to a (contravariant) vector.

Global basis functions must have continuous tangential field across element boundaries, but the component of the field normal to interelement boundaries may be discontinuous. An example is shown in figure 7(c). Discontinuous fields may seem disquieting at first, but if one thinks of the continuity of electric fields across material boundaries, one will appreciate that the component of the field \mathbf{E} normal to an interface boundary is allowed to be discontinuous. In typical cases where an interelement boundary is not also a material boundary, there may be a tiny discontinuity in the field \mathbf{E} but it will be negligible.

Unlike the two dimensional example above, the choice of basis functions is critical to the success of three dimensional electromagnetic finite element calculations. This is the motivation for using mixed order-basis functions⁷. For a set of quadratic basis functions in local coordinates u , v and w , the covariant field component E_w is linear in w and quadratic in u and v .

Why is this choice of basis important? The remaining two Maxwell's equations, $\nabla \cdot \mathbf{D} = 0$ and $\nabla \cdot \mathbf{B} = 0$, are not considered explicitly in the strong and weak formulations above. This admits a class of solutions, the spurious modes, that have non-zero divergence. However, the spurious solutions to equations (36) or (37) also have zero frequency, so in principal they are easily separated from the desired physical modes which have non-zero frequency. However, when one passes from the weak formulation to the finite dimensional

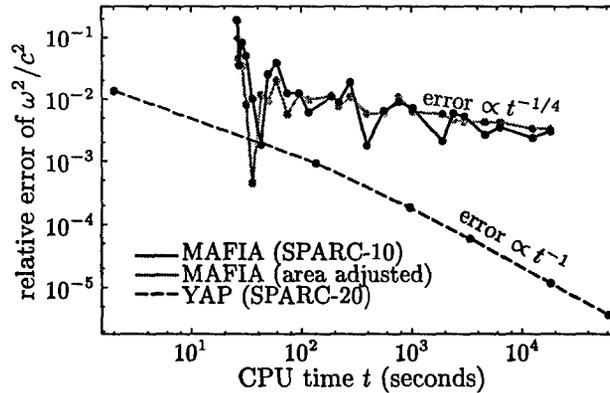


Figure 8: Relative eigenvalue error versus CPU time for MAFIA and YAP. The test structure is a pillbox cavity with radius $\rho = 1$ and height $l = 1$. The YAP results are electric field calculations. The MAFIA results employed a uniform grid. The area adjusted MAFIA results scale the uniform x - y grid so that the "circular" cross-sectional area is correct.

Galerkin formulation, there is no guarantee that the spurious modes will still have zero frequency. A proper choice of basis functions is required to ensure that the frequency of the spurious modes will still be identically zero.

The work in a finite element calculation is a lot more complex than the corresponding finite difference calculation, but the result is ultimately a better solution in less computational time. Figure 8 shows a comparison of the finite difference code MAFIA with the finite element code YAP. The codes are computing the lowest eigenfrequency of a test cylindrical cavity. The error in MAFIA is dominated by the error in representing the curved boundary. This error is $\mathcal{O}(h)$, where h is the element size. If an error of 0.1% in the frequency is tolerable then the convenience of the finite difference method makes it attractive. However, for higher accuracy calculations, a finite element calculation is more appropriate and may even be necessary. In this example, the finite element code YAP employed quadratic basis functions on quadratic hexahedral elements to achieve an error of $\mathcal{O}(h^4)$. The accuracy of the finite element calculation is unaffected by the curved cavity walls.

2.3 Finite Volume and Boundary Element Methods

The generalized finite difference method was described in subsection 2.1 as the generation of finite difference equations by integrating Maxwell's equations over element surfaces. Another method for arriving at finite difference equations in

three dimensional calculations is to integrate Maxwell's equations over element volumes instead of element surfaces. This is called the finite volume method. For example,

$$\int_V \frac{\partial \mathbf{B}}{\partial t} dV = - \int_V \nabla \times \mathbf{E} dV = - \oint_{\partial V} ds \times \mathbf{E}, \quad (40)$$

where V is a three dimensional element. In finite volume methods, the location on the grid of the computed field quantities is different from the Yee algorithm. For example, the computed magnetic field \mathbf{B} is typically located at the element centers instead of the center of element faces.

Finite volume methods are getting a lot of attention from the antenna and radar communities. They claim that finite volume methods handle curved surfaces much better than finite difference methods. However, the finite volume calculations require more computation for each element. Hybrid finite difference/finite volume codes have been written to employ the more accurate finite volume methods in regions near curved boundaries but everywhere else they take advantage of the speed and efficiency of finite difference methods.

Electromagnetic field calculations can be posed as finding the distribution of surface currents and charge densities instead of electromagnetic fields in a volume of space. Such methods are called boundary methods. Moment methods, EFIE and MFIE are versions of boundary methods. Boundary methods potentially reduce the size of the calculation by reducing the number of unknowns. Instead of the grid covering the volume of a cavity, the grid only covers the surfaces of the cavity. When the size of the elements in both cases is similar, the boundary methods typically employ fewer elements and thus fewer unknowns. Also, grid generation may be easier for boundary methods since modern CAD programs usually describe complicated cavities in terms of surfaces.

