MIDDLE EAST TECHNICAL UNIVERSITY
FACULTY OF ENGINEERING
ELECTRICAL ENGINEERING DEPARTMENT

NUMERICAL AND EXPERIMENTAL INVESTIGATION OF THE SCATTERING OF ELECTROMAGNETIC WAVES BY PERFECTLY CONDUCTING ENSEMBLES

Submitted to the Faculty of Engineering for Partial Fulfillment of the Requirements for the Degree of Master of Science in Electrical Engineering

by

FATİH CANATAN

Ankara, Turkey
November, 1971

Approved by :

_____

_____

_____

_____

_____

ERRATA

PAGE	PARAG.	LINE.	WRONG FORM	CORRECTED AS
Abstract	1	2	and	an
Abstract	1	3	Numerical	The numerical
Abstract	3	1	of	with
Abstract	4	2	scandard	standard
Abstract	4	3	A satisfactory	Satisfactory
L.of Pr.S.		1	PRINCIPLE	PRINCIPAL
L.of Pr.S.		4	Cronecker's	Kronecker's
L.of Pr.S.		5	Indical	Indicial
L.of Pr.S.		15-17	Associate	Associated
Table of C.		4	Surway	Survey
Table of C.		12	Associate	Associated
Table of C.		14	Indical	Indicial
Table of C.		16	19	18
1	1	5	total	the total
1	1	6	seperated	separated
1	1	6	Two as the	
1	1	7	the	
1	1	7	if	
1	1	8	a	
1	2	Head.	SURWAY	SURVEY
1	2	7	applying	by applying
3	3	2	sligth	slight
3	4	3	complately	completely
7	EQ(2.1.19)		$\bar{n} \times (\bar{H}_{in} + \bar{H}_{sc}) = \bar{H}_I$	$(\bar{H}_{in} + \bar{H}_{sc}) = \bar{H}_I$
10	EQS(2.2.1-2)		$\bar{K}(\bar{r}')$	$\bar{n}' \times \bar{H}_I(\bar{r}')$
13	1	Head.	ASSOCIATE	ASSOCIATED
13	1	3	Associate	Associated
13	1	4	associatable to	associated with
13	1	5	Following	The following
15	4	2	enlargening	increasing
16	1	Head.	INDICAL	INDICIAL
35	3	4	partionize	partition
36	1	2	partioned	partitioned
40	2	4	compatable	compatible
40	2	4	available	the available
47	1	7	increase	increases
47	1	7	disadvatageous	disadvantageous
47	1	9	increase	increases
47	2	4-6	complate	complete
48	2	1	to	of
48	3	7	experiental	experimental
49	2	12	Associate	Associated
55	1	1	Abromowitz	Abramowitz
Page numbers			18,19,17	17,18,19

ACKNOWLEDGEMENTS

This work has been carried out in the laboratories of the Department of Electrical Engineering and the Department of Computer Science. The author is indebted to the members of those departments of the Middle East Technical University, who have been very helpful with their enlightening advices. He wishes to acknowledge with great gratitude the supervision of Dr. Altunkan Hızal and the valuable comments of Dr. Tuncay Birand.

Acknowledgements are also due to his friends Mr. Haluk Tosun, Mr. Tuncay Ege, Mr. Zafer Yasa, Mr. Edip Niver who have been encouraging all the time and especially Mr. Alev Erlevent. The author is indebted to Destek Matbaacılık Ticaret ve Sanayii for the printing of the thesis and Yüksek Teknik Öğretmen Okulu for the xerox duplicates of the program outputs.

ABSTRACT

In this thesis, the problem of electromagnetic scattering from and ensemble of perfectly conducting smooth bodies is investigated numerically and experimentally. Numerical part of the work done is based on a theory which employes multipole expansion techniques.

The scattering problem is considered as a problem of radiation from the induced surface sources. The re-radiated fields are expanded in terms of spherical vector wave functions. The unknown multipole coefficients are determined from the linear relations obtained from the properties of the spherical vector wave functions,expansion of the incident field and surface current relations.

Numerical techniques are developed for use of a general ensemble of arbitary bodies. Due to difficulties arising from the limitations of the available computer , toroidally symmetric ensembles are investigated using a similar technique to the one used for the general problem.

Back-scattering cross-sections are measured by a method based on comparison of thescattering from scandard scatterers (i.e.spheres) and the ensembles. A satisfactory agreement is obtained with numerical results.

LIST OF PRINCIPLE SYMBOLS

$\bar{a}, \bar{u}$	Base Vectors
$\alpha E, \alpha M, X, Y$	Multipole Coefficients
δ_{ij}	Cronecker's Delta
$\Delta, \gamma, \xi, \eta$	Indical Functions
G_0	Free Space Green's Function
$\bar{H}$	Magnetic Field
$H_J^{(1)}(r)$	Hankel Functions(Spherical)
h	Scale Factors
$\overleftrightarrow{I}$	Identity Dyadic
JE, JM, CE, CM	Scattering Matrix Submatrices
$J_k(r)$	Spherical Bessel's Function
k	Propagation Constant
L	Lower Triangle Matrix
$P_K^m(\cos\theta)$	Associate Legendre Functions
P	Permutation Matrices
$Q_K^m(\theta)$	Associate Legendre Functions Divided by $\sin\theta$
$(r,\theta,\varphi), (\rho,\beta,\varphi), (x,y,z)$	Coordinate Variables
S	Transformation Matrix
S	Scattering Matrix
$\sigma(\theta,\varphi)$	Re-radiation Diagram
σ_b	Back Scattering Cross-section
U	Upper Triangle Matrix
U	Excitation Matrix
$\bar{X}(\theta,\varphi)$	Angular Vector Function
$Z(r)$	Radial Function

TABLE OF CONTENTS

ILLUSTRATIONS AND TABLES

1. INTRODUCTION

1.1. DESCRIPTION OF THE PROBLEM

Electromagnetic fields, produced by time varying current and charge distributions localizedin some region of an unbounded homogenous medium, can be calculated by the inhomogoneous wave equation. If perfectly conducting ensembles are present in the free space subject to illuminating electromagnetic waves ,total wave distribution can be seperated into two as the incident and the re-radiated fields. Re-radiated fields may be thought as if a radiation due to induced surface sources. The problem is , given the incident wave and the obstacle, to find the re-radiated fields. The incident wave may be thought as a collection of monochromatic waves, in particular plane waves.

1.2. A BRIEF SURWAY OF THE PREVIOUS TECHNIQUES AND THE METHOD USED

In classical scattering problems, generally, the shapes of the bodies of concern are taken to be coincident with some coordinate system, thus the method of seperation of variables is easily applicable. By this method, only single bodies were investigated. The incident and the scattered fields are expanded in terms of the vector wave functionsand the unknown coefficients are determined applying boundary conditions for the surfaces .

For instance, in the case of a sphere, the scattering coefficients can be found in closed form [1], [2].

An exact formulation by means of an integral equation was given by Maue and a derivation of this equation is given by Van Bladel [2]. For a three dimensional problem, integral equation solution by ordinary numerical methods is hard. Too many grid points are necessary for this and a convergent iterative scheme has to be developed. For every excitation this scheme must be repeated from the beginning.

Another method is to approximate the surface currents by the first N terms of an infinite series, thus it is possible to obtain N linear algebraic equations. This method was used by Mei and Bladel [3].

Waterman has developed a theory based on the Rayleigh [4] hypothesis, which is approximate for shapes other than a sphere.

For multibodies, the most significant research is by John H.Bruning and Yuen T.Lu [5]. They made theoretical,numerical and experimental investigations on the problem of two spheres illuminated by a plane wave. A method of shifting has been used for the vector wave functions. Their experiments are also worth mentioning. With the equipment of the Radiation Laboratory , Illinois, they have made perfect measurements on every case of two spheres.

The only significant research on toroidal bodies was made for thin wire loops, by Angelakos [6].

Victor Twersky [7] proposed a method for the general scattering problem which makes use of the isolated scatterer values to find multipole coefficients. This paper is purely theoretical, and in practice no numerical research is made.

The method proposed by Altunkan Hızal [8] in 1970 is employed in this research with a sligth modification. The method and the modification is given in Chapter 2 . Though he has proposed the theory, the obtained results were for isolated rotational bodies illuminated with a plane wave in the direction of the axis of revolution.

In this thesis an introduction to numerical analysis of the arbitary multibody problem is made and toroidal and rotational configurations are investigated complately.

2. FORMULATION OF THE PROBLEM

2.1. ORIGIN EXCLUSIVE TYPE OF FORMULATION

It is assumed that a finite number of smooth, conducting scatterers are existing within the limits $r_0 < r < r_1$ with respect to some coordinate frame. For $r > r_1$ and $r < r_0$ the incident and the scattered fields are investigated seperately, but for the bulky region, $r_0 < r < r_1$, the total field is considered (Fig. 2.1.1).

Since the bodies are assumed to be perfect conductors, re-radiated field is given by the integral equation

$$\bar{H}_{sc}(\bar{r}) = \nabla \times \oint_S \bar{n}' \times \left(\bar{H}_{in}(\bar{r}') + \bar{H}_{sc}(\bar{r}')\right) G_0(\bar{r}|\bar{r}')\, dS' \qquad (2.1.1)$$

where S is the total surface of the bodies, $\bar{H}_{sc}(\bar{r})$ is the re-radiated field at $\bar{r}$, $\bar{H}_{in}(\bar{r})$ is the incident field (its sources are assumed to be free from interractions), $G_0(\bar{r},\bar{r}')$ is the free space Green's function which can be expressed as

$$G_0(\bar{r}|\bar{r}') = \frac{e^{ik|\bar{r}-\bar{r}'|}}{4\pi|\bar{r}-\bar{r}'|} \qquad (2.1.2)$$

where $k = \omega / c = 2\pi/\lambda$, is the propagation coefficient.

Green's function can be treated as a dyadic since $\bar{F}\, G_0 = \bar{F} \cdot G_0 \overleftrightarrow{I}$. This dyadic can be expanded in terms of vector wave functions as follows [8]

For $r > r'$

$$G_0(\bar{r}|\bar{r}')\overleftrightarrow{I} = \sum_{p,m} ik \left\{ \bar{M}^{1*}_{p,m}(\bar{r}')\, \bar{M}^{s}_{p,m}(\bar{r}) + \frac{1}{k^2} \bar{N}^{1*}_{p,m}(\bar{r}')\, \bar{N}^{s}_{p,m}(\bar{r}) \right\} \qquad (2.1.3)$$

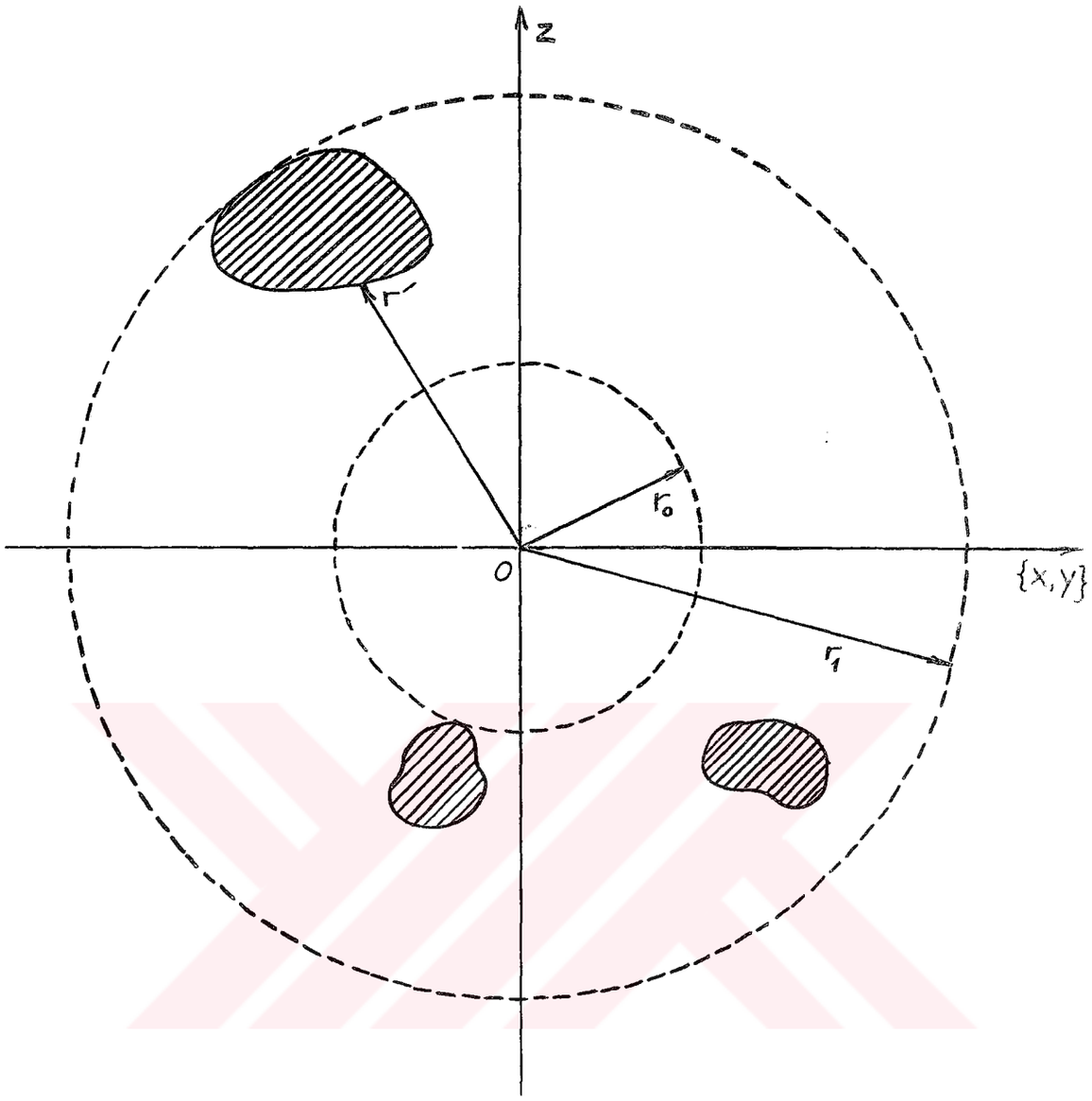

Fig 2.1.1.Scattering Geometry for Multibody Scattering.

