ANALYSIS OF THE FINITE ELEMENT RADIATION MODEL CODE FOR ANTENNA MODELING *

Marcos C. Medina, Constantine A. Balanis, Jian Peng and Panayiotis A. Tirkas
Department of Electrical Engineering
Telecommunications Research Center
Arizona State University
Tempe, Arizona 85287-7206

Abstract

The Finite Element Radiation Model (FERM) code is applied to examine radiation patterns and input impedance of various antenna configurations on complex helicopters. This paper addresses the strengths and weaknesses of the FERM code when applied in antenna analysis and is compared with other national codes, such as the Numerical Electromagnetics Code (NEC). In addition, guidelines are outlined for various applications of the code.

I. INTRODUCTION

The Finite Element Radiation Model (FERM) code is based on the Method of Moments and uses triangular surfaces patches to model the geometry. The triangular patches allow for enormous flexibility in modeling complex, curved three-dimensional geometries, such as helicopters. The basis functions used by the code are in accordance with the Rao *et al.* [1] basis functions. The FERM code was developed in 1987 at the MIT Lincoln Laboratory [2], and is a general purpose code with applications in radar cross section prediction and antenna modeling. A similar code is the Electromagnetic Surface Patch (ESP) code [3], which is based on quadrilateral patches. Unlike the FERM and ESP codes, the NEC is based on a wire grid model of the geometry. The FERM, ESP and NEC codes have all been applied by the Advanced Helicopter Electromagnetics (AHE) consortium for analysis of antennas on helicopters. In this paper, helicopter antennas are analyzed using the FERM code.

II. CODE MODIFICATIONS

The FERM code is written in a modular structure, allowing it to be highly portable. The more computational intensive portions of the code can therefore be transported to computers with larger memory. The modular form of the FERM code is a distinct advantage. The code was originally installed on a VAX 3100 workstation. The VAX workstation required too much time to simulate antenna radiation problems related to helicopter applications. In fact, if the number of unknowns reached 1800, the VAX was incapable of completing the computation due to memory limitations. The VAX did not have the memory needed for the segmentation most geometries required; therefore, the code was transported to an IBM RISC/6000 350 workstation.

At first, the more computationally intensive programs were transported to the IBM RISC workstation. Since antenna radiation studies were the main areas of interest, the programs

*This work was supported by the Advanced Helicopter Electromagnetics (AHE) Program and NASA under Grant NAG-1-1082.

EFIE2, EFIE3, and EFIE4V were the first to be transferred. The program EFIE2 numerically evaluates the EFIE's for the desired geometry and creates the impedance or Z-matrix. The program EFIE3 implements LU decomposition to decompose the Z-matrix. EFIE4V solves for the current distribution using the decomposed Z-matrix and the voltage sources stored in the primary data file. The execution of EFIE1, EFIE5V, and EFIE5R remained on the VAX workstation. The program EFIE1 converts the user geometry directives into an internal format for the numerical processing accomplished via EFIE2, EFIE3, and EFIE4V. Eventually, EFIE1 was migrated to the IBM RISC to avoid the problems associated with transferring data from VMS system to a UNIX system. EFIE5V and EFIE5R interface with the graphics package DISSPLA. Since DISSPLA is licensed strictly to the VAX workstation, these two routines must be executed on the VAX. These last two programs output the antenna radiation patterns.

The routines EFIE2, EFIE3, and EFIE4V required minor modifications for execution on the IBM RISC. They consisted of disengaging the subroutine used to calculate the CPU time, modifying the OPEN directives for temporary data files, and altering the format of the data files created by the FERM code. The IBM RISC is capable of listing the real-time and CPU time used during the execution of a program. Therefore, the subroutine CLOCK0, used to determine the execution time of a program on a VAX workstation, was eliminated.

All calls to temporary or scratch files were modified. VAX/VMS Fortran requires that when a SCRATCH file is created or opened the initial size of the file must be specified using the INITIALSIZE specifier [4]. Otherwise, no memory allocation is made. VS Fortran does not recognize the specifier INITIALSIZE and it was deleted from the OPEN statement for use on the IBM RISC [5].

The internally-formatted data files created by the FERM are unformatted and are not readable text files. These files cannot traverse the platform from a VAX workstation to an IBM workstation. Data was lost during the use of the File Transfer Protocol (FTP) to transport the data from EFIE4V (IBM workstation) to EFIE5R (VAX workstation). To circumvent this problem, the data files written to and read from (STORAGE, RESMAT, and RESULT) by the subprograms had to be formatted and required commas between each piece of data to ensure no data was lost. This involved rewriting all the format statements in EFIE1, EFIE2, EFIE3, and EFIE4V, as well as the subroutines used in these programs such as MISC and EFIE.