Green's functions are used to compute electromagnetic fields from current and charge distributions. For harmonic fields in a cavity, the electric field is

$$\mathbf{E}(\mathbf{x}) = \int_{\partial\Omega} \mathbf{G}(\mathbf{x}, \mathbf{x}') \cdot \mathbf{J}(\mathbf{x}') da + \nabla \int_{\partial\Omega} g(\mathbf{x}, \mathbf{x}') \sigma(\mathbf{x}') da, \quad (41)$$

where \mathbf{G} is a dyadic Green's function applied to the surface current distribution \mathbf{J} and g is a scalar Green's function applied to the surface charge distribution σ . These Green's functions depend on the frequency ω . The surface current density \mathbf{J} and the charge density σ are related by charge conservation.

As in the finite element method, one can multiply $\mathbf{E}(\mathbf{x})$ by test functions $\mathbf{F}(\mathbf{x})$ which are tangential to the metal walls of the cavity. Integrating

over the surface $\partial\Omega$ once more gives a weak formulation of the boundary problem. The eigenmode problem is to find surface current density \mathbf{J} and corresponding eigenfrequencies ω such that for all test fields \mathbf{F} tangential to the cavity walls,

$$\int_{\partial\Omega} \mathbf{F} \cdot \mathbf{E}(\mathbf{J}, \sigma; \omega) da = 0, \quad (42)$$

where $\mathbf{E}(\mathbf{J}, \sigma; \omega)$ is given by equation (41).

Next, one partitions the surface into elements and writes \mathbf{J} and σ as linear combinations of basis functions,

$$\mathbf{J}(\mathbf{x}) = \sum_{i=1}^M j_i \mathbf{N}_i(\mathbf{x}). \quad (43)$$

Recall that the surface charge density σ can be obtained from the current density \mathbf{J} due to charge conservation. Installing these linear combinations into equation (42) and selecting a finite set of test functions \mathbf{F} gives a finite dimensional problem which can be solved on the computer. Unlike the finite element method, test functions \mathbf{F} very different the surface current density basis functions \mathbf{N}_i are common. Choosing the test functions to be delta functions (perhaps two at the center of each element) makes the boundary methods look very different from typical finite element methods.

Matrices are obtained by factoring the summation outside of the integrals and performing the integrations. These integrations must be performed carefully since the Green's function has singularities.

Unlike the finite element method, the matrices in boundary methods are full matrices, not sparse matrices. This is an unavoidable drawback of boundary methods compared to finite element and finite difference methods. As problems get larger due to more cavity features or a finer grid, the relative advantage of having fewer unknowns will be overwhelmed by the disadvantage of having to compute a full matrix. Thus, beyond some level of complexity or desired accuracy in a cavity field calculation, finite element and finite difference methods will always be more efficient than boundary methods.

There are a few exceptions⁸, but another drawback of boundary methods is that the eigenmode problem is usually not an algebraic eigenvalue problem. That is, the eigenvalue problem cannot be written as $\mathbf{A}\mathbf{x} = \lambda\mathbf{x}$ or $\mathbf{A}\mathbf{x} = \lambda\mathbf{B}\mathbf{x}$. An eigenvalue λ cannot be factored out of the computed matrix. This makes the solution of the eigenmode problem similar to finding the roots of an arbitrary function. One varies ω until a non-trivial solution to the matrix equations exists. The corresponding eigenvector give the wall currents and the Green's function can be used to compute fields inside the cavity if desired.

For frequency domain problems, the distinction between algebraic and non-algebraic eigenvalue problems is less important because there is less to gain from factoring an eigenvalue outside of the matrices. If one is solving for the fields at many different frequencies then there is some advantage to algebraic (i.e., finite difference or finite element method) formulations which do not have to recompute matrices for each frequency. Frequency domain calculations employing boundary methods, on the other hand, will have to recompute a full matrix for each desired frequency.

One application where boundary methods excel is in open, unbounded problems where the fields can radiate into free space. Such problems are not handled perfectly by finite element and finite difference methods, although this is an active research area⁹. Some codes combine finite element and boundary element methods into a hybrid code which takes advantage of each method's strengths and mitigates each method's weaknesses. A finite element mesh will cover the "interior" and details of the field near the structure. At some modest distance from the structure a modest-sized boundary element mesh handles interaction of the structure with the free space environment.

Acknowledgements

This work was supported by DOE, contract W-7405-ENG-36.

References

1. J.F. Deford *et al*, Particle Accel. **45**, 135-148 (1994).
2. K.S. Yee, IEEE Trans. Antennas Prop. **14**, 302-307 (1966).
3. T. Weiland, Particle Accel. **17**, 227-242 (1985).
4. Courant, R., Friedrichs, K. and Lewy, H., Mathematische Annale **100**, 32-74 (1928). English translation IBM Journal **11**, 215-234 (1967).
5. Claes Johnson, Numerical Simulation of Partial Differential Equations by the Finite Element Method, Cambridge (1987) is my favorite.
6. M.L. Barton, and Z.J. Cendes, J. Appl. Phys. **61**, 3919-21 (1987).
7. C.W. Crowley *et al*, IEEE Trans. Magnetics **24**, 397-400 (1988).
8. P. Arcioni *et al*, IEEE Trans. Microwave Theory Tech. **43**, 1848-56 (1995).
9. For example, see IEEE Trans. Antennas Prop. **45**, 313-572 (1997).

M98003193



Report Number (14) CA-UR--97-4080
CONF-9609245--

Publ. Date (11) 19960909

Sponsor Code (18) DOE/MA, XF

UC Category (19) UC-910, DOE/ER

DOE