For $r<r'$,

$$G_0(\bar{r}|\bar{r}')\overleftrightarrow{I}=\sum_{p,m} ik\left\{\bar{M}^{1}_{p,m}(\bar{r})\,\bar{M}^{2*}_{p,m}(\bar{r}')+\frac{1}{k^2}\bar{N}^{1}_{p,m}(\bar{r})\,\bar{N}^{2*}_{p,m}(\bar{r}')\right\} \qquad (2.1.4)$$

where the vector wave functions M^i_{pm} and N^i_{pm} are given as

$$\bar{M}^{i}_{p,m}(\bar{r})=Z^{i}_{p}(kr)\,\bar{X}_{p,m}(\theta,\varphi)=\frac{1}{k^2}\nabla\times\bar{N}^{i}_{p,m}(\bar{r}) \qquad (2.1.5)$$

$$\bar{N}^{i}_{p,m}(\bar{r})=\nabla\times\bar{M}^{i}_{p,m}(\bar{r}) \qquad (2.1.6)$$

where the superscript takes the values 1,2,s. The general radial function $Z^i_p(kr)$ replaces the following spherical functions :

$$Z^{1}_{p}(kr)=J_p(kr) \qquad (2.1.7)$$

$$Z^{2}_{p}(kr)=H^{2}_{p}(kr)=H^{1*}_{p}(kr) \qquad (2.1.8)$$

$$Z^{s}_{p}(kr)=H^{1}_{p}(kr)=H^{2*}_{p}(kr) \qquad (2.1.9)$$

where J_p is the spherical Bessel function and H^1_p ,H^2_p are the spherical Hankel functions of the first and the second kind . X_{pm} is the vector spherical harmonic and it can be written as

$$\bar{X}_{p,m}(\theta,\varphi)=\frac{\bar{r}\times\nabla}{i\sqrt{p(p+1)}}\sqrt{\frac{(2p+1)\,(p-m)!}{4\pi\;\;(p+m)!}}\;P^{m}_{p}(\cos\theta)\,e^{im\varphi} \qquad (2.1.10)$$

When Eqs (2.1.3) and (2.1.4) are substituted in (2.2.1) two different multipole expansions for the re-radiated fields are found, depending on r and r'. These are

For $r<r_0$ $$\bar{H}_0=\sum_{p,m}\left(\alpha EO_{p,m}\bar{M}^{1}_{p,m}-\frac{i}{k}\alpha MO_{p,m}\bar{N}^{1}_{p,m}\right) \qquad (2.1.11)$$

For $r>r_1$ $$\bar{H}_s=\sum_{p,m}\left(\alpha ES_{p,m}\bar{M}^{s}_{p,m}-\frac{i}{k}\alpha MS_{p,m}\bar{N}^{s}_{p,m}\right) \qquad (2.1.12)$$

where the multipole coefficients are equal to :

$$\alpha EO_n = ik\oint_s \bar{n}' \times (\bar{H}'_{in} + \bar{H}'_{sc}) \cdot \bar{N}'^{2*}_{p,m}\, ds' \tag{2.1.13}$$

$$\alpha MO_n = -k^2 \oint_s \bar{n}' \times (\bar{H}'_{in} + \bar{H}'_{sc}) \cdot \bar{M}'^{2*}_{p,m}\, ds' \tag{2.1.14}$$

$$\alpha ES_n = ik\oint_s \bar{n}' \times (\bar{H}'_{in} + \bar{H}'_{sc}) \cdot \bar{N}'^{1*}_{p,m}\, ds' \tag{2.1.15}$$

$$\alpha MS_n = -k^2 \oint_s \bar{n}' \times (\bar{H}'_{in} + \bar{H}'_{sc}) \cdot \bar{M}'^{1*}_{p,m}\, ds' \tag{2.1.16}$$

All these integrals are over the total surface of the ensemble scatterers. For the region $r_0 < r < r_1$ it can also be written similar expansions, but since at both sidessecondary sources exist the proper form is

$$\bar{H}_I = \sum_{p,m} \left\{ \left(\alpha EI^1_{p,m} \bar{M}^1_{p,m} - \frac{i}{k} \alpha MI^1_{p,m} \bar{N}^1_{p,m}\right) + \left(\alpha EI^2_{p,m} \bar{M}^2_{p,m} - \frac{i}{k} \alpha MI^2_{p,m} \bar{N}^2_{p,m}\right) \right\} \tag{2.1.17}$$

But unfortunately, these multipole coefficients cannot be given by simple integrals as before. Therefore they should be obtained by boundary relationships.Taking the incident field $\bar{H}_{in}$ can be written as an expansion of the form

$$\bar{H}_{in} = \sum_{p,m} \left(\alpha Ei_{p,m} \bar{M}^1_{p,m} - \frac{i}{k} \alpha Mi_{p,m} \bar{N}^1_{p,m}\right) \tag{2.1.18}$$

As it was mentioned before, it is assumed that no interractions occur with the scattered fields. Then

$$\bar{n} \times (\bar{H}_{in} + \bar{H}_{sc}) = \bar{H}_I \tag{2.1.19}$$

Assuming that the series converge, a finite number of terms are enough to meet a certain precission. Smoothness assures lack of singularity for the domain $\bar{r}'$ and (2.1.19) with(2.1.13-16) can be written as an array relation

$$\begin{bmatrix} IE^{11}_{pmp'm'} & IM^{11}_{pmp'm'} & IE^{12}_{pmp'm'} & IM^{12}_{pmp'm'} \\ CE^{11}_{pmp'm'} & CM^{11}_{pmp'm'} & CE^{12}_{pmp'm'} & CM_{pmp'm'} \\ IE^{21}_{pmp'm'} & IM^{21}_{pmp'm'} & IE^{22}_{pmp'm'} & IM^{22}_{pmp'm'} \\ CE^{21}_{pmp'm'} & CM^{21}_{pmp'm'} & CE^{22}_{pmp'm'} & CM^{22}_{pmp'm'} \end{bmatrix} \begin{vmatrix} \alpha EI^{1}_{p'm'} \\ \alpha MI^{1}_{p'm'} \\ \alpha EI^{2}_{p'm'} \\ \alpha MI^{2}_{p'm'} \end{vmatrix} = \begin{vmatrix} \alpha MS_{p,m} \\ \alpha ES_{p,m} \\ \alpha MO_{p,m} \\ \alpha EO_{p,m} \end{vmatrix} \tag{2.1.20}$$

The elements of the square matrice can be written as

$$IE^{ij}_{pmp'm'} = k^2 \oint_s (\bar{M}'^{i*}_{p,m} \times \bar{M}'^{j}_{p',m'}) \cdot \bar{n}' ds' \tag{2.1.21}$$

$$IM^{ij}_{pmp'm'} = -ik \oint_s (\bar{M}'^{i*}_{p,m} \times \bar{N}'^{j}_{p',m'}) \cdot \bar{n}' ds' \tag{2.1.22}$$

$$CE^{ij}_{pmp'm'} = -ik \oint_s (\bar{N}'^{i*}_{p,m} \times \bar{M}'^{j}_{p',m'}) \cdot \bar{n}' ds' \tag{2.1.23}$$

$$CM^{ij}_{pmp'm'} = - \oint_s (\bar{N}'^{i*}_{p,m} \times \bar{N}'^{j}_{p',m'}) \cdot \bar{n}' ds' \tag{2.1.24}$$

These entities have interesting properties and the essential ones can be derived from Green's second vector identity[8]and vector wave functions,these are

$$IE^{i,j}_{p,m,p',m'} = CM^{i,j}_{p,m,p',m'} \tag{2.1.25}$$

$$IM^{i,j}_{p,m,p',m'} = - CE^{j,i}_{p,m,p',m} \tag{2.1.26}$$

$$IE^{j,i}_{p,m,p',m'} = - (IE^{i,j}_{p',m',p,m})^* \tag{2.1.27}$$

$$IM^{j,i}_{p,m,p',m'} = - (IM^{i,j}_{p',m',p,m})^* \tag{2.1.28}$$

where i=1,2 and j=1,2 only.

These properties are very important for testing the numerical results and increasing the computational efficiency.

Continuity of the total electromagnetic field accross the spherical boundaries, orthogonality property of vector wave functions on a spherical surface give

$$\begin{bmatrix} \alpha MS_{p,m} \\ \alpha ES_{p,m} \\ \alpha MO_{p,m} \\ \alpha EO_{p,m} \end{bmatrix} = \begin{bmatrix} 0 & \delta_{pp'}\delta_{mm'}A_2^{p} & 0 & \delta_{pp'}\delta_{mm'}A_3^{p} \\ \delta_{pp'}\delta_{mm'}A_2^{p} & 0 & \delta_{pp'}\delta_{mm'}A_3^{p} & 0 \\ 0 & \delta_{pp'}\delta_{mm'} & 0 & \dfrac{\delta_{pp'}\delta_{mm'}}{A_1} \\ \delta_{pp'}\delta_{mm'} & 0 & \dfrac{\delta_{pp'}\delta_{mm'}}{A_1} & 0 \end{bmatrix} \begin{bmatrix} \alpha EI^{1}_{p',m'} \\ \alpha MI^{1}_{p',m'} \\ \alpha EI^{2}_{p',m'} \\ \alpha MI^{2}_{p',m'} \end{bmatrix} - \begin{bmatrix} A_2^{p}\alpha Mi_{p,m} \\ A_2^{p}\alpha Ei_{p,m} \\ \alpha Mi_{p,m} \\ \alpha Ei_{p,m} \end{bmatrix} \qquad (2.1.29)$$

where

$$A_1^{p} = J_p(kr_0)/H_p^{2}(kr_0) \qquad (2.1.30)$$

$$A_2^{p} = J_p(kr_1)/H_p^{1}(kr_1) \qquad (2.1.31)$$

$$A_3^{p} = H_p^{2}(kr_1)/H_p^{1}(kr_1) \qquad (2.1.32)$$

Equations (2.1.20) and (2.1.19) reduce the problem to a simple system of equations of the form

$$Y = S\,X = B\,X + U \qquad (2.1.33)$$

where S,B and U are named as scattering,boundary and excitation matrices,respectively. For example the U matrix entities for a circularly polarized plane wave progressing in the positive z direction is given by [8]:

$$\alpha Ei_{p,m} = -m\,i^{p+1}\left(4\pi(2p+1)\right)^{1/2} \qquad (2.1.34)$$

$$\alpha Mi_{p,m} = i^{p+1}\left(4\pi(2p+1)\right)^{1/2} \qquad (2.1.35)$$

where m=+1 corresponds left hand polarization, m=-1 right hand polarization. Although $m=\pm 1$ this does not imply $m'=\pm 1$. The values of m' depend on the shape of the ensemble, too . Thus the system is coupled and cannot be partionized into small systems. This is possible for rotational ensembles and for this case m' takes the values of m only.

2.2. MODIFICATION OF THE THEORY FOR INCLUSION OF THE ORIGIN

If the origin is included by one of the bodies, the previous formulation fails since for the region $r<r_o$ s type expansion is not valid. In the sphere of radius r_o and on it the total field must vanish, then the following equations can be written considering the orthogonality of the spherical vector wave fuctions on a spherical surface :

$$\alpha E_{in} + ik \oint_S \frac{4\pi}{c} \bar{K}(\bar{r}') \cdot \bar{N}_n^{2*}(\bar{r}')\, ds' = 0 \qquad (2.2.1)$$

$$\alpha M_{in} - k^2 \oint_S \frac{4\pi}{c} \bar{K}(\bar{r}') \cdot \bar{M}_n^{2*}(\bar{r}')\, ds' = 0 \qquad (2.2.2)$$

With previous definitions, the boundary matrix B becomes :

$$B = \left|\begin{array}{c|c|c|c} 0 & \delta_{pp'}\delta_{mm'} A_2^p & 0 & \delta_{pp'}\delta_{mm'} A_3^p \\ \hline \delta_{pp'}\delta_{mm'} A_2^p & 0 & \delta_{pp'}\delta_{mm'} A_3^p & 0 \\ \hline 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \end{array}\right| \qquad (2.2.3)$$

On the other hand a careful investigation of Green's identities with exclusion of singularities [8] yield the following symmetry relations :

$$IE^{iJ}_{nn'} = CM^{iJ}_{nn'} \quad ; \quad n = p, m \tag{2.2.4}$$

$$IM^{11}_{nn'} = -CE^{11}_{nn'} \tag{2.2.5}$$

$$IM^{22}_{nn'} = -CE^{22}_{nn'} - 2\,\delta_{nn'} \tag{2.2.6}$$

$$IM^{iJ}_{nn'} = -CE^{iJ}_{nn'} - \delta_{nn'} \quad ; \quad i \neq J \tag{2.2.7}$$

$$[IE^{Ji}_{nn'}] = -[(IE^{iJ}_{nn'})^*]^T \tag{2.2.8}$$

$$[IM^{11}_{nn'}] = -[(IM^{11}_{nn'})^*]^T \tag{2.2.9}$$

$$[IM^{22}_{nn'}] = -[(IM^{22}_{nn'})^*]^T - 2\,\delta_{nn'} \tag{2.2.10}$$

$$[IM^{Ji}_{nn'}] = -[(IM^{iJ}_{nn'})^*]^T - \delta_{nn'} \quad ; \quad i \neq J \tag{2.2.11}$$

Again these equations can be used as a check for the results or a short cut for the algorithms. The coupledness of the equations is still present and to obtain uncoupled equations the configuration must be rotationary again.

3. NUMERICAL ANALYSIS OF THE PROBLEM

3.1. INTRODUCTION

Numerical part of the problem is the evaluation of the necessary functions, taking surface integrals and the solution of the scattering equations. Though it seems very easy, from the computer programming point of view there exist handicaps which limit the choice of the method.

The most essential dilemma of the problem is the question of need for time-core efficiency and the precision required for the results. Short-cut algorithms are necessary. Selection of the method is very important. On the other hand, since the core size of the available computer is limited, storage strategy is vital. Though external storage devices(disk and tape) are available excessive usage of them increases the execution time. Since at least at the primary steps , double precision is necessary,situation turns out to be a cross word puzzle, since double precision increases core demand and execution time considerably.

For toroidal bodies, there is not much trouble. If the lower ilumination with a circularly polarized plane wave is considered, everything becomes easy. Hovever generality of the problem reduces.

3.2 GENERATION OF ASSOCIATE LEGENDRE FUNCTIONS

The first step in the evaluation of the scattering integrals is the preceise generation of the necessary functions. Associate Legendre functions have very nice recursive properties, their derivatives are also associatable to the [18] functions in a simple fashion. Following property of them is quite important:

$$\gamma_p^{-m} P_p^{-m}(\cos\theta) = \gamma_p^{m} P_p^{m}(\cos\theta) \qquad (3.2.1)$$

For lower illumination of the toroidal configurations with circularly polarized plane waves, m=1 case will yield the results of m=-1 case with slight modifications.