DRAW, a graphics package developed at Arizona State University to prepare rectangular and polar plots of radiation patterns, has replaced DISSPLA. DISSPLA is licensed strictly for execution on the VAX 3100 workstation. DRAW is executed on a UNIX workstation. In addition to the ease of interface with the FERM code on the IBM RISC, DRAW produces higher quality rectangular and polar plots than DISSPLA. DISSPLA was also used for the initial viewing of aircraft geometry. On a UNIX based workstation this is now done using GEOMVIEW [9], which is available via an anonymous ftp site.

In addition to eliminating DISSPLA from use as a graphics package, the Pattern Analysis and Database (PAD) was dismantled. PAD is not user-friendly. It is capable of organizing and manipulating data and storing numerous radiation patterns; however, for antenna radiation prediction, all that is needed is the computation and storage of radiation patterns. PAD was replaced by PLOTR, a subroutine created specifically to process radiation pattern computations. PLOTR uses the SPATTERN output data file created by EFIE5R and calculates normalized or absolute antenna gain.

III. GEOM INTERFACE

Super 3D [6] is a popular interactive graphics package capable of constructing solid surface helicopter models. This can be done by employing either a mouse or an input data file. The geometry can be rotated or scaled, thus allow-

ing modifications to be accomplished easily and accurately. Such a graphics package is important in the analysis and design of antennas on helicopters. Super 3D is used for constructing the helicopter geometry and attaching the various antennas to the fuselage. The solid surface geometry is then exported in ASCII format. GEOM, a geometry interface between the Super 3D and the electromagnetics codes, translates the solid surface model to either a wire grid model for NEC or surface patch model for ESP [7], [8].

The GEOM program was originally constructed to allow interaction between the NEC and ESP codes and the Super 3D. However, with the increasing use of the FERM code, by the AHE consortium, for antenna radiation pattern predictions and input impedance studies of HF antennas, a link between the FERM and Super 3D was also necessary. A subroutine, to convert geometry files from the Super 3D format to the FERM format, was also incorporated into the existing GEOM program.

The ESP code is a surface patch code, similar to the FERM. However, the method it employs in modeling geometries is much different. The ESP creates a surface by piecing together three-dimensional patches. Meanwhile, the FERM uses a series of lines to create a two-dimensional polygon representing a cross-section of the geometry. This two-dimensional polygon is then expanded along the third dimension to join with another cross-sectional polygon. The circumference of the polygon may also be scaled as it is stretched to form conical shaped structures. The FERM generates the triangular patches via the combination of the segmentation used between two polygons and the segmentation used for the lines of the polygon.

The FERM's method of constructing geometries is very efficient for rectangular, cylindrical, and spherical structures, but diminishes as the geometries become more irregular in shape. This allows the FERM to be highly suited for modeling aircraft fuselages, but lacking in the construction of the stabilizers and wings.

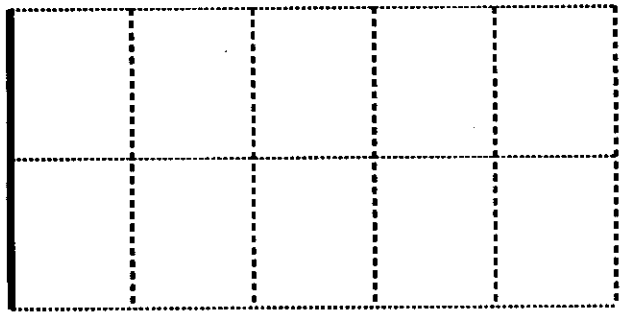
This modeling technique also eases the burdens encountered during segmentation. As mentioned before, when the FERM segments a surface, it relies upon the number of segments per line comprising the cross-sectional polygon and the number of sections between the polygons. Therefore, the number of patches in one local area can be increased without affecting the segmentation at other areas. For example, the number of patches along the section of the fuselage where the antenna is attached may be increased while the segmentation along nose and tail of the aircraft remains unaffected.

