ECE 6350

Instructions for Executing EMPACK

- The code EMPACK is a collection of Fortran 90 files contained in the distributed zipfile, EMPACK for Hwk6.zip. Unzipping the file into a directory puts all the F90 files in the designated directory and adds a subdirectory (…/Testsuite) with sample problem cases included. The data files needed to run various problems are located in appropriate subdirectories under /Testsuite.

- Once a project is assembled and compiled on a machine with Compaq Visual Fortran (CVF) installed, double clicking on the *.dsw file will reactivate the project; in INTEL Fortran, the corresponding project name is *.sln.

- Upon executing the program, a DOS window will appear, and the program will ask for an input filename. Note that since EMPACK *assumes* the file extension will be *.dat, you should *not include* the filename's extension. If you do not, you'll receive the message "cannot find the file *.dat.dat" when you try to run the code!

- On most Windows machines, you can save tedious typing of the input filename and its path by navigating to the folder where the input data file exists and dragging-and-dropping the filename into the DOS window. *Don't forget to remove the ".dat" file extension before proceeding, however*!

- Once the input filename is entered, EMPACK will ask for a filename to write the output data to. Again, you can usually save typing paths and filenames by using the "up arrow" key to get the path and filename from the previous step to appear in the window! (This assumes you want to write the output file to the same directory as the input data file.) Note that if you already have an output file by the same name in the same directory, however, you will be asked if you want to overwrite it. If you don't want to do this, you'll have to modify the output data filename in some way.

- If you know how to use the debugger, you should follow the code through a few cycles of reading input data, computing an element's geometry, filling an element matrix, assembling its contents in the system matrix, filling the right hand side vector, and finally solving the system of equations. Many of the things mentioned in class (and a few not mentioned) will become obvious very quickly by doing this! If you are unfamiliar with the debugger, you should get some help and learn how to do the following:
    - run to cursor and/or to a breakpoint
    - show next statement
    - step into, over, and out of subroutines or functions (procedures)
    - add variables to a watch window

- A typical input (*.dat) file is organized as follows:

```
TEstripEFIE.dat models a 0.5 [m] strip along the x-axis  ←Title/comment line for data file

Nodelist:

21                ! geometry nodes       ←The number of global node points (listed next)
 -0.250  0.0  0.0  ! geometry nodal coordinates (x,y,z) ←(x,y,z) coordinates of node 1
 -0.225  0.0  0.0  ! geometry nodal coordinates (x,y,z)
      ⋮                                                                    .
  0.200  0.0  0.0  ! geometry nodal coordinates (x,y,z)
  0.225  0.0  0.0  ! geometry nodal coordinates (x,y,z)
  0.250  0.0  0.0  ! geometry nodal coordinates (x,y,z) ←(x,y,z) coordinates of node 21

Elementlist:

20    19          ! number_of_elements, number_of_unknowns←# elements, # of DoFs
1  strip 1    2          ! element_id, element_type,  element_model_order, ← begin #1
                            ,number_element_model_nodes
  1   2                 ! global_model_node_numbers
  1                     ! number_unknown_species (e.g., electric only)
  electric div  0  2    !  unknown_species_type; conforming_property, basis order,
                            ,number_local_interp_nodes
  0   1                 ! global_dof_numbers
  2-d efie_te  0.00     ! greens_function;  boundary_condition;  element_attribute←end #1

      ⋮
```

Definitions of the above element quantities

Element_id	Element number
element_type	Element geometry type: (infinite) strip, wire, triangle, quadrilateral, tetrahedron, etc.
element_model_order	Order of polynomial interpolating the element; EMPACK currently supports only order 1 (linear elements)
number_element_model_nodes	Number of nodes to describe linear element geometry (strip, wire=2; triangle=3; tetrahedron =4; etc.)
global_model_node_numbers	Global node numbers of nodes defining the element
number_unknown_species	Number of types of DoFs defined per DoF node
unknown_species_type	”electric” or ”magnetic” current or field

conforming_property ”div” or “curl” = div- or curl-conforming

basis order Polynomial order of interpolating basis/test function; EMPACK currently supports only “0 order complete” (constant div or curl, mixed order currents/fields)

number_local_interp_nodes Number of nodes per element with associated DoFs (TM strip=1; TE strip,wire=2; triangle=3; div tetrahedron =4; curl tetrahedron =6, etc.)

global_dof_numbers Global DoF numbers; “0” means no DoF

greens_function “2-d” , “3- d”, “Layered Media”, etc.

boundary_condition “efie_te”, “mfie_tm”, “pmchwt”, etc.

element_attribute wire radius, element thickness, etc.

⋮ ⋮

```
20 strip 1    2        ! element_id, element_type,  element_model_order, ← begin #20
                           number_element_model_nodes
  20 21                ! global_model_node_numbers
  1                    ! number_unknown_species (e.g., electric only)
  electric div  0  2  !  unknown_species_type; conforming_property; basis order,
                           number_local_interp_nodes
  -19  0               ! global_dof_numbers
  2-d efie_te  0.00 ! greens_function;  boundary_condition;  element_attribute ← end #20
```

Excitation data:

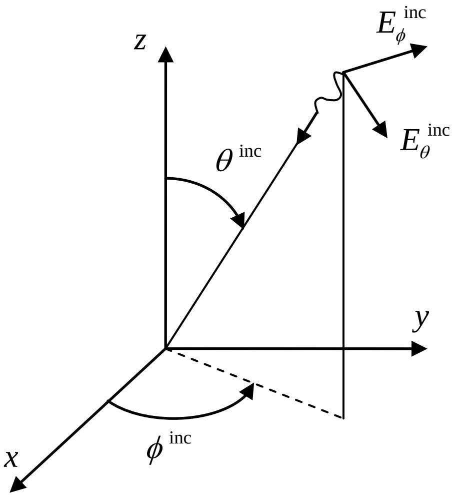
1 No. of *different* plane waves and/or voltage sources at a given freq.

1 0 No. of plane waves, number of voltage sources

90.0 90.0 (0.,0.) (-377.0,0.0) θ^{inc}, ϕ^{inc}, E_θ^{inc} (complex), E_ϕ^{inc} (complex)

1 No. of frequencies, each listed on a line below

3.000E+8 SOLVE_BY_LU Frequency, solution_method



Output control:

ALL Write out "ALL" or “NONE” unknowns or print those listed in a “LIST”

1 90. 90. # of far field points along θ: θ start and ending values along the cut

181 0. 360. # of far field points along ϕ: ϕ start and ending values along the cut

NONE Write out “NONE” near field values or print those listed in a “LIST”