First two values have to be evaluated from explicit expressions :

$$P_1^1(\cos\theta) = -\sin\theta \qquad (3.2.2)$$

$$\frac{d}{d\theta} P_1^1(\cos\theta) = -\cos\theta \qquad (3.2.3)$$

$$P_2^1(\cos\theta) = -3\sin\theta\cos\theta \qquad (3.2.4)$$

$$\frac{d}{d\theta} P_2^1(\cos\theta) = -3(\cos^2\theta - \sin^2\theta) \qquad (3.2.5)$$

Other ones are generated by the recursive formulas :

$$P_k^1(\cos\theta) = \frac{(2k-1)}{(k-1)} P_{k-1}^1(\cos\theta)\cos\theta - \frac{k}{k-1} P_{k-2}^1(\cos\theta) \qquad (3.2.6)$$

$$\frac{d}{d\theta} P_k^1(\cos\theta) = k\cot\theta\, P_k^1(\cos\theta) - \frac{k+1}{\sin\theta} P_{k-1}^1(\cos\theta) \qquad (3.2.7)$$

For a case where m takes other values too, there exist other recursion relations which may be found in any mathematical handbook(i.e. 'Mathematical Handbook' [20],

Speigel, McGraw Hill Book Co.,1968)

For $\theta = 0$ $P_k^1(\cos\theta) = 0$, k= 1,2,3...n. But in the analysis of scattering cross-sections $P_k^1(\cos\theta)/\sin\theta$ has to be evaluated. Since the computer cannot take the limit for 0/0 case it is necessary to modify the previous equations as :

$$Q_k(\theta) = P_k^1(\cos\theta)/\sin\theta \tag{3.2.8}$$

$$Q_1(\theta) = -1 \tag{3.2.9}$$

$$Q_k(\theta) = \frac{(2k-1)}{k} Q_{k-1}(\theta)\cos\theta - \frac{k}{k-1} Q_{k-1}(\theta) \tag{3.2.10}$$

$$Q_2(\theta) = -3\cos\theta \tag{3.2.11}$$

$$\frac{d}{d\theta} P_1^1(\cos\theta) = -\cos\theta \tag{3.2.12}$$

$$\frac{d}{d\theta} P_2^1(\cos\theta) = -3(\cos^2\theta - \sin^2\theta) \tag{3.2.13}$$

$$\frac{d}{d\theta} P_k^1(\cos\theta) = k\cos\theta\, Q_k(\theta) - (k+1)Q_{k-1}(\theta) \tag{3.2.14}$$

3.3 GENERATION OF SPHERICAL BESSEL,NEUMAN,HANKEL FUNCTIONS

Spherical Bessel,Neuman,Hankel function also offer some recursive relations. For Bessel and Neumann functions first two values are calculated from :

$$J_0(x) = \sin x / x \tag{3.3.1}$$

$$Y_0(x) = -\cos x / x \tag{3.3.2}$$

$$J_1(x) = (J_0(x)/x + Y_0(x)) \tag{3.3.3}$$

$$Y_1(x) = (Y_0(x)/x - J_0(x)) \tag{3.3.4}$$

Both functions and any linear combination of them-such as Hankel functions-have the following recursive properties :

$$f_m(x) = \frac{2m-1}{x} f_{m-1}(x) - f_{m-2}(x) \tag{3.3.5}$$

$$\frac{d}{dx} x f_m(x) = x f_{m-1}(x) - m f_m(x) \tag{3.3.6}$$

By means of these equations all of the radial functions can be evaluated easily. In Fortran , since computation by means of proper equivalencing real and complex quantities allows direct references to real and imaginary parts, it may be useful to generate only one of the Hankel functions. For this again first two values have to be evaluated by means of explicit equations:

$$H_0^{(1)}(x) = \frac{e^{ix}}{ix} \tag{3.3.7}$$

$$H_1^{(1)}(x) = (-i + \frac{1}{x}) H_0^{(1)}(x) \tag{3.3.8}$$

Other ones are generated again by the previous recursion equations. Complex conjugation(which is not a time taking job) will give other Hankel function. Bessel and Neumann functions are used refering only one part of the Hankel functions. [18]

For small values of argument these methods may fail due to enlagening errors during recursions. Miller method(reverse recursion and scaling method) and series solution may be useful but time consuming for this case. Since the position of the bodies is not critical,one can avoid small arguments.

3.4 INDICAL FUNCTIONS

There is no other way,except direct evaluation, for the indical functions. Since this is done only once,no trouble arises. They are used in the expressions together and therefore it is good to evaluate the following expressions, instead :

$$\xi_k = \gamma_k^1 \Delta_k = \left\{ \frac{2k+1}{4\pi} \right\}^{1/2} \tag{3.4.1}$$

$$\eta_k = \frac{\gamma_k^1}{\Delta_k} = \frac{\xi_k}{k(k+1)} \tag{3.4.2}$$

If they are stored arraywise, it is better to evaluate them once,and assign them during compilation. Thus, since the object form of a working program is used and no compilation is required they won't effect duration.

3.5 NECESSARY MAPPINGS FOR INTEGRATION

Assume a collection of bodies is present at a certain region of the frame concerned. The surface integral over the total surface can be seperated into single surface integrals as

$$\oint_{\sum_i S_i} \bar{A} \cdot d\bar{S} = \oint_{S_1} \bar{A} \cdot d\bar{S} + \oint_{S_2} \bar{A} \cdot d\bar{S} \cdots + \oint_{S_n} \bar{A} \cdot d\bar{S} \tag{3.5.1}$$

where n is the total number of bodies.Therefore the problem of finding the scattering matrice reduces to taking the integrals of

$$\oint_{S_i} \bar{A} \cdot d\bar{S} \tag{3.5.2}$$

type and summing them. Every surface canbe described by two independent variables,parametrically. For example:

$$x = f(u,v) \tag{3.5.3}$$

$$y = g(u,v) \tag{3.5.4}$$

$$z = h(u,v) \tag{3.5.5}$$

This is a mapping into (u,v) system.Then with proper substitutions integrals may be converted into a form like :

$$\oint_{S_i} \bar{A}.d\bar{S} = \iint_{A^i_{uv}} \bar{a}(u,v).d\bar{S}(u,v) \tag{3.5.6}$$

where A^i_{uv} is the image of the actual surface S_i. But if the transformation is not one to one ,to describe A^i_{uv} for the computer becomes very difficult(especially for a general program) and therefore the best is to find a nice transformation where A^i_{uv} is just a rectangle. This can be established for a convex body if the body is described with respect to an origin inside the body.

For surface element $d\bar{S}$ one can make the following separation :

$$d\bar{S} = \bar{n}\,dS \tag{3.5.7}$$

where $\bar{n}$ is the unit normal(outward) and dS is the unit surface element. On the other hand if the surface is described as r=r(u,v), where r shows the magnitude of the position vector, one can write:

$$dS = \left|\left(\frac{\partial \bar{r}}{\partial u}du\right)\times\left(\frac{\partial \bar{r}}{\partial v}dv\right)\right| = \left|\frac{\partial \bar{r}}{\partial u}\times\frac{\partial \bar{r}}{\partial v}\right| du\,dv \tag{3.5.8}$$

but $\pm \frac{\partial \bar{r}}{\partial u} \times \frac{\partial \bar{r}}{\partial v} = \bar{n} \left| \frac{\partial \bar{r}}{\partial u} \times \frac{\partial \bar{r}}{\partial v} \right|$ (3.5.9)

$$\bar{n} = \pm \left[\frac{\partial \bar{r}}{\partial u} \times \frac{\partial \bar{r}}{\partial v} \right] \Big/ \left| \frac{\partial \bar{r}}{\partial u} \times \frac{\partial \bar{r}}{\partial v} \right| \qquad (3.5.10)$$

Since these derivatives are tangent to the surface ,their cross product will be in the direction of the normal. Then

$$\pm d\bar{S} = \pm \bar{n}\, dS = \frac{\partial \bar{r}}{\partial u} \times \frac{\partial \bar{r}}{\partial v}\, du\, dv \qquad (3.5.11)$$

is found and (3.5.2) becomes

$$\oint_{S_i} \bar{A} \cdot d\bar{S} = \iint_{A^i_{uv}} \bar{a}(u,v) \cdot \left\{ \frac{\partial \bar{r}}{\partial u} \times \frac{\partial \bar{r}}{\partial v} \right\} du\, dv \qquad (3.5.12)$$

Here it is necessary to take the cross multiplication which will produce outgoing normal direction.[16]

3.6. GENERAL STRUCTURES

The best way to express general convex structures is to describe the bodies with respect to their body frame spherically. Thus the surface of a certain body will be :

$$\bar{r} = r(\theta, \varphi)\, \bar{u}_r \qquad (3.6.1)$$

This ensures smoothness if r is analytical with respect to θ, φ . Then

$$\frac{\partial \bar{r}}{\partial \theta} = \frac{\partial r}{\partial \theta}\, \bar{u}_r + r\, \bar{u}_\theta \qquad (3.6.2)$$

since $\frac{\partial \bar{u}_r}{\partial \theta} = \bar{u}_\theta$ (3.6.3)

$$\frac{\partial \bar{r}}{\partial \varphi} = r \sin\theta\, \bar{u}_\varphi + \frac{\partial \bar{r}}{\partial \varphi}\, \bar{u}_r \qquad (3.6.4)$$

since $$\frac{\partial \bar{r}}{\partial \varphi} = \sin\theta \, \frac{\partial \bar{u}_r}{\partial \varphi} \tag{3.6.5}$$

and thus

$$d\bar{S} = \left(\frac{\partial \bar{r}}{\partial \theta} \times \frac{\partial \bar{r}}{\partial \varphi}\right) d\theta d\varphi = \left(r^2 \sin\theta \, \bar{u}_r + r\frac{\partial r}{\partial \theta} \sin\theta \, \bar{u}_\theta + r\frac{\partial r}{\partial \varphi} \bar{u}_\varphi\right) d\theta d\varphi \tag{3.6.7}$$

or $$\oint_{S_L} \bar{A} \cdot d\bar{S} = \int_0^{2\pi}\int_0^{\pi} \bar{a}(\theta,\varphi) \cdot \left\{ r^2 \sin\theta \, \bar{u}_r + r\frac{\partial r}{\partial \theta} \sin\theta \, \bar{u}_\theta - r\frac{\partial r}{\partial \varphi} \bar{u}_\varphi \right\} d\theta d\varphi \tag{3.6.8}$$

is obtained.

But unfortunately $\bar{a}(\theta,\varphi)$ can best be evaluated with respect to prime coordinate frame(original one).Therefore one has to describe vectors in another system. A vectorial coordinate change requires two things, change of the point descriptions, base vector transformations. Assume the coordinates of the body frame is given as (x_c, y_c, z_c), then

$$x_p = r_b \sin\theta_b \cos\varphi_b + x_c \tag{3.6.9}$$

$$y_p = r_b \sin\theta_b \sin\varphi_b + y_c \tag{3.6.10}$$

$$z_p = r_b \cos\theta_b + z_c \tag{3.6.11}$$

where the subscripts b,p show body and prime frames respectively. On the other hand in terms of prime spherical coordinates are

$$r_p = (x_p^2 + y_p^2 + z_p^2)^{1/2} \tag{3.6.12}$$

$$\theta_p = \arccos(z_p / r_p) \tag{3.6.13}$$

$$\varphi_p = \operatorname{arctg}(y_p / r_p) \tag{3.6.14}$$

Thus prime coordinates can easily be found. To find a direct relation between vectorial components of the frames is a bit

difficult . But one can use a rectangular system as an intermediate step. For rectangular systems if axes are kept parallel to each other, chosing a particular frame is immaterial for vectorial transformations.

$$\begin{vmatrix} \bar{a}_x \\ \bar{a}_y \\ \bar{a}_z \end{vmatrix} = \begin{vmatrix} \sin\theta_b \cos\varphi_b & \cos\theta_b \cos\varphi_b & -\sin\varphi_b \\ \sin\theta_b \sin\varphi_b & \cos\theta_b \sin\varphi_b & \cos\varphi_b \\ \cos\theta_b & -\sin\theta_b & 0 \end{vmatrix} \begin{vmatrix} \bar{a}_r \\ \bar{a}_\theta \\ \bar{a}_\varphi \end{vmatrix}_b \tag{3.6.15}$$

gives body-frame cartesian components.For later reference we will use

$$S(\alpha,\beta) = \begin{vmatrix} \sin\alpha \cos\beta & \cos\alpha \cos\beta & -\sin\beta \\ \sin\alpha \sin\beta & \cos\alpha \sin\beta & \cos\beta \\ \cos\alpha & -\sin\alpha & 0 \end{vmatrix} \tag{3.6.16}$$

see that (3.6.17)

thus the transformation is orthogonal and inverse of S is equal to its transpose. Using this, a second transformation can be written immidiately

$$\begin{vmatrix} \bar{a}_r \\ \bar{a}_\theta \\ \bar{a}_\varphi \end{vmatrix}_p = S^T(\theta_p, \varphi_p) \begin{vmatrix} \bar{a}_x \\ \bar{a}_y \\ \bar{a}_z \end{vmatrix} \tag{3.6.18}$$

Therefore

3.7 TOROIDAL STRUCTURES AND DEGENERATE SPHERICAL COORDINATES

With the name 'toroidal body', a revolutionary volume is understood,which is made revolving a plane figure around the z axis,which is chosen coincident with the direction of incidence. Although a toroidal ensemble cannot be assumed to be a general multibody structure,it presents very nice properties, which reduce the calculations considerably. For such a structure , the equations given in chapter 2 become uncoupled for m,m' . Thus a huge system of equations can be partioned into smaller and simpler systems. If the incident field is expandable for a finite variation of m's,as in case of a c.p.plane wave in z direction , the number of the partioned systems are also finite.Regardless these facts, the previous method described in 3.6 is inadequate for solving toroidal scattering problems(for example from a ring) since with that transformation mapping is not one-to-one, for toroids.Therefore more or less a method to handle toroidal configurations will be useful.