During the conversion of a Super 3D geometry file to a FERM format, sufficient cross-sections of the geometry must be taken to ensure that every contour of the surface is closely followed. The Super 3D cross-section coordinates are read into the FERM and used to construct the lines of the polygon. The number of segments per line is input by the user. After a cross-sectional polygon is built, it is joined to the previous one to build the fuselage. However, care must be taken when joining two polygons. The number of segments per line in a polygon must be identical for two polygons to be attached. Because of this, it was decided that the segmentation of each line, supplied by the user, remain constant throughout the FERM data file. The number of segments per line of a cross-sectional polygon determines the total segmentation around the circumference of the fuselage. Likewise, the total number of sections used between each polygon determines the total segmentation of the fuselage length. If an increase in the segmentation of one area is desired, the number of segments per line in that local area may be easily modified by the user to suit the application.

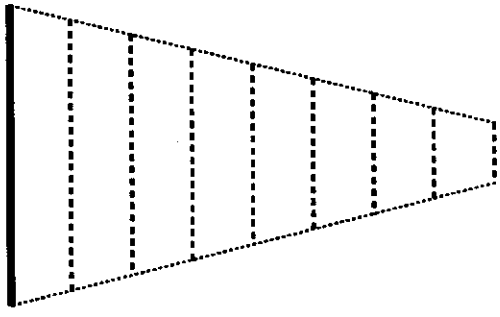
The stabilizers and wings of an aircraft are easily modeled with the ESP; they are simply two-dimensional plates. However, the FERM is mostly suited toward three-dimensional applications. The wings and stabilizers are much more difficult to model than the fuselage. The FERM generates plates in three ways. If the PLATE directive is employed, only two lines with a common end point are needed. The FERM then creates a plate by generating two lines of the same length as the original two lines and plac-

ing them parallel. Thus, the PLATE command is limited to generating squares or rhomboids. If a plate has more than four sides, or has two opposite sides that are not parallel, then this directive cannot be used.

Implementing the TRIA directive rather than the PLATE directive provides a more flexible method. The TRIA (short for triangle) command also requires two lines with a common end point, but generates a triangle rather than a rhombus. Moreover, any two-dimensional plate can be constructed from a collage of triangular patches. Therefore, this method provides the most flexibility in generating arbitrary shapes with minimal difficulty.



(a)



(b)

Figure 1: The use of the PRISM directive to produce (a) a rectangular plate and (b) a tapered wing.

Another technique employed by the FERM to build two-dimensional plates follows along the same lines as the construction of the three-dimensional surfaces. However, it is limited to working on plates with four sides where at least one pair of opposite sides are parallel. The PRISM directive takes a polygon and deposits copies of it at specified intervals along the direction toward the desired ending point. A line may be redefined as a polygon and used to generate a rectangular plate. In addition, the line's length may be scaled as it is moved from interval to interval so as to make tapered plates. Figure 1(a) exhibits the use of the PRISM command for generating a rectangular plate. The original line, represented by the solid line, is separated into two segments to provide a width of two segments. The dashed lines represent the copies of the original line placed so that they divide the length of the plate into six sections. While the segmentation along the width of the plate is determined by the segmentation of the original line, the segmentation along the length is accomplished via a numerical argument in the PRISM command. The dotted lines represent the two other sides of the plate created by implementing PRISM. In Figure 1(b) a model of a wing is depicted. This structure was also created using the PRISM directive, but the length of the original line was scaled by a value less than unity to produce a narrowing width at each section and yield a tapered shape.

The Super 3D format does not differentiate between patches used for wings or patches used for the fuselage. However, the GEOM subroutine must decide whether the set of data it is given represents a cross-sectional polygon or a two-dimensional plate. If it is a cross-sectional polygon, the lines must be segmented, the polygon must be built and joined to the previous polygon, if one exists. If a two-dimensional plate must be constructed, then the lines building the plate must be segmented and the plate divided into as many triangles as needed to cover the surface area. To differentiate between a cross-sectional polygon and a two-dimensional plate, the Super 3D data file must be modified. The lines of data representing the two-dimensional plate must use a "plate" directive rather than the "polygon" directive normally used. This alteration is a necessity for the GEOM subroutine to perform correctly. If a

line is encountered, separate from other lines or polygons, then it is modeled as a wire monopole antenna.

The completion of the GEOM subroutine to transfer Super 3D data files to FERM data file marks a milestone. Now the identical geometry analyzed by the NEC and ESP can be studied with the FERM without inaccuracies resulting from the reconstruction of the structure.

IV. RESULTS

A. PATTERN PREDICTION

Figures 2 and 3 illustrate the geometry and overall dimensions of the Blackhawk helicopter, and the FERM model used in this analysis, respectively. A loop or towel-bar antenna is mounted beneath the tail of the Blackhawk helicopter. The loop had a length of 3.97 meters and a height of 0.4 meters. Because a strip antenna was implemented, a width of 4.0 cm, equivalent to a wire radius of 1.0 cm, was used [10]. The loop consists of only three strips with the metallic body of the helicopter comprising the fourth side. The antenna is fed along the strip closest to the front of the helicopter.

There are two problems associated with the FERM code in modeling wire loop antennas such as the one illustrated in Figure 3. The first is that two wires must be collinear or connected end-to-end. In generating a loop antenna, the wires cannot be placed end-to-end at the 90° bends. The second problem is that wires cannot be electrically attached to surfaces in the current version of the FERM code. To solve the first problem, the loop is created using a strip equivalent model. The strip can be electrically connected at right angles while the wires cannot. Figure 4 exhibits the differences between the wire and strip models at the corner of the loop antenna. The strip can also be electrically attached to the helicopter fuselage if there are two common nodes between the strip mesh and the fuselage mesh. This was not possible to do for the loop antenna which was modeled as a strip due to the changing size of the helicopter tail, and so the edges of the strip are just placed near the surface of the helicopter fuselage. This is not expected to influence much the computation of the antenna radiation patterns.

Figure 5 displays the radiation patterns of the loop antenna mounted on the Blackhawk helicopter at 10 MHz. FERM and NEC computed radiation patterns for the co-polarized yaw plane are compared. The patterns obtained by the two codes were normalized to 0 dB based on their individual maxima. The FERM exhibits a magnitude approximately 2 dB greater than that of the NEC around the $\phi = 90°$ and $\phi = 270°$. However, the results from the two codes agree well in the remaining portions of the pattern. In Figure 6, the co-polarized pitch plane radiation patterns are shown. Unlike the yaw plane, the agreement between the NEC and FERM in the pitch plane is not as good. The co-polarized roll plane is shown in Figure 7. The best agreement between the two codes is seen in this plane.

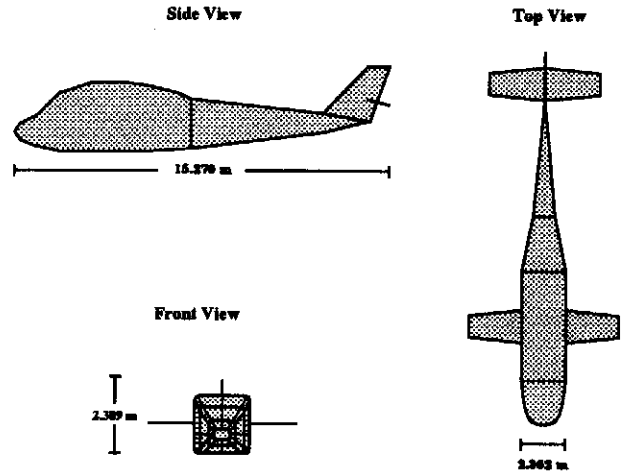


Figure 2: Geometry of the Blackhawk helicopter.

B. INPUT IMPEDANCE PREDICTION

For a strip antenna to be electrically connected to the helicopter fuselage or any other geometry, it must have a common edge with that geometry; i.e., a multiple edge between the strip antenna and the ground plane (or helicopter fuselage) must exist for the two to be electrically

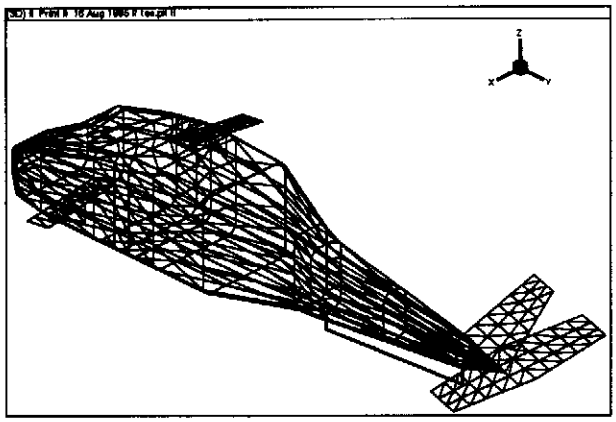


Figure 3: Geometry of the 4.0 m towel-bar antenna mounted on the segmented Blackhawk helicopter.