Toroidal ensembles can best be treated by degenerate spherical coordinates. For this system surface integrals become simple integrals . Assume the characteristic curve of the toroid (cross section at only one side of a plane including z axis) can be described with a polar system(if the curve is convex)by a one to one mapping from β to ρ . Where β is the angular variable , ρ denotes the radial variable. Then the surface integrals become integrals of the form

$$\int_{\beta_{min}}^{\beta_{max}} \left(F_r(\beta)\, dS_r(\beta) + F_\theta(\beta)\, dS_\theta(\beta) \right) d\beta = \int_{\beta_{min}}^{\beta_{max}} f(\beta)\, d\beta \qquad (3.7.1)$$

where the limits are shown in Fig.(3.7.1)

The necessity for a one-to-one correspondence is not a handicape, since any non-convex shape may be sectioned and may be described with respect to several suitable origins and limits. Fundamental equations of the degenerate spherical coordinates are given in terms of cartesian variables as

$$X_1 = (a + \rho \cos\beta) \cos\varphi \qquad (3.7.2)$$

$$X_2 = (a + \rho \cos\beta) \sin\varphi \qquad (3.7.3)$$

$$X_3 = (b + \rho \sin\beta) \qquad (3.7.4)$$

However this is not enough. Some other properties have to be known. The most importants are

$$h_\rho = \left\{ \sum_{i=1}^{3} \left(\frac{\partial x_i}{\partial \rho} \right)^2 \right\}^{1/2} = 1 \qquad (3.7.5)$$

$$h_\beta = \left\{ \sum_{i=1}^{3} \left(\frac{\partial x_i}{\partial \beta} \right)^2 \right\}^{1/2} = \rho \qquad (3.7.6)$$

$$h_\varphi = \left\{ \sum_{i=1}^{3} \left(\frac{\partial x_i}{\partial \varphi} \right)^2 \right\}^{1/2} = a + \rho \cos\beta \qquad (3.7.7)$$

Remembering [15]

$$d\bar{r} = h_\rho\, d\rho\, \bar{u}_\rho + h_\beta\, d\beta\, \bar{u}_\beta + h_\varphi\, d\varphi\, \bar{u}_\varphi \qquad (3.7.8)$$

$$\frac{\partial \bar{r}}{\partial \beta} = \frac{\partial \rho}{\partial \beta}\, \bar{u}_\rho + \rho\, \bar{u}_\beta \qquad (3.7.9)$$

$$\frac{\partial \bar{r}}{\partial \varphi} = \frac{\partial \rho}{\partial \varphi}\, \bar{u}_\rho + (a + \rho \cos\beta)\, \bar{u}_\varphi \qquad (3.7.10)$$

Now taking the cross product

$$\bar{n}\,dS = \frac{\partial \bar{F}}{\partial \beta} \times \frac{\partial \bar{F}}{\partial \varphi}\, d\beta\, d\varphi \tag{3.7.11}$$

$$\bar{n}\,dS = \left[(a+\rho\cos\beta)\rho\,\bar{u}_\rho - (a+\rho\cos\beta)\frac{\partial\rho}{\partial\beta}\,\bar{u}_\beta - \rho\frac{\partial\rho}{\partial\varphi}\,U_\varphi\right] d\beta\, d\varphi \tag{3.7.12}$$

is obtained. However for toroidal bodies further reductions are possible because

$$\bar{U}_\varphi \cdot \bar{F} = 0 \tag{3.7.13}$$

$$\frac{\partial\rho}{\partial\varphi} = 0 \tag{3.7.14}$$

in short a modified surface element is produced as

$$\bar{n}\,dS' = 2\pi\,(a+\rho\cos\beta)\left(\rho\,\bar{u}_\rho - \frac{\partial\rho}{\partial\beta}\,\bar{u}_\beta\right) d\beta \tag{3.7.15}$$

Since the vector field to be integrated is known in terms of its spherical components, transformation matrice has to be known for vectorial conversions. This can be derived from F.3.7.3. and the following result is achieved :

$$\begin{vmatrix} \bar{U}_r \\ \bar{u}_\theta \\ \bar{u}_\varphi \end{vmatrix} = \begin{vmatrix} \sin(\beta+\theta) & \cos(\beta+\theta) & 0 \\ \cos(\beta+\theta) & -\sin(\beta+\theta) & 0 \\ 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} \bar{u}_\rho \\ \bar{u}_\beta \\ \bar{u}_\varphi \end{vmatrix} \tag{3.7.16}$$

Now the surface element vector can be transfered into a spherical system as

$$dS_r = 2\pi\,(a+\rho\cos\beta)\left(\rho\sin(\beta+\theta) - \frac{\partial\rho}{\partial\beta}\cos(\beta+\theta)\right) d\beta \tag{3.7.17}$$

$$dS_\theta = 2\pi\,(a+\rho\cos\beta)\left(\rho\cos(\beta+\theta) + \frac{\partial\rho}{\partial\beta}\sin(\beta+\theta)\right) d\beta \tag{3.7.18}$$

in these equations following identity is used :

$$\theta = \operatorname{arctg} \frac{a + \rho \cos\beta}{b + \rho \sin\beta} \tag{3.7.19}$$

Similarly

$$r = \left\{ (a + \rho \cos\beta)^2 + (b + \rho \sin\beta)^2 \right\}^{1/2} \tag{3.7.20}$$

By means of these transformations, integrals at the end become simple integrals. They can be evaluated by Gauss-Legendre type of quadrature. Related formulas for this time are simply

$$\int_{\beta_{min}}^{\beta_{max}} f(\beta)\, d\beta = \sum_i f(\beta_i)\, W_i' \tag{3.7.21}$$

where

$$\beta_i = \frac{\beta_{max} - \beta_{min}}{2} X_{qi} + \frac{\beta_{max} + \beta_{min}}{2} \tag{3.7.22}$$

$$W_i' = \frac{\beta_{max} - \beta_{min}}{2} W_{qi} \tag{3.7.23}$$

in these equations x and w stands for quadrature arguments and weights respectively.

3.8. SOLUTION OF THE SYSTEM OF EQUATIONS

After evaluation of the scattering matrice and the boundary matrice preceisely,one has to find the solution to the system given in chapter 2. It is better to find X from the equation

$$(S-B)\ X = U \tag{3.8.1}$$

than obtain Y from the boundary relations. Since the scattering integrals are complex a careful investigation is necessary . Solution of this matrice system is vital because this is the part which enforce the computer core capacity. Therefore a preceise, fast, less-core requiring algorithm has to be used. If the system is not an uncoupled one for m,m`, matrice sizes increase tre-

mendously. This is still a problem to be solved. If the equations are uncoupled, solutions for a sequence of equations of simpler of simpler form is necessary and can be solved with less trouble. If we can limit expansion of the incident wave over m to a finite number of values, problem reduces more. This is the case for a lower illimunation of a toroidal system by a c.p. plane wave. Since we always want to increase the number of terms in the expansions to provide a better approximation, whatsoever the equation system is, it is a large one.

Several methods are employed during investigations. Some of these methods and their drawbacks are outlined below :

a) Seperation of Real and Imaginary Parts:

Given a system of complex linear equations of the form

$$A\,X = B \qquad (3.8.2)$$

It can be seperated into real and imaginary parts in the form

$$\left|\begin{array}{c|c} A_r & -A_i \\ \hline A_i & A_r \end{array}\right| \left|\begin{array}{c} X_r \\ \hline X_i \end{array}\right| = \left|\begin{array}{c} B_r \\ \hline B_i \end{array}\right| \qquad (3.8.3)$$

Thus a real system of equations is obtained. Although this seems as if an easy method, it is not efficient because it enlargenes the system unnecessarily. Because the system size is now 2n if there were n equations at the beginning. Core capacity requirement makes the usage of single precission necessary. Since the

largeness of the system brings enough error, single precision replacement makes the situation worse. For this reason, this is not a good method for solving such a system.

b) Crout Iterative Refinement Technique:

This is a rather simple technique. If A is a nonsingular complex matrice, it can be factorized in the form [19]

$$A=LU \tag{3.8.4}$$

where L is a lower triangle matrix, U is upper triangular. The elements of the matrices L and U of the crout factorization may be determined in the order r-th column of L, r-th row of U where r varies from 1 to n, by means of the equations :

$$\sum_{k=1}^{r-1} l_{ik} u_{kr} + l_{ir} = a_{ir} \qquad (i=r,\dots n) \tag{3.8.5}$$

$$\sum_{k=1}^{r-1} l_{rk} u_{ki} + l_{rr} u_{ri} = a_{ri} \quad (i=r+1,\dots n) \tag{3.8.6}$$

After partioning, to obtain the solution two steps are necessary. First the system

$$Ly=b \tag{3.8.7}$$

is solved, then the subsidary system

$$Ux=y \tag{3.8.8}$$

is solved. Although this seems to be a rather simple algorithm, stability has to be assured by means of some pivoting. Pivoting

performed can best be described in terms of the ususal algorithmic interpretation including the storage arrangement. There are n major steps and at the beginning of the rth step, the stored array is illustrated when n=4,r=3

$$\begin{bmatrix} l_{11} & u_{12} & u_{13} & u_{14} \\ l_{21} & l_{22} & u_{23} & u_{24} \\ l_{31} & l_{32} & a_{33} & a_{34} \\ l_{41} & l_{42} & a_{43} & a_{44} \end{bmatrix} \qquad (3.8.9)$$
$$\begin{bmatrix} int_1 & int_2 & int_3 & int_4 \end{bmatrix}$$

The rth step is as follows

i)Compute $l_{ir} = a_{ir} - \sum_{k=1}^{r-1} l_{ik} u_{kr}$ and over write on a_{ir} $(i=r,r+1,\ldots,n)$

ii)If int_r is the smallest integer for which $|l_{int_r,r}| = \max_{i \geq r} |l_{ir}|$, then store the integer int_r and interchange the whole rows of r and int_r in the current array.

iii)Compute $u_{ri}=(a_{ri} - \sum_{k=1}^{r-1} l_{rk}u_{ki})/l_{rr}$ and over write on a_{ri} $(i=r+1,\ldots,n)$

After such n steps A is replaced by L and U and the n integers $int_i (i=1,\ldots,n)$ give the details of the interchanges. If P_r denotes the permutation matrix,premultiplication by which gives interchange of rows r and int_r , then

$$P_n P_{n-1} \cdots P_1 A = \tilde{A} = LU \qquad (3.8.10)$$

in other words we have the orthodox Crout decomposition of A defined by Eq(3.8.10). The triangles L and U and the array of

integers int, enable us to solve the equations A x = b corresponding any right side b. For we have

$$P_n P_{n-1} \ldots P_1 A x = b \qquad (3.8.11)$$

or

$$L U x = P_n \ldots P_1 b \qquad (3.8.12)$$

We have only to subject b to the appropriate interchanges to give say $\tilde{b}$ and solve

$$L y = \tilde{b} \qquad (3.8.13)$$

$$U x = y \qquad (3.8.14)$$

In general the computer solution of A x = b can be refined, using the crout factorization, by means of the iterative procedure

$$x^{(o)} = 0 \qquad (3.8.15)$$

$$r^{(s)} = b - A x^{(s)} \qquad (3.8.16)$$

$$L U d^{(s)} = P_n \ldots P_1 r^{(s)} \qquad (3.8.17)$$

$$x^{(s+1)} = x^{(s)} + d^{(s)} \qquad (3.8.18)$$

Crout method is a preceise method. But both factorizing and iterating schemes reserve large amount of core for their instructions. Time requirement is also important. Since undestroyed form of A has to be preserved, external devices are necessary to solve storage requirements. However Crout scheme requires less operations than Gauss elimination (without iterations.).

c) Gauss Elimination Method :

This is a well known method and needs no description. It applies equally well for Complex Systems too. However if some pivoting is not done, results are strange due to numerical in - stabilities.

In scattering algorithms two methods of pivoting is used, namely column-row pivoting and only row pivoting. Row pivoting gives satisfactory results if the system is not too large and some strong ill conditionedness does not occur. Other method is time consuming, but more preceise. In pivoting square of the absolute value is used as a measure for significance.

Gauss elimination scheme allows iterative refinement too. Assume ϵ denotes some deviation (in vector form) from the actual result due to operational truncations and numerical instabilities. If the result found is substituted in the equation, it will yield a difference Δb from the actual constant vector. But the following two equations show that ϵ and Δb are related to each other with the same coefficient matrice. Therefore an iterative refinement technique is applicable until a certain tolerance is achieved.

$$A(x+\epsilon) = b + \Delta b \qquad (3.8.19)$$

$$A\epsilon = \Delta b \qquad (3.8.20)$$

If iterative refinement is used, Crout scheme becomes more economical from every respect. [9],[10],[12],[13],[14]

d)Gauss Seidel Iteration :

If the scattering equations are re-arranged such that x vector is now in the form :

$$X' = \begin{vmatrix} \alpha MI^1 \\ \hline \alpha EI^1 \\ \hline \alpha MI^2 \\ \hline \alpha EI^2 \end{vmatrix} \qquad (3.8.21)$$

the iterative technique becomes applicable without any modi - fication. This is due to great possibility of having larger terms on the diagonal. The simplest of the methods which can be applied is the Gauss-Seidel iterations and this can be applied with suitable error comparison-i.e. using the square of the absolute value-to complex systems also. But convergence cannot be guaranteed for every case. Otherwise it is very suitable for large system solving; it requires less time and has a good precission. [17]

e)Larger Systems :

It is necessary to develope non-iterative methods to solve large systems with less core requirement. The most general problem still requires this. Although not tried, one possibility is to partionize the coefficient matrice and invert.

Steps are as follows : Assume A is the given matrice and it is partioned A_{11} being a square matrice.

$$A = \left|\begin{array}{c|c} A_{11} & A_{12} \\ \hline A_{21} & A_{22} \end{array}\right| \qquad (3.8.22)$$

The inverse of A can be written as :

$$A^{-1} = \left|\begin{array}{c|c} A_{11}^{-1} + X\theta^{-1}Y & -X\theta^{-1} \\ \hline -\theta^{-1}Y & \theta^{-1} \end{array}\right| \qquad (3.8.23)$$

Where

$$X = A_{11}^{-1} A_{12} \qquad (3.8.24)$$

$$Y = A_{21} A_{11}^{-1} \qquad (3.8.25)$$

$$\theta = A_{22} - Y A_{12} \qquad (3.8.26)$$

$$A_{22} - A_{21} X = A_{22} - A_{21} A_{11}^{-1} A_{12} \qquad (3.8.27)$$

This method seems to be applicable only with some external storage devices for the computer available at METU Computation Center.

Consider Hankel functions for asymptotical values :

$$h_p^{(1)}(r) = (-i)^{p+1} \frac{e^{ir}}{r} \tag{3.9.7}$$

neglecting the terms varying with $1/R^2$ compared with the terms varying with $1/R$, using the relation for toroidal lower illumination

$$(E^{sc})_\theta = 2\frac{e^{ir}}{r}\sum_p -\frac{\gamma_p^1}{\Delta_p}(-i)^{p+1}\cos\varphi\left(i\alpha Es_p^+ \frac{dP_p^1}{d\theta} + \alpha Ms_p^+ \frac{P_p^1}{\sin\theta}\right) \tag{3.9.8}$$

$$(E^{sc})_\varphi = 2\frac{e^{ir}}{r}\sum_p \frac{\gamma_p^1}{\Delta_p}(-i)^{p+1}\left(i\alpha Es_p^+ \frac{P_p^1}{\sin\theta} + \alpha Ms_p^+ \frac{dP_p^1}{d\theta}\right)\sin\varphi \tag{3.9.9}$$

and the definition for polar re-radiation diagram

$$\sigma(\theta,\varphi) = \frac{4\pi r^2 \quad \bar{u}(\theta,\varphi) \cdot \bar{S}^{sc*}(\theta,\varphi)}{\bar{u}_z \cdot \bar{S}_{inav}} \tag{3.9.10}$$

a simple expression of the form

$$\sigma\{\theta,\varphi\} = \frac{4\pi}{k^2}\left[\cos^2\varphi\left|\sum_p \bar{F}_p(\theta)\right|^2 + \sin^2\varphi\left|\sum_p \bar{G}_p(\theta)\right|^2\right] \tag{3.9.11}$$

is obtained. Where

$$F_p(\theta) = \frac{-\gamma_p(-i)^{p+1}}{\Delta_p}\left(i\alpha Es_p^+ \frac{dP_p^1}{d\theta} + \alpha Ms^+ \frac{P_p^1}{\sin\theta}\right) \tag{3.9.12}$$

$$G_p(\theta) = \frac{\gamma_p(-i)^{p+1}}{\Delta_p}\left(i\alpha Es_p^+ \frac{P_p^1}{\sin\theta} + \alpha Ms^+ \frac{dP_p^1}{d\theta}\right) \tag{3.9.13}$$

Specially for $\varphi=0$ and $\varphi=\frac{\pi}{2}$ we can write

$$\frac{\sigma(\theta,0)}{\lambda^2} = \frac{1}{4\pi^2}\left|\sum_p \frac{\sqrt{2p+1}}{p^2+p}(-i)^{p+1}\left(i\alpha Es_p^+ \frac{dP_p^1}{d\theta} + \alpha Ms_p^+ \frac{P_p^1}{\sin\theta}\right)\right|^2 \tag{3.9.14}$$

$$\frac{\sigma(\theta,\frac{\pi}{2})}{\lambda^2} = \frac{1}{4\pi^2}\left|\sum_p \frac{\sqrt{2p+1}}{p^2+p}(-i)^{p+1}\left(i\alpha Ms_p^+ \frac{dP_p^1}{d\theta} + i\alpha Es_p^+ \frac{P_p^1}{\sin\theta}\right)\right| \tag{3.9.15}$$

these expressions for $\theta=\pi$ give the back scattering cross-section.

3.10. PROGRAMMING

Several computer programs are prepared based on the methods described. For runs of these programs IBM-360/40 system of METU is used. This computer, although is a relatively fast computer , it has not enough core capacity to handle a general scattering problem without using some tricks and external devices; but this makes a considerable amount of increase in duration. But for lower illumination of symmetric structures a complate subprogram is available to evaluate upto 8 multipole coefficients without any extra device or trick. After 8 coefficients multiphase programming and usage of disk become necessary.

The main subroutine which solves toroidal problem does not depend on the selection of the origin, with proper branchings origin inclusive and origin exclusive problems can be treated. Program works for single and multibody, different quadratures, without re-running infinitely many configurations. It can be used for a frequency sweep over some configurations. If required with a communication code S matrice, coefficients, polar diagram data, polar diagram plots are given one by one or all together.

Equation solving and Bessel, Legendre generating sections are seperate moduls which can be replaced with others. For example instead of forward recursion, Miller method or series method can be used, Gauss Seidel may be used instead of Crout, etc.

Programming language is in general Fortran, IV 360 DOS levelwhich is sufficient for a problem of scientific nature. There are many versions of the programs which are used for different purposes. They are run in their object deck forms to save compilation times(since the minimum compilation time recorded is 5 min.)

Other languages are also investigated at the beginning. Algol which provides a recursive flexibility cannot be used since the available system is not equipped with Algol compiler. Pl-I seems compatable with Fortran but available compiler is for a subset of it,therefore no attempt is made to write in Pl-I. Assembler language is not proper because of clerical tasks which are necessary, but when Fortran fails it can be mixed with Fortran with proper links. This is done at the primary steps where duration of the operations had to be measured. This can be done by the Assembler Language but not by Fortran of 360/40. Getime macro can be used for this [11] purpose, if this is done with Binary it can be transfered into Fortran being an integer variable. Thus the time islearned by Fortran. Several other tricks can be done, for example standard Fortran function REAL(.) can be established for complex double precission variables also(which is not available) with minimum operations.

With this form of the programs no attempt should be made to use with an arbitary system, because some unconventional features are also used either for saving time or core. Suitable changes can be made easily however.

4. EXPERIMENTAL INVESTIGATIONS

Some back-scattering measurements are made from a few configurations for which numerical values were present. Due to set up problems no experimental re-radiation diagram is obtained. Toroidal configurations were impossible to manifacture with the tools present.

The procedure to measure the back-scattering cross-section from a configuration for a certain frequency takes about four hours, since repetitions are necessary. Frequencies of the incident waves are chosen to be in the range of 8-11 GHZ. This is to minimize the Rayleigh distanceof the radiating antennas. Otherwise in-room measurements are impossible and the ensemble has to be located very far from the antenna. The bodies were manifactured from brass. Tolerance of manifacturing is less then 0.05 mm. Only certain shapes can be made easily, such as spheres, prolate spheroids, eight like figures,etc. But their ensembles may be chosen in various forms. Some of the configurations reduce the total back-scattering cross sections below the proper measurement limit. This is due to tendency to cancel each other of the re-radiated fields of different objects. For example if a body is put in the shadow region of another one,sometimes the backscattering cross-section of the ensemble is less then the back-scattering cross sections of the isolated bodies.

Spherical TEM waves which are produced by a horn antenna may be assumed to be plane waves after Rayleigh distance and this may be used as a plane wave source. But if the region of interest is too long,decay in the magnitude disturbs the accuracy of the results. Phase variations may be reduced placing the bodies far from the horn antenna. Another possibility is to place the ensemble close to the aperture near the symmetry axis of the antenna. If the uniform illumination of the apeture can be obtained, plane wave assumption for small objects is valid. Both of these methods are used to check the accuracy of the results.

It is necessary to isolate the set-up with dissipative sheets to prevent unwanted scatterings . Thus an artificial free space is established. First the antenna is matched to empty range. Then a region where fields are less changing with distance is found. This can be found if the antenna is sligthly mismatched. The reflection level from the ensemble is recorded. Then the same measurement is repeated with spheres of several diameters. The difference betwwen the sphere and the ensemble scatterings are measured by means of a calibrated attenuator, obtaining the same amount of deflection of VSWRM in every measurement.

Backscattering cross-sections of the spheres are calculated for every frequency and they are compiled as tables. From these tables actual back-scattering cross-section of the ensemble

can be calculated.Some data which is obtained in this fashion is shown in the tables I-III.

Evidently for elongated configurations, the results will be deviated from the calculated ones,because simulation of a plane wave will fail for such ensembles.This is observed for the objects seperated too much from each other. Other important factor is the position of the standard spheres. It is difficult to replace the ensemble by a sphere, esspecially keeping the coincidence of the symmetry axis with the axis of the antenna. Therefore for different spheres, different results are obtained.

The seperation of the bodies has to be established properly. Otherwise at high frequencies some troubles may occur due to increase in the error of the optical length of the seperation. This is provided placing a dielectric seperator of fixed tickness in between the bodies. This dielectric has the properties very close to the properties of the air.

The results which are few in quantity are satisfactory. Some preceise measuring techniques seem to be necessary for further research in this field. Some deviations are observed for elongated bodies as expected. A more uniform source is necessary to investigate the full range of the numerical results.

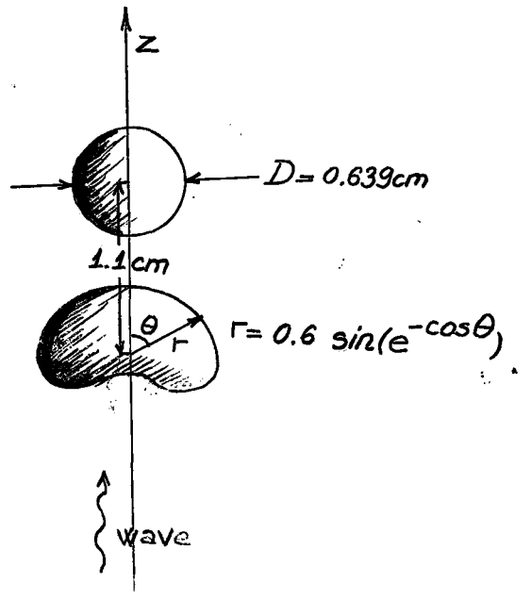

Fig 4.1. - Configuration (A)

Table I - Configuration (A), f= 10.0 GHZ, R=18 cm.

Sphere ※	Diam. (cm)	Optical Radius	Sphere σ_{db}	Rel.to Sphere (db)	σ_{db}	σ_b/π
1	.639	.669	-3.1	11.5	8.4	2.2
2	.793	.830	1.46	7.75	8.2	2.1
3	.940	.982	3.99	4.0	8.	2.0
4	1.109	1.160	5.06	2.85	7.9	1.95
5	1.200	1.256	4.94	2.9	7.84	1.92
6	1.268	1.325	4.59	3.25	7.8	1.9

CALCULATED : $\frac{\sigma_b}{\pi}$ =1.94 , σ_b =7.9 db.

Table II - Configuration (A), f=9.6 GHZ, R = 18 cm.

Sphere *	Diam. (cm.)	Optical Radius	Sphere σ_{db}	Rel. to Sphere (db)	σ_{db}	σ_b/π
1	.639	.642	-3.709	11.9	8.2	2.2
2	.793	.796	1.025	6.5	7.525	1.8
3	.940	.945	3.832	3.5	7.532	1.8
4	1.109	1.112	5.307	2.0	7.307	1.7
5	1.200	1.205	5.414	1.9	7.314	1.7
6	1.268	1.272	5.244	2.0	7.244	1.68

CALCULATED : σ_b/π = 1.73 , σ_{db} =7.4 db

Table III - Configuration (A), f=9.2 GHZ, R=18 cm.

Sphere *	Diam. (cm.)	Optical Radius	Sphere σ_{db}	Rel. to Sphere	σ_{db}	σ_b/π
1	.639	.615	-4.36	9.95	5.59	1.15
2	.793	.765	0.569	5.20	5.77	1.17
3	.940	.906	3.633	2.3	5.933	1.25
4	1.109	1.067	5.440	0.6	6.04	1.27
5	1.200	1.158	5.752	0.3	6.052	1.28
6	1.268	1.22	5.747	0.38	6.12	1.3

CALCULATED : σ_b/π = 1.505 , σ_{db} =6.6 db

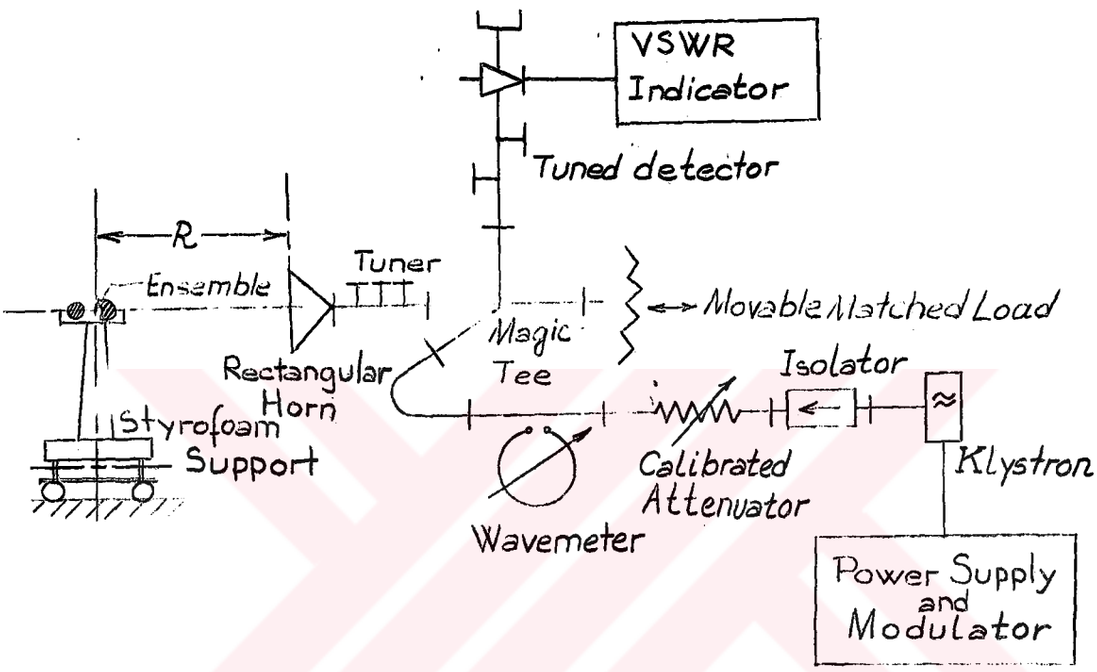

Fig.4.2. Experimental Set-Up for σ_b Measurements

5.GENERAL CONCLUSIONS

Numerical analysis of the problem of scattering of electromagnetic waves from arbitary perfectly conducting ensembles show the fact the method is limited for such ensembles due to computational difficulties. A large storage is necessary to utilize the algorithms proposedfor this problem. Usage of external storage devices may solve this problem however. But this is disadvatageous since the duration increase with such devices, esspecially if frequent references are made. This means the cost of the research increase. However even with the systems of larger core capacity to increase the number of multipole coefficients may be impossible after a certain limit.

Toroidal ensembles may be evaluated with less trouble if the direction of the incident field is coincident with the axis of revolution. Programs are made for this case requiring approximately the complate storage of a 64 K.byte computer for evaluation of 8 multipole coefficients . Execution of the program for a complate solution leading to re-radiation diagrams and plots of them takes 3 minutes for a certain configuration,with the system IBM 360 40-F. However for an arbitary excitation of such an ensemble disk usage and multiphase programming may be necessary. For this case there is a chance of using the external

devices more efficiently due to the uncoupled property of the scattering equations. Thus the number of references to an external file reduces. Incidence with a different angle can be investigated expanding the wave with respect to the main coordinate system.This also requires external storage butwith uncoupled equations.

Experiments show the difficulty to provide idealized measurement conditions. Since the dimensions increase with the second body due to the seperation between the bodies, one has to find a more uniform wave source first, or produce smaller body dimensions. Then there is a limit for observable backscattering cross-sections with the used equipment and smaller bodies produce smaller back-scattering cross-sections. On the other hand to locate the bodies with the required precision is very difficult.

Results are satisfactory and show the validity of the method followed. The modification of the theory seems to be valid also. The most essential property of the formulation is the inclusion of the multiple scatterings without extra effort, since the problem is considered as a radiation due to induced surface sources. If some part of the ensemble touches one of the boundary seperation spheres, along a continuous surface segment,the formulation fails since the expansions are no more valid with their previous forms. However, this is not a drawback,it is possible to chose a second origin which allows a valid expansion.