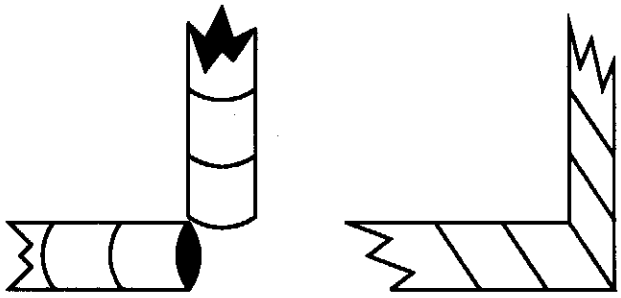


Figure 4: The modeling of a right angle of a loop antenna using (a) wires and (b) strips.

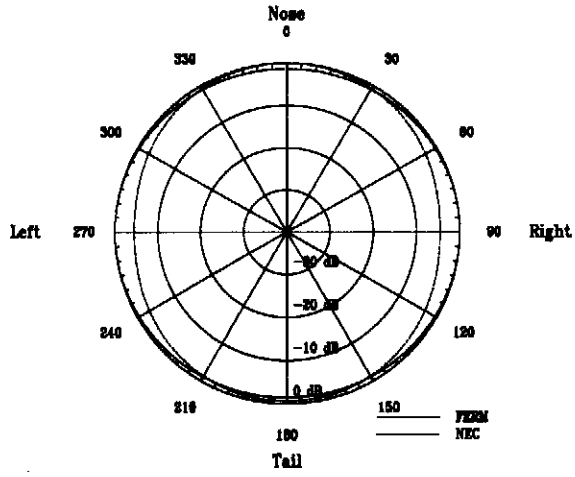


Figure 5: The co-polarized yaw plane radiation pattern of the loop antenna on the Blackhawk helicopter model at 10 MHz. There is no rotor blade structure.

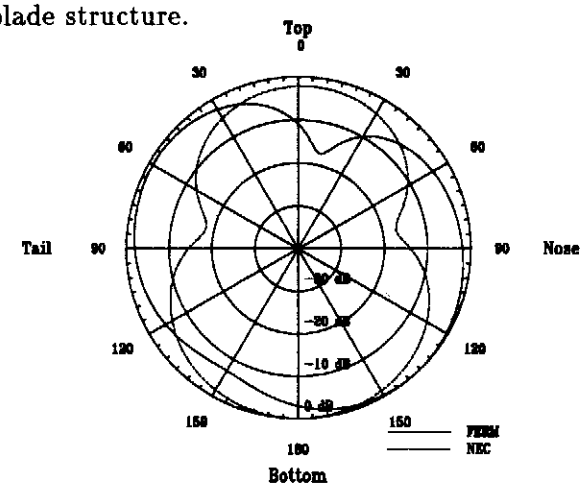


Figure 6: The co-polarized pitch plane radiation pattern of the loop antenna on the Blackhawk helicopter model at 10 MHz. There is no rotor blade structure.

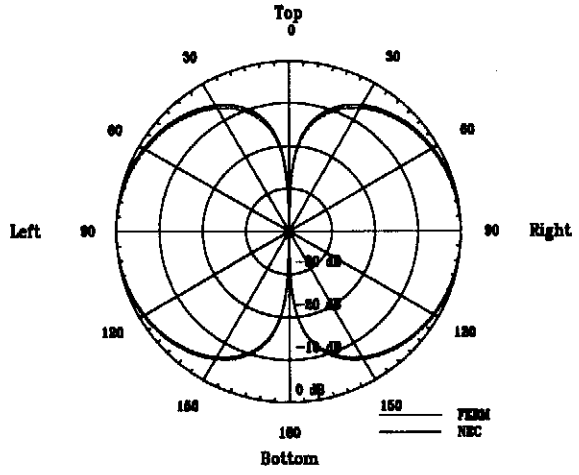


Figure 7: The co-polarized roll plane radiation pattern of the loop antenna on the Blackhawk helicopter model at 10 MHz. There is no rotor blade structure.