APPENDIX 1. IE AND IM TYPE INTEGRANDS IN SPHERICAL COORDINATES

Throughout this section, distances will be normalized by k, thus every distance is optical. This helps simplifications and generalizations. Every free k in the expressions is associatable to a distance unit of the surface elements or the expressions.

a) IE type

If the preceeding argument is applied the integrand becomes

$$\overline{IE}^{\alpha\beta}_{pqmm'}(\bar{r}) = \overline{M}^{\alpha *}_{pm}(\bar{r}) \times \overline{M}^{\beta}_{qm}(\bar{r}) \tag{A.1.1}$$

where $\overline{M}^{\alpha}_{pm}$ can be written in normalized form as

$$\overline{M}^{\alpha}_{pm}(\bar{r}) = Z^{\alpha}_{p}(r)\, \overline{X}^{\alpha}_{pm}(\theta,\varphi) \tag{A.1.2}$$

where Z and X are defined in chapter 2 . Evidently

$$\left\{\overline{M}^{\alpha}_{pm}\right\}_r = 0 \tag{A.1.3}$$

$$\left\{\overline{M}^{\alpha}_{pm}\right\}_\theta = Z^{\alpha}_{p}(r)\left\{\overline{X}_{pm}(\theta,\varphi)\right\}_\theta \tag{A.1.4}$$

$$\left\{\overline{M}^{\alpha}_{pm}\right\}_\varphi = Z^{\alpha}_{p}(r)\left\{\overline{X}_{pm}(\theta,\varphi)\right\}_\varphi \tag{A.1.5}$$

$$\left\{\overline{X}_{pm}\right\}_r = 0 \tag{A.1.6}$$

$$\left\{\overline{X}_{pm}\right\}_\theta = -m\,\frac{\gamma_{pm}}{\Delta_{pm}}\,\frac{P^{m}_{p}(\cos\theta)}{\sin\theta}\,e^{im\varphi} \tag{A.1.7}$$

$$\left\{\overline{X}_{pm}\right\}_\varphi = -i\,\frac{\gamma_{pm}}{\Delta_{pm}}\,e^{im\varphi}\,\frac{d}{d\theta}P^{m}_{p}(\cos\theta) \tag{A.1.8}$$

where P^{m}_{p} is the well known Associate Legendre Function, and

$$\gamma_p^m = \sqrt{\frac{(2p+1)}{4\pi}\frac{(p-m)!}{(p+m)!}} \tag{A.1.9}$$

$$\Delta_p = ((p+1)p)^{1/2} \tag{A.1.10}$$

are indical functions. If the cross product is taken term by term

$$\left[\overline{IE}^{\alpha\beta}_{pqmm'}(\bar{r})\right]_r = m\frac{\gamma_p^m\,\gamma_q^{m'}}{\Delta_p\,\Delta_q\sin\theta}\left\{m P_p^m(\cos\theta)\frac{d}{d\theta}P_q^{m'}(\cos\theta) + P_q^{m'}(\cos\theta)\frac{d}{d\theta}P_p^m(\cos\theta)\right\}\left\{i Z_p^{\alpha *}(\bar{r})\,Z_q^{\beta}(r)\right\}e^{i(m-m')\varphi} \tag{A.1.11}$$

$$\left[\overline{IE}^{\alpha\beta}_{pqmm'}(\bar{r})\right]_\theta = 0 \tag{A.1.12}$$

$$\left[\overline{IE}^{\alpha\beta}_{pqmm'}(\bar{r})\right]_\varphi = 0 \tag{A.1.13}$$

b) IM type

Again using the normalization and using

$$\bar{N}^{\alpha}_{pm}(\bar{r}) = \bar{\nabla}\cdot\bar{M}_{pm}(\bar{r}) \tag{A.1.14}$$

$$\left[\bar{N}^{\alpha}_{pm}\right]_r = i\Delta_p\frac{Z_p^{\alpha}(r)}{r}\gamma_p^m\,P_p^m(\cos\theta)\,e^{im\varphi} \tag{A.1.15}$$

$$\left[\bar{N}^{\alpha}_{pm}\right]_\theta = -\frac{1}{r}\left\{\bar{X}_{pm}(\theta,\varphi)\right\}_\theta\frac{d}{dr}\left[r Z_p^{\alpha}(r)\right] \tag{A.1.16}$$

$$\left[\bar{N}^{\alpha}_{pm}\right]_\varphi = \frac{1}{r}\left\{\bar{X}_{pm\varphi}(\theta,\varphi)\right\}_\varphi\frac{d}{dr}\left[r Z_p^{\alpha}(r)\right] \tag{A.1.17}$$

Then

$$\left[\overline{IM}^{\alpha\beta}_{pqmm'}(\bar{r})\right]_r = -\frac{\gamma_p^m\,\gamma_q^{m'}}{r\,\Delta_p\,\Delta_q}\left\{\frac{P_p^m(\cos\theta)\,P_q^{m'}(\cos\theta)}{\sin^2\theta} + \frac{dP_p^m(\cos\theta)}{d\theta}\frac{dP_q^{m'}(\cos\theta)}{d\theta}\right\}\left\{i Z_p^{\alpha *}(r)\frac{d}{dr}Z_q^{\beta}(r)\right\}e^{i(m-m')\varphi} \tag{A.1.18}$$

$$\left[\overline{IM}^{\alpha\beta}_{pqmm'}(\bar{r})\right]_\theta = \frac{\Delta_q\,\gamma_p^m\,\gamma_q^{m'}}{\Delta_p\,r}\,P_q^{m}(\cos\theta)\,\frac{d\left(P_p^{m'}(\cos\theta)\right)}{d\theta}\left\{i Z_p^{\alpha *}(r)\,Z_q^{\beta}(r)\right\}e^{i(m-m')\varphi} \tag{A.1.19}$$

$$\left[\overline{IM}^{\alpha\beta}_{pqmm'}(\bar{r})\right]_\varphi = m\frac{\gamma_p^m\,\gamma_q^{m'}\,\Delta_q}{\Delta_p\,r\sin\theta}\,Z_p^{\alpha *}(r)\,Z_q^{\beta}(r)\,P_p^m(\cos\theta)\,P_q^{m'}(\cos\theta)\,e^{i(m-m')\varphi} \tag{A.1.20}$$

APPENDIX II. USAGE OF THE PROGRAM GIVEN IN THE APPENDIX

The given program package is one of the versions(CUBEX) which is used to investigate some standard configurations, for a range of frequencies.Data cards are arranged as follows :

The first card contains a description of the problem with a field of (10I1,10X,2D30.15) where the parameters are:

1st integer	Number of surfaces.
2nd integer	0 if the origin is included.
3rd integer	1 if S matrix is required.
4th integer	1 if the program will continue on the pr.
5th integer	1 if the multipole coefficients are req.
6th integer	1 if the program will continue on the pr.
7th integer	1 if $\sigma(\theta,\varphi)$ is required.
8th integer	1 if the plot of $\sigma(\theta,0)$ is required.
9th integer	1 if the plot of $\sigma(\theta\,\pi/2)$ is required.
10th integer	Number of frequency steps.

Other parameters are the starting frequency and the increment in terms of GHZ. After the first card blocks for different bodies follow.

Cards for ith body is as follows :

1st card(32A1) Literal description of the body.

2nd card(2D35.18)Body origin coordinates(cm)($p=(x^2+y^2)^{1/2}$,z)

3rd card(2D35.18)β_{max}, β_{min}

4th card(D35.18,44X,I1)Maximum(cm)size of the body and type of the crosssecticnal shape(number code as in the subroutine ARGON)

Then the quadrature information follows(If this is the first ensemble to be solved) :

1st Card (I4) Number of quadrature points.

2nd and the others (2D35.18) argument and weights for positive values of the argument only.

If a sequence of problems have to be solved, same type of cards follow, but no quadrature cards are necessary after other ensemble descriptions.

REFERENCES

1. J.D.Jackson,"Classical Electrodynamics",John Wiley and Sons,1962.

2. J.Van Bladel,"Electromagnetic Fields",Mc Graw Hill,1964.

3. K.K.Mei, J.G.Van Bladel,"Scattering by Perfectly Conducting Rectangular Cylinders",IEEE Transactions on Antennas and Propagation,pp 185-192,March 1963.

4. P.C.Waterman,"Matrix Formulation of the Electromagnetic Scattering",Proc. IEEE,Vol53, p805,August 1965.

5. John H.Bruning,Yuen T.Lo,"Multiple Scattering of EM waves by Spheres,Multipole and Ray-Optical Solution",IEEE Transactions on Antennas and Propagation,1971

6. A.Baghdasarian and D.J.Angelakos, "Solution of Circular Loop Antennas and Scattering from Conducting Loops by Numerical Methods",Proc.IEEE,Vol.53,No 8,p.818,Aug.1965

7. Victor Twersky,"Multiple Scattering of Electromagnetic Waves by Arbitary Configurations",Journal of Mathematical Physics, Vol.8, No 3,pp589-610, March 1967.

8. Altunkan Hizal,"A New Approach to the Use of the Multipole Expansion Method in Electromagnetic Scattering Problems", Ph.D.Dissertation,METU EE Dept,Ankara,July 1970.

9. L.Fox,"An Introduction to Numerical Linear Algebra", N.Y.,Oxford University Press,pp 99-124,1965.

10. M.G.Salvadori,M.L.Baron,"Numerical Methods in Engineering" Prentice Hall of India,pp 1-47,1966.

11. C.B.Germain,"Programming the IBM 360",Prentice Hall,1967.

12. B.Carnaham,H.A.Luther,O.Wilkes,"Applied Numerical Methods" pp 69-140,269-340, John Wiley and Sons,1969.

13. D.Mc Cracken,W.S.Dorn,"Numerical Methods and Fortran Programming ",pp 160-194,256-284,John Wiley and Sons,1966.

14. J.M.Mc Gormic,M.G.Salvadori,"Numerical Methods in Fortran", Prentice Hall of India,1968.

15. A.Lichnerowictz,"Elements of Tensor Calculus",Methuen and Co ltd.,pp 53-73,1970.

16. R.C.Wrede,"Vector and Tensor Analysis",John Wiley and Sons,1963,pp 254-383.

17. P.Henrici,"Elements of Numerical Analysis",John Wiley and Sons,1964,pp 97-119.

18. Abromowitz and I.Stegun,"Handbook of Mathematical Functions", Dover Edition,1965.

19. H.J.Bowdler,R.S.Martin,G.Peters,J.H.Wilkinson,"Solution of Real and Complex Systems of Linear Equations,Handbook Series Linear Algebra",Numerische Mathematik,Vol 8, pp 217-234,1966.