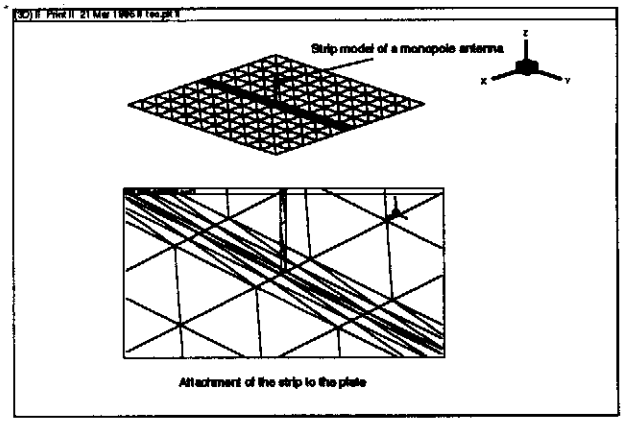


Figure 8: An example of a strip antenna connected to a ground plate for FERM modeling.

connected.

To illustrate how a strip antenna is properly connected to a ground, a simpler geometry is considered. The geometry is that of a $\lambda/4$ monopole on a $1 \times 1\lambda$ square ground plane at 100 MHz. Of interest in this problem is the calculation of the input impedance of the antenna. The geometry of the strip antenna and the way it is properly attached to the ground plate is illustrated in Figure 8. The thickness of the strip was chosen four times the wire radius (w=20 mm). To connect the strip to the ground plate the two must have a common edge along a triangular patch. The strip in this case was oriented in the xz-plane, and therefore, the discretization of the plate in the x direction was made nonuniform. This was required because the grid size in the x-direction near the middle of the plate (where the strip is attached) must be equal to the width of the strip. This is illustrated in Figure 8.

Using the simple geometry of the monopole on a square ground plate the input impedance of the antenna was calculated in the frequency range from 10 to 100 MHz. The FERM computed input resistance is compared with similar predictions using the NEC code in Figure 9, and the antenna input reactance in Figure 10. The results of having the strip antenna on top of the plate without being electrically connected to the plate are also included. As illustrated by both figures, there is only a few Ohms difference between the FERM predictions with the properly connected antenna and the NEC results.

It is expected that if the monopole antenna is properly attached to the helicopter fuselage then similar input impedance results will be obtained, especially at HF frequencies. It was previously demonstrated that the current distribution at HF frequencies is localized in the vicinity of the antenna, and that the platform used to mount the antenna is not expected to significantly affect the input impedance calculations [11].

C. EXECUTION TIMES OF THE FERM ON DIFFERENT COMPUTERS

A comparison of the execution times of the FERM code subroutines was accomplished to determine whether the IBM RISC 6000 is more efficient than the VAX 3100. Since the VAX 3100 workstation cannot handle the large scratch files generated for geometries with more than 1500 unknowns, the scope of the comparison was bounded by this upper limit. The four main numerical processing subroutines used in predicting antenna radiation patterns were considered: EFIE1, EFIE2, EFIE3, and EFIE4V. The results of this test are displayed in Table 1.

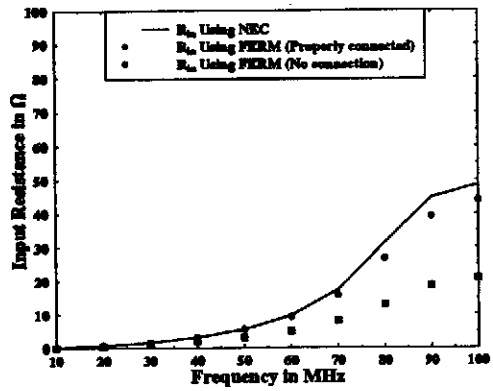


Figure 9: NEC and FERM computed input resistance of a strip antenna on a finite-size conducting ground plate, in the frequency range from 10 to 100 MHz.

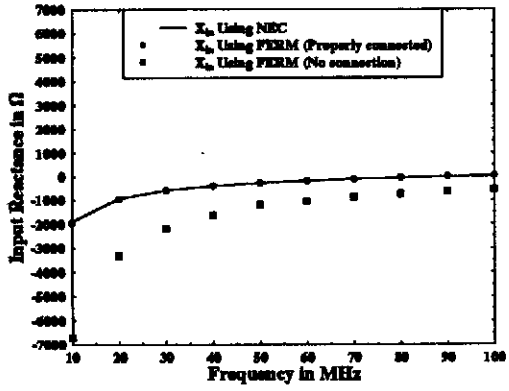


Figure 10: NEC and FERM computed input reactance of a strip antenna on a finite-size conducting ground plate, in the frequency range from 10 to 100 MHz.

Table 1: Execution times for FERM subroutines on various computers.

Comp. Type	No. of Unkn.	Execution Time (seconds)			
		EFIE1	EFIE2	EFIE3	EFIE4V
VAX	299	1.0	209.0	28.0	11.0
IBM	299	0.05	0.72	0.440	3.26
VAX	517	3.0	840.0	103.0	96.0
IBM	517	1.3	150.2	81.6	73.8
VAX	1201	10.0	4570	842	375
IBM	1201	4.7	1661.7	818.1	300.2

For the first two entries in the table, a $\lambda/4$ wire monopole located over a perfectly conducting ground plane was used for comparison. At the test frequency of 50 MHz, the physical length of the antenna was 1.5 meters while the sides of the ground plane were 6.0 meters in length. The diameter of the antenna was 1.5 cm. The gap between the antenna and the ground plane was matched to the diameter of the wire. The plate is segmented into 100 triangular pairs (ten segments per side). The wire antenna is divided into ten segments with the voltage excitation placed between the first and second antenna segments closest to the plate. This configuration spawned 299 unknowns.

There was little difference between execution times on the VAX and the IBM RISC for the first (EFIE1), third (EFIE3), and fourth (EFIE4V) subroutines. The VAX workstation produced results within 30 seconds of the IBM RISC for these subroutines. There was a substantial difference in run times for the second subroutine, EFIE2. This routine evaluates the electric field integral equation for the input geometry, creates an impedance matrix, and writes the matrix to a data file. This computation is of order N^2, where N is the number of unknowns. The IBM RISC performed this task in under 1 second of CPU time, while the VAX required approximately 200 seconds of CPU time. Since the number of unknowns is relatively small, the amount of time needed to

write to and read from the scratch file TEMP is not significant. Therefore, the difference in execution times for the second subroutine can be attributed to the IBM RISC's more efficient numerical evaluation of the electric field integral equations.

A block helicopter modeled with a strip antenna, shown in Figure 11(b), provided the next set of unknowns for comparison. The test frequency for this geometry was established as 70 MHz. The block helicopter length (4.0 m) was partitioned into 10 segments. The $\lambda/4$ (1.07 m) antenna also had 10 segments with a 1.0 volt excitation located between the first and second segments closest to the fuselage. This geometry yielded 517 unknowns for processing on the two workstations.

This portion of the examination was very similar to the test runs with 299 unknowns. The VAX workstation had execution times 22-23 seconds slower than those of the IBM RISC for the EFIE3 and EFIE4V subroutines. EFIE2 took 5.6 times as long to execute on the VAX as it did on the IBM RISC. This case differed from the first in that EFIE2 had to use disk memory to complete the numerical evaluation on the VAX. This process slows the execution time substantially on the VAX. More on this subject will be seen as the number of unknowns increases towards the upper limit.

The final geometry used for this comparison was a modified version of the Blackhawk helicopter in Figure 3. The segmentation along the length of the helicopter was decreased as was the segmentation in the wings and stabilizers. This was done so that the number of unknowns would remain under 1500. This modified helicopter geometry produced 1201 unknowns.

As the number of unknowns was quadrupled from 299 to 1201, the difference in CPU times for the EFIE3 subroutine remained constant. EFIE3 is the subroutine that implements LU decomposition to decompose the impedance matrix created by the EFIE2 subroutine. Similarly, the differences in execution times for the EFIE1 and EFIE4V increased very slightly with the increase in the number of unknowns. The 5.3 second difference between the VAX's run time and the IBM RISC's run time for the EFIE1 subroutine is insignificant. As the number of unknowns was increased from 517 to 1201, the execution time on the IBM RISC for the second subroutine increased by more than an order of magnitude. Meanwhile, the VAX's execution time for the same subroutine only increased by a factor of five. Clearly, for the case of 1201 unknowns, the difference in CPU time between VAX and IBM RISC increased, but their ratio is decreasing. This suggests that as the number of unknowns increases, the efficiency of the VAX workstation approaches that of the IBM RISC workstation. However, this conclusion cannot be completely verified because of the memory limitations on the VAX 3100 workstation. For geometries with less than 1200 unknowns, the IBM RISC 6000 is a better computer platform for execution of the FERM code because it operates more efficiently. Yet, for geometries with a larger number of unknowns, the IBM RISC 6000 is still better, because of the substantially larger amount of memory required by the temporary scratch files written during the numerical evaluation of the electric field integral equations in the subroutine EFIE2.