20.Murray R.G.Spiegel,"Mathematical Handbook of Formulas and Tables",Schaum's Outline Series,Mc Graw Hill,1968.

```
      IMPLICIT REAL*8(A-H,O-Z)                                          CUBE   1
      REAL*8 QA(9),QB(9),QSW(4)                                         CUBE   2
      COMMON MQX(4),RSW(4)                                              CUBE   3
      COMMON/FLICK/AA(9),BB(9),TMX(9),TMN(9),ZBCX(4,9),RMAX,RMIN,NSF    CUBE   4
      EXTERNAL ARGON                                                    CUBE   5
   16 READ(1,22,END=1453)NSF,ICODE,I1,I2,I3,I4,I5,I6,I7,INKA,FSTR,FINC, CUBE   6
     1((ZBCX(I,J),I=1,4),QA(J),QB(J),TMX(J),TMN(J),QSW(J),MQX(J),       CUBE   7
     2J=1,NSF)                                                          CUBE   8
   22 FORMAT(10I1,10X,2G30.15/(4A8/2G35.18/2G35.18/G35.18,44X,I1))      CUBE   9
      FRAC=FSTR-FINC                                                    CUBE  10
      DO 38 I=1,INKA                                                    CUBE  11
      RMAX=0.D00                                                        CUBE  12
      RMIN=0.D00                                                        CUBE  13
      IF(ICODE.NE.0)RMIN=25.D00                                         CUBE  14
      FRAC=FRAC+FINC                                                    CUBE  15
      FKAC=(0.2D00*3.141592653589793/3.D00)*FRAC                        CUBE  16
      DO 34 J=1,NSF                                                     CUBE  17
      AA(J)=QA(J)*FKAC                                                  CUBE  18
      BB(J)=QB(J)*FKAC                                                  CUBE  19
   34 RSW(J)=QSW(J)*FKAC                                                CUBE  20
      CALL VENUS(ARGON,I1,I2,I3,I4,I5,I6,I7)                            CUBE  21
      WRITE(3,6)FRAC                                                    CUBE  22
    6 FORMAT(///T8,3X,'FREQUENCY = ',F12.7,' GHZ')                      CUBE  23
   38 CONTINUE                                                          CUBE  24
      GO TO 16                                                          CUBE  25
 1453 STOP                                                              CUBE  26
      END                                                               CUBE  27
```

```
      SUBROUTINE ARGON(R,RT,COZE,SIZE,ZETA,LSF)                         ARGO   1
      IMPLICIT REAL*8(A-H,O-Z)                                          ARGO   2
      COMMON MQX(4),RSW(4)                                              ARGO   3
      BEK=RSW(LSF)                                                      ARGO   4
      MET=MQX(LSF)                                                      ARGO   5
      GO TO (1,2,3,4,5,6,7,8),MET                                       ARGO   6
C-----------------------------------------------------------------------ARGO   7
C     SPHERE                                                            ARGO   8
C-----------------------------------------------------------------------ARGO   9
    1 R=BEK                                                             ARGO  10
      RT=0.D00                                                          ARGO  11
      RETURN                                                            ARGO  12
C-----------------------------------------------------------------------ARGO  13
C     PROLATE SPHEROID                                                  ARGO  14
C-----------------------------------------------------------------------ARGO  15
    2 R=BEK/(2.D00-SIZE)                                                ARGO  16
      RT=R*COZE/(2.D00-SIZE)                                            ARGO  17
      RETURN                                                            ARGO  18
C-----------------------------------------------------------------------ARGO  19
C     VERTICAL EIGHT                                                    ARGO  20
C-----------------------------------------------------------------------ARGO  21
    3 R=BEK*(2.D00-COZE**2+SIZE**2)/3.D00                               ARGO  22
      RT=4.D00*BEK*SIZE*COZE/3.D00                                      ARGO  23
      RETURN                                                            ARGO  24
C-----------------------------------------------------------------------ARGO  25
C     LONG BEAN                                                         ARGO  26
C-----------------------------------------------------------------------ARGO  27
    4 CST=DEXP(-SIZE)                                                   ARGO  28
      R=BEK*DSIN(CST)                                                   ARGO  29
      RT=-BEK*DCOS(CST)*CST*COZE                                        ARGO  30
      RETURN                                                            ARGO  31
C-----------------------------------------------------------------------ARGO  32
C     HORIZONTAL EIGHT                                                  ARGO  33
C-----------------------------------------------------------------------ARGO  34
    5 R=BEK*(2.D00+COZE*COZE-SIZE*SIZE)/3.D00                           ARGO  35
      RT=-4.D00*BEK*SIZE*COZE/3.D00                                     ARGO  36
```

```
      RETURN                                                            ARGO  37
C-----------------------------------------------------------------------ARGO  38
C     SHORT BEAN                                                        ARGO  39
C-----------------------------------------------------------------------ARGO  40
    6 R=BEK*DSIN(2.D00-SIZE)/3.D00                                      ARGO  41
      RT=-BEK*DCOS(2.D00-SIZE)*COZE/3.D00                               ARGO  42
      RETURN                                                            ARGO  43
C-----------------------------------------------------------------------ARGO  44
C     STAR OF FOUR LEAVES                                               ARGO  45
C-----------------------------------------------------------------------ARGO  46
    7 R=BEK*(2.D00+DCOS(4.D00*ZETA))/3.D00                              ARGO  47
      RT=-4.D00*BEK*DSIN(4.D00*ZETA)/3.D00                              ARGO  48
      RETURN                                                            ARGO  49
C-----------------------------------------------------------------------ARGO  50
C     OBLONG                                                            ARGO  51
C-----------------------------------------------------------------------ARGO  52
    8 R=BEK*(2.D00-SIZE)/3.D00                                          ARGO  53
      RT=-BEK*COZE/3.D00                                                ARGO  54
      RETURN                                                            ARGO  55
      END                                                               ARGO  56
```

```
      SUBROUTINE VENUS(ILKNUR,Q1,Q2,Q3,Q4,Q5,Q6,Q7)                     VENU   1
      IMPLICIT COMPLEX*16(A-H),REAL*8(O-Z)                              VENU   2
      REAL*8 XQ(200),YQ(200),KSI(5),ETA(5),AA(9),BB(9),AABS,BABS,       VENU   3
     1     COTA,COZE                                                    VENU   4
     2     ,P(5),PT(5),PK(5),                                           VENU   5
     3     SIFIR(37),DOKSAN(37),SS(2,5,2,2,5,2,2)                       VENU   6
      REAL TYPX(2),ADI(2),ADM(3),XO(74),X1(74),ZO(74),Z3(74),AB1(3),    VENU   7
     1     AB2(3),ORD(3),ARAS1(25),ARAS2(25),STAR(74)                   VENU   8
      INTEGER ITET(37),IPROB/0/,Q1,Q2,Q3,Q4,Q5,Q6,Q7                    VENU   9
      COMPLEX*16 ADA(400),S(5,2,2,5,2,2),H1(5),H2(5),B(5,2,2),U(5,2,2), VENU  10
     1     US(5,2),SEV,SEB,Z1(5,2),Z2(5,2),Y1(5,2),Y2(5,2)              VENU  11
      DIMENSION ZBCX(4,9)                                               VENU  12
      DIMENSION TMX(9),TMN(9)                                           VENU  13
      COMMON/FLICK/AA,BB,TMX,TMN,ZBCX,RMAX,RMIN,NSF                     VENU  14
      EQUIVALENCE (SS(1),S(1),ADA(1))                                   VENU  15
      IF(IPROB)104,781,104                                              VENU  16
  781 TAB=3.14159265358979 3                                            VENU  17
      TAC=TAB/180.D00                                                   VENU  18
      TAB=TAB*2.D00                                                     VENU  19
C-----------------------------------------------------------------------VENU  20
C     READ QUADRATURE POINTS                                            VENU  21
C-----------------------------------------------------------------------VENU  22
      READ(1,101)NQD,(XQ(I),YQ(I),I=1,NQD,2)                            VENU  23
  101 FORMAT(I4/(2G35.30))                                              VENU  24
      DO 102 I=2,NQD,2                                                  VENU  25
      XQ(I)=-XQ(I-1)                                                    VENU  26
  102 YQ(I)=YQ(I-1)                                                     VENU  27
C-----------------------------------------------------------------------VENU  28
C     PREPARE INDICAL FUNCTIONS                                         VENU  29
C-----------------------------------------------------------------------VENU  30
      STRG=4.D00*3.14159265358979 3                                     VENU  31
      DO 103 I=1,5                                                      VENU  32
      KSI(I)=DSQRT((2*I+1)/STRG)                                        VENU  33
  103 ETA(I)=KSI(I)/DFLOAT(I*I+I)                                       VENU  34
C-----------------------------------------------------------------------VENU  35
C     START OF THE PROBLEM                                              VENU  36
```

```
C-----------------------------------------------------------------------VENU  37
  104  CONTINUE                                                          VENU  38
       IPROB=IPROB+1                                                     VENU  39
C-----------------------------------------------------------------------VENU  40
C      CLEAR UP THE INTEGRATION AREA                                     VENU  41
C-----------------------------------------------------------------------VENU  42
       DO 106 I=1,400                                                    VENU  43
  106  ADA(I)=(0.D00,0.D00)                                              VENU  44
C-----------------------------------------------------------------------VENU  45
C      START OF INTEGRATION                                              VENU  46
C-----------------------------------------------------------------------VENU  47
       DO 110 LSF=1,NSF                                                  VENU  48
       TAF=TAC*(TMX(LSF)-TMN(LSF))/2.D00                                 VENU  49
       TAK=TAC*(TMX(LSF)+TMN(LSF))/2.D00                                 VENU  50
       TAM=TAF*TAB                                                       VENU  51
       ZAVL=AA(LSF)                                                      VENU  52
       ZBVL=BB(LSF)                                                      VENU  53
       DO 110 LQD=1,NQD                                                  VENU  54
       ZETA=TAF*XQ(LQD)+TAK                                              VENU  55
       COZE=DCOS(ZETA)                                                   VENU  56
       SIZE=DSIN(ZETA)                                                   VENU  57
       WEB=TAM*YQ(LQD)                                                   VENU  58
       CALL ILKNUR(AABS,BABS,COZE,SIZE,ZETA,LSF)                         VENU  59
       ZZ=AABS*SIZE                                                      VENU  60
       PP=AABS*COZE                                                      VENU  61
       ZZ=ZBVL+ZZ                                                        VENU  62
       PP=PP+ZAVL                                                        VENU  63
       SEL=AABS*PP*WEB                                                   VENU  64
       SEM=-BABS*PP*WEB                                                  VENU  65
       RR=DSQRT(ZZ**2+PP**2)                                             VENU  66
       IF(RR.GT.RMAX)RMAX=RR                                             VENU  67
       IF(RR.LT.RMIN)RMIN=RR                                             VENU  68
       COTA=ZZ/RR                                                        VENU  69
       SOTA=PP/RR                                                        VENU  70
       ZCS=COZE*COTA-SIZE*SOTA                                           VENU  71
       ZSS=COZE*SOTA+SIZE*COTA                                           VENU  72
```

```
      TSR=SEL*ZSS+SEM*ZCS                                               VENU  73
      TST=SEL*ZCS-SEM*ZSS                                               VENU  74
      CALL BADE(P,PT,PK,COTA,SOTA,5)                                    VENU  75
      CALL YAPRAK(Z1,Y1,5,RR)                                           VENU  76
      DO 110 KM=1,5                                                     VENU  77
      YWA=ETA(KM)*TSR                                                   VENU  78
      YWB=-YWA/RR                                                       VENU  79
      YWC=KSI(KM)*TST*P(KM)/RR                                          VENU  80
      DO 110 LM=1,5                                                     VENU  81
      RIE=YWA*ETA(LM)*(PK(LM)*PT(KM)+PK(KM)*PT(LM))                     VENU  82
      RIM=YWB*ETA(LM)*(PK(LM)*PK(KM)+PT(LM)*PT(KM))                     VENU  83
      TIM=YWC*ETA(LM)*PT(LM)                                            VENU  84
      DO 110 IA=1,2                                                     VENU  85
      AK=(0.D00,1.D00)*DCONJG(Z1(LM,IA))                                VENU  86
      ARIE=AK*DCMPLX(RIE,0.D00)                                         VENU  87
      ARIM=AK*DCMPLX(RIM,0.D00)                                         VENU  88
      ATIM=AK*DCMPLX(TIM,0.D00)                                         VENU  89
      DO 110 IB=1,2                                                     VENU  90
      S(LM,2,IA,KM,1,IB)=S(LM,2,IA,KM,1,IB)+ARIE*Z1(KM,IB)              VENU  91
  110 S(LM,2,IA,KM,2,IB)=S(LM,2,IA,KM,2,IB)+ARIM*Y1(KM,IB)+ATIM*Z1(KM,IBVENU  92
     1      )                                                           VENU  93
C-----------------------------------------------------------------------VENU  94
C     FILL IN CM + CE MATRICES                                          VENU  95
C-----------------------------------------------------------------------VENU  96
      DO 111 I=1,5                                                      VENU  97
      DO 111 J=1,5                                                      VENU  98
      DO 111 L=1,2                                                      VENU  99
      DO 111 M=1,2                                                      VENU 100
      S(I,1,L,J,1,M)=-S(I,2,L,J,2,M)                                    VENU 101
  111 S(I,1,L,J,2,M)=S(I,2,L,J,1,M)                                     VENU 102
C-----------------------------------------------------------------------VENU 103
C     GENERATE B MATRICE                                                VENU 104
C-----------------------------------------------------------------------VENU 105
      IF(RMIN-1.D-03)307,307,308                                        VENU 106
  307 DO 688 I=1,2                                                      VENU 107
      DO 688 J=1,2                                                      VENU 108
```

```
      RENGIN=-DFLOAT(I/2+J/2)                                           VENU 109
      DO 688 K=1,5                                                      VENU 110
  688 SS(1,K,1,I,K,1,J)=SS(1,K,1,I,K,1,J)+RENGIN                        VENU 111
      DO 689 I=1,5                                                      VENU 112
      B(I,2,1)=(0.D00,0.D00)                                            VENU 113
  689 B(I,2,2)=(0.D00,0.D00)                                            VENU 114
      GO TO 309                                                         VENU 115
  308 CALL FIDAN(H1,H2,5,RMIN)                                          VENU 116
      DO 112 I=1,5                                                      VENU 117
      STRG=H1(I)                                                        VENU 118
      B(I,2,1)=(1.D00,0.D00)                                            VENU 119
  112 B(I,2,2)=H2(I)/STRG                                               VENU 120
  309 CALL FIDAN(H1,H2,5,RMAX)                                          VENU 121
      DO 113 I=1,5                                                      VENU 122
      STRG=H1(I)                                                        VENU 123
      B(I,1,1)=STRG/H1(I)                                               VENU 124
  113 B(I,1,2)=H2(I)/H1(I)                                              VENU 125
C-----------------------------------------------------------------------VENU 126
C     GENERATE U MATRICE                                                VENU 127
C-----------------------------------------------------------------------VENU 128
      SEV=(0.D00,1.D00)                                                 VENU 129
      DO 114 I=1,5                                                      VENU 130
      SEB=-SEV                                                          VENU 131
      SEV=SEV*(0.D00,1.D00)                                             VENU 132
      VANTUZ=DSQRT(3.141592653589793*(8*I+4))                           VENU 133
      U(I,1,2)=SEV*VANTUZ                                               VENU 134
      U(I,2,2)=SEB*VANTUZ                                               VENU 135
      U(I,1,1)=B(I,1,1)*U(I,1,2)                                        VENU 136
  114 U(I,2,1)=B(I,1,1)*U(I,2,2)                                        VENU 137
      IPAG=0                                                            VENU 138
C-----------------------------------------------------------------------VENU 139
C     OUTPUT SECTION                                                    VENU 140
C-----------------------------------------------------------------------VENU 141
      IF(Q1-1)782,783,782                                               VENU 142
  783 DO 115 KM=1,2                                                     VENU 143
      DO 115 IA=1,2                                                     VENU 144
```

```
      IPAG=IPAG+1                                                       VENU 145
      WRITE(3,116) IPROB,IPAG,RMAX,RMIN,NSF,(I,AA(I),I,BB(I),           VENU 146
     1      I,TMX(I),I,TMN(I),I,(ZBCX(J,I),J=1,4),                      VENU 147