V. CONCLUSIONS

The Finite Element Radiation Model code is a Method of Moment solver used primarily for radar cross section prediction and antenna modeling. The code has been applied in this paper to predict the antenna radiation patterns of a loop mounted on the Blackhawk helicopter and the input impedance of a monopole antenna on a ground plate. The results have shown that when modeling a loop or towel-bar antenna with the FERM code, the antenna must be comprised of thin strips rather than wires. This ensures that the corners of the loop are connected both physically and electrically. The width of the strips was taken to be four times the equivalent radius of a wire with circular cross-section [10]. However, there are some difficulties in properly attaching the loop edges to the heli-

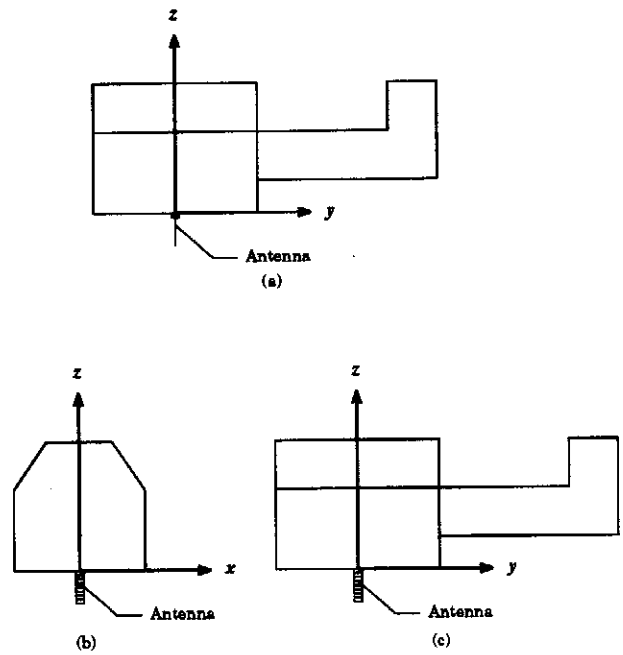


Figure 11: Antenna geometries on the block helicopter model: (a) wire antenna, (b) strip antenna with the width along the x-axis, and (c) strip antenna with the width along the y-axis.

copter surface due to the complexity of the finite element mesh. To alleviate this problem the loop antenna was placed near the helicopter surface. The radiation patterns resulting from this analysis were compared with similar ones obtained using the NEC code. For antenna input impedance calculations a simpler structure was considered, that of a monopole on a ground plate. It is demonstrated that accurate input impedance results can be obtained when the antenna is properly grounded on the plate.

References

[1] S.M. Rao, D.R. Wilton, and A.W. Glisson, "Electromagnetic scattering by surfaces of arbitrary shape," *IEEE Trans. Antennas Propag.*, AP-30, 409-418, May 1982.

[2] S. Lee, D.A. Shnidman, and F.A. Lichauco, *Numerical Modeling of RCS and Antenna Problems*, Tech. Report 785, MIT Lincoln Laboratory, Lexington, Massachusetts, Dec. 1987.

[3] E.H. Newman, *A User's Manual for Electromagnetic Surface Patch Code: ESP Version 4*, The Ohio State University, ElectroScience Lab., Rep. No. 716199-11, Aug. 1988

[4] *Guide to Programming on VAX/VMS (FORTRAN edition)*, Fourth Edition, Digital Equipment Corp., Maynard, MA, Sept. 1984.

[5] *VS FORTRAN Version 2: Language and library Reference*, Third Edition, IBM Corp., San Jose, CA, June 1987.

[6] Super 3D, copyrighted by Silicon Beach Software, 1986.

[7] J. Peng, J. Choi, and C.A. Balanis, "Automation of the geometry data for the NEC and ESP using Super 3D," *Proc. 6th Annual Progress Review Appl. Comp. Electromag.*, pp.33-38, Mar. 1990.

[8] J. Peng, *Guidelines and Automation of Antenna Patterns of Complex Structures*, M.S. Thesis, Arizona State University, Aug. 1992.

[9] GEOMVIEW, copyrighted by the Geometry Center, University of Minnesota, 1994.

[10] C.A. Balanis, *Antenna Theory: Analysis and Design*. New York, New York: Harper and Row, 1982.

[11] W. Andrew, C.A. Balanis, C. Birtcher, P.A. Tirkas, "Finite-difference time domain of HF antennas," *1994 IEEE MILCOM Conf. Rec.* (Ft. Monmouth, NJ), pp. 7-11, Oct. 1994.