     2      I=1,NSF)                                                    VENU 148
  116 FORMAT(1H1,T8,'PROBLEM - ',I2,T118,'PAGE  ',I2///                 VENU 149
     1    T8,'K*RMAX = ',F5.2,2X/T8,'K*RMIN = ',F5.2/                   VENU 150
     2    T8,'#OF SF = ',I5///(T8,'P',I1,' = ',F5.2,5X,'Z',I1,' = ',F5.2VENU 151
     1    ,5X,'BMX',I1,' =' ,F7.2,5X,'BMN',I1,' =' ,F7.2                VENU 152
     1,5X,'R',I1,' = ',4A8))                                            VENU 153
      DO 115 IB=1,2                                                     VENU 154
      DO 115 M=1,2                                                      VENU 155
  115 WRITE(3,2116)TYPX(M),ADI(KM),IA,IB,((SS(M,I,2,IA,J,KM,IB),J=1,5), VENU 156
     1      I=1,5)                                                      VENU 157
 2116 FORMAT(//T63,A4,'(',A2,I1,I1,')'//(T2,5D25.13))                   VENU 158
  782 IF(Q2-1)125,785,125                                               VENU 159
C-----------------------------------------------------------------------VENU 160
C     GENERATE AND SOLVE BOUNDARY RELATIONS                             VENU 161
C-----------------------------------------------------------------------VENU 162
  785 DO 117 K=1,5                                                      VENU 163
      DO 117 J=1,2                                                      VENU 164
      US(K,J)=U(K,J,1)                                                  VENU 165
      DO 117 I=1,2                                                      VENU 166
      S(K,2,I,K,2,J)=S(K,2,I,K,2,J)-B(K,I,J)                            VENU 167
  117 S(K,1,I,K,1,J)=S(K,1,I,K,1,J)-B(K,I,J)                            VENU 168
      CALL CAPRI(ADA,U,20)                                              VENU 169
      DO 118 I=1,5                                                      VENU 170
      US(I,1)=US(I,1)+B(I,1,1)*U(I,1,1)+B(I,1,2)*U(I,1,2)               VENU 171
  118 US(I,2)=US(I,2)+B(I,1,1)*U(I,2,1)+B(I,1,2)*U(I,2,2)               VENU 172
C-----------------------------------------------------------------------VENU 173
C      OUTPUTS U,US                                                     VENU 174
C-----------------------------------------------------------------------VENU 175
      IF(Q3-1)786,787,786                                               VENU 176
  787 IPAG=IPAG+1                                                       VENU 177
      WRITE(3,116) IPROB,IPAG,RMAX,RMIN,NSF,(I,AA(I),I,BB(I),           VENU 178
     1      I,TMX(I),I,TMN(I),I,(ZBCX(J,I),J=1,4),                      VENU 179
     2      I=1,NSF)                                                    VENU 180
```

```
      DO 119 I=1,2                                                      VENU 181
  119 WRITE(3,120)(ADM(I),K=1,2),((U(L,M,I),M=1,2),L=1,5)               VENU 182
      WRITE(3,120)(ADM(3),K=1,2),((US(L,M),M=1,2),L=1,5)                VENU 183
  120 FORMAT(///T24,'ALPHA E',A4,T98,'ALPHA M',A4//                     VENU 184
     1    (T2,2D25.13,25X,2D25.13))                                     VENU 185
C-----------------------------------------------------------------------VENU 186
C     BISTATIC CROSS SECTIONS                                           VENU 187
C-----------------------------------------------------------------------VENU 188
  786 IF(Q4-1)125,788,125                                               VENU 189
  788 STRG=1.D00/6.283185307179586                                      VENU 190
      RAA=3.141592653589793/36.D00                                      VENU 191
      CZERO=(0.D00,-1.D00)                                              VENU 192
      DO 641 I=1,5                                                      VENU 193
      CNINE=CZERO                                                       VENU 194
      CZERO=CZERO*(0.D00,-1.D00)                                        VENU 195
      SEV=DCMPLX(STRG *DSQRT(DFLOAT(2*I+1))/DFLOAT(I*(I+1)),0.D00)      VENU 196
      US(I,1)=US(I,1)*SEV*CNINE                                         VENU 197
  641 US(I,2)=US(I,2)*SEV*CZERO                                         VENU 198
      DO 122 I=1,37                                                     VENU 199
      TETA=RAA*DFLOAT(I-1)                                              VENU 200
      COTA=DCOS(TETA)                                                   VENU 201
      SOTA=DSIN(TETA)                                                   VENU 202
      ITET(I)=5*(I-1)                                                   VENU 203
      CALL BADE(P,PT,PK,COTA,SOTA,5)                                    VENU 204
      CNINE=(0.D00,0.D00)                                               VENU 205
      CZERO=CNINE                                                       VENU 206
      DO 121 K=1,5                                                      VENU 207
      APTO=DCMPLX(PT(K),0.D00)                                          VENU 208
      APKO=DCMPLX(PK(K),0.D00)                                          VENU 209
      CZERO=US(K,2)*APKO+US(K,1)*APTO+CZERO                             VENU 210
  121 CNINE=US(K,2)*APTO+US(K,1)*APKO+CNINE                             VENU 211
      SIFIR(I)=CZERO*DCONJG(CZERO)                                      VENU 212
      DOKSAN(I)=CNINE*DCONJG(CNINE)                                     VENU 213
      XO(I)=SIFIR(I)*SOTA                                               VENU 214
      ZO(I)=SIFIR(I)*COTA                                               VENU 215
      X1(I)=DOKSAN(I)*SOTA                                              VENU 216
```

```
      Z3(I)=DOKSAN(I)*COTA                                              VENU 217
      LSF=I+37                                                          VENU 218
      XO(LSF)=-XO(I)                                                    VENU 219
      X1(LSF)=-X1(I)                                                    VENU 220
      ZO(LSF)=ZO(I)                                                     VENU 221
  122 Z3(LSF)=Z3(I)                                                     VENU 222
C-----------------------------------------------------------------------VENU 223
C     OUTPUT BISTATIC-CROSS SECTIONS                                    VENU 224
C-----------------------------------------------------------------------VENU 225
      IF(Q5-1)789,790,789                                               VENU 226
  790 IPAG=IPAG+1                                                       VENU 227
      WRITE(3,116) IPROB,IPAG,RMAX,RMIN,NSF,(I,AA(I),I,BB(I),           VENU 228
     1      I,TMX(I),I,TMN(I),I,(ZBCX(J,I),J=1,4),                      VENU 229
     2      I=1,NSF)                                                    VENU 230
      WRITE(3,123)(ITET(I),SIFIR(I),DOKSAN(I),I=1,37)                   VENU 231
  123 FORMAT(///T8,'THETA<DEG>',10X,'FOR<PHI=0>',10X,'FOR<PHI=90>'//    VENU 232
     1    (T8,3X,I3,7X,D20.10,D20.10))                                  VENU 233
  789 IF(Q6-1)791,792,791                                               VENU 234
  792 CALL ZEYNEP(XO,ZO,STAR,74,ORD,AB1,ARAS1,1)                        VENU 235
  791 IF(Q7-1)125,794,125                                               VENU 236
  794 CALL ZEYNEP(X1,Z3,STAR,74,ORD,AB2,ARAS2,2)                        VENU 237
  125 RETURN                                                            VENU 238
      DATA TYPX/8HREALIMAG/,ADI/8HIEIEIMIM/,ADM/12HI(1)I(2)S    /,      VENU 239
     1    AB1/12H        XUNIT/,AB2/12H        YUNIT/,ORD/12H       ZUNIT/VENU 240
     1,   ARAS1/100HBISTATIC CROSS-SECTION FOR PHI=0 DEGREE             VENU 241
     5                                                                /, VENU 242
     1ARAS2/100HBISTATIC CROSS-SECTION FOR PHI=90 DEGREES               VENU 243
     5                                                               /VENU 244
     6,STAR/74*4H****/                                                  VENU 245
      END                                                               VENU 246
```

```
      SUBROUTINE ZEYNEP(X,Y,Z,N,ORD,ASB,ARAS,ICHART)                    ZEYN   1
      DIMENSION X(N),Y(N),Z(N),ORD(3),ASB(3),ARAS(25),VAR(5)            ZEYN   2
 1001 FORMAT(1H1,T3,2A4,1H(,A4,1H))                                     ZEYN   3
 1002 FORMAT(T2,E13.6,1H-)                                              ZEYN   4
 1003 FORMAT(1H+,T15,102A1)                                             ZEYN   5
 1004 FORMAT(T16,101(1H|)/T16,5(1H|,19X))                               ZEYN   6
 1005 FORMAT(T13,5(E13.6,7X),2A4,1H(,A4,1H))                            ZEYN   7
 1006 FORMAT(///T10,6HCHART ,I3,1H-,25A4)                               ZEYN   8
C-----------------------------------------------------------------------ZEYN   9
C     SORT  ORDINATE VARIABLES IN DESCENDING ORDER                      ZEYN  10
C-----------------------------------------------------------------------ZEYN  11
   10 DO 11 I=1,N                                                       ZEYN  12
      JASMIN=I                                                          ZEYN  13
      COBRA=Y(I)                                                        ZEYN  14
      DO 66 J=I,N                                                       ZEYN  15
      IF(Y(J)-COBRA)66,66,64                                            ZEYN  16
   64 COBRA=Y(J)                                                        ZEYN  17
      JASMIN=J                                                          ZEYN  18
   66 CONTINUE                                                          ZEYN  19
      IF(JASMIN-I)70,11,70                                              ZEYN  20
   70 Y(JASMIN)=Y(I)                                                    ZEYN  21
      Y(I)=COBRA                                                        ZEYN  22
      COBRA=X(JASMIN)                                                   ZEYN  23
      X(JASMIN)=X(I)                                                    ZEYN  24
      X(I)=COBRA                                                        ZEYN  25
      COBRA=Z(JASMIN)                                                   ZEYN  26
      Z(JASMIN)=Z(I)                                                    ZEYN  27
      Z(I)=COBRA                                                        ZEYN  28
   11 CONTINUE                                                          ZEYN  29
C-----------------------------------------------------------------------ZEYN  30
C     FIND BASE AND AMPLITUDE WIDTHS                                    ZEYN  31
C-----------------------------------------------------------------------ZEYN  32
      YWIDTH=ABS(Y(1)-Y(N))                                             ZEYN  33
      XMIN=X(1)                                                         ZEYN  34
      XMAX=X(1)                                                         ZEYN  35
      DO 20 I=1,N                                                       ZEYN  36
```

```
      IF(X(I).GT.XMAX)XMAX=X(I)                                         ZEYN  37
      IF(X(I).LT.XMIN)XMIN=X(I)                                         ZEYN  38
   20 CONTINUE                                                          ZEYN  39
      XWIDTH=ABS(XMAX-XMIN)                                             ZEYN  40
C-----------------------------------------------------------------------ZEYN  41
C     PREPARATION BEFORE ACTUAL PRINTING                                ZEYN  42
C-----------------------------------------------------------------------ZEYN  43
      WRITE(3,1001)(ORD(I),I=1,3)                                       ZEYN  44
C-----------------------------------------------------------------------ZEYN  45
C     START ACTUAL GRAPHING                                             ZEYN  46
C-----------------------------------------------------------------------ZEYN  47
      XPITCH=XWIDTH/100.0                                               ZEYN  48
      YPITCH=YWIDTH/50.0                                                ZEYN  49
      IF(XWIDTH.EQ.0.0)XPITCH=0.01                                      ZEYN  50
      IF(YWIDTH.EQ.0.0)YPITCH=0.02                                      ZEYN  51
      JASMIN=1                                                          ZEYN  52
      DO 30 I=1,51                                                      ZEYN  53
C-----------------------------------------------------------------------ZEYN  54
C     SINGLE ROW PRINTING                                               ZEYN  55
C-----------------------------------------------------------------------ZEYN  56
      COBRA=Y(1)-(I-1)*YPITCH                                           ZEYN  57
      WRITE(3,1002)COBRA                                                ZEYN  58
   31 IF(ABS(Y(JASMIN)-COBRA).GT.(YPITCH/2.0))GO TO 30                  ZEYN  59
      JESPA=(X(JASMIN)-XMIN)/XPITCH+1                                   ZEYN  60
      WRITE(3,1003)(BLANK,LALE=1,JESPA),Z(JASMIN)                       ZEYN  61
      IF(JASMIN.EQ.N)GO TO 30                                           ZEYN  62
      JASMIN=JASMIN+1                                                   ZEYN  63
      GO TO 31                                                          ZEYN  64
   30 CONTINUE                                                          ZEYN  65
C-----------------------------------------------------------------------ZEYN  66
C     UNDERSCALING                                                      ZEYN  67
C-----------------------------------------------------------------------ZEYN  68
      WRITE(3,1004)                                                     ZEYN  69
      DO 41 J=1,5                                                       ZEYN  70
   41 VAR(J)=XMIN+XPITCH*(J-1)*20                                       ZEYN  71
      WRITE(3,1005)(VAR(J),J=1,5),(ASB(L),L=1,3)                        ZEYN  72
```

```
      WRITE(3,1006)ICHART,(ARAS(J),J=1,25)                ZEYN  73
      RETURN                                              ZEYN  74
      DATA BLANK/1H /                                     ZEYN  75
      END                                                 ZEYN  76
```

```
      SUBROUTINE CAPRI(A,B,N)                                           CAPR   1
      COMPLEX*16 A(N,N),B(N),E                                          CAPR   2
      REAL*8 DTMP,DMAX                                                  CAPR   3
      NN=N-1                                                            CAPR   4
      DO 106 K=1,NN                                                     CAPR   5
      KK=K+1                                                            CAPR   6
      LMAX=0                                                            CAPR   7
      DMAX=0.D00                                                        CAPR   8
      DO 102 J=K,N                                                      CAPR   9
      DTMP=A(J,K)*DCONJG(A(J,K))                                        CAPR  10
      IF(DTMP-DMAX)102,102,101                                          CAPR  11
  101 DMAX=DTMP                                                         CAPR  12
      LMAX=J                                                            CAPR  13
  102 CONTINUE                                                          CAPR  14
      IF(LMAX-K)103,105,103                                             CAPR  15
  103 E=B(LMAX)                                                         CAPR  16
      B(LMAX)=B(K)                                                      CAPR  17
      B(K)=E                                                            CAPR  18
      DO 104 L=K,N                                                      CAPR  19
      E=A(LMAX,L)                                                       CAPR  20
      A(LMAX,L)=A(K,L)                                                  CAPR  21
  104 A(K,L)=E                                                          CAPR  22
  105 DO 106 L=KK,N                                                     CAPR  23
      E=-A(L,K)/A(K,K)                                                  CAPR  24
      B(L)=B(L)+E*B(K)                                                  CAPR  25
      DO 106 J=KK,N                                                     CAPR  26
  106 A(L,J)=A(L,J)+E*A(K,J)                                            CAPR  27
      B(N)=B(N)/A(N,N)                                                  CAPR  28
      DO 108 L=1,NN                                                     CAPR  29
      K=N-L                                                             CAPR  30
      KK=K+1                                                            CAPR  31
      DO 107 J=KK,N                                                     CAPR  32
  107 B(K)=B(K)-A(K,J)*B(J)                                             CAPR  33
  108 B(K)=B(K)/A(K,K)                                                  CAPR  34
      RETURN                                                            CAPR  35
      END                                                               CAPR  36
```

```
      SUBROUTINE YAPRAK(H,G,N,X)                                        YAPR   1
      IMPLICIT COMPLEX*16(A-H),REAL*8(O-Z)                              YAPR   2
      COMPLEX*16 G(N,2),H(N,2)                                          YAPR   3
      SARAY(A)=A                                                        YAPR   4
      DRILL(A)=DCMPLX(SARAY(A),0.D00)                                   YAPR   5
      C=DCMPLX(DSIN(X)/X,DCOS(X)/X)                                     YAPR   6
      H(1,2)=C*DCMPLX(1.D00/X,1.D00)                                    YAPR   7
      XX=X*X                                                            YAPR   8
      H(2,2)=C*DCMPLX((3.D00-XX)/XX,3.D00/X)                            YAPR   9
      G(1,2)=C*DCMPLX((XX-1.D00)/X  ,-1.D00)                            YAPR  10
      G(2,2)=X*H(1,2)-2.D00*H(2,2)                                      YAPR  11
      DO 101 I=3,N                                                      YAPR  12
      H(I,2)=(2*I-1)*H(I-1,2)/X-H(I-2,2)                                YAPR  13
  101 G(I,2)=X*H(I-1,2)-I*H(I,2)                                        YAPR  14
      DO 102 I=1,N                                                      YAPR  15
      H(I,1)=DRILL(H(I,2))                                              YAPR  16
  102 G(I,1)=DRILL(G(I,2))                                              YAPR  17
      RETURN                                                            YAPR  18
      END                                                               YAPR  19
```

```
      SUBROUTINE BADE(P,PT,PK,COTA,SOTA,N)                              BADE   1
      IMPLICIT REAL*8(A-H,O-Z)                                          BADE   2
      REAL*8 P(N),PT(N),PK(N)                                           BADE   3
      PK(1)=-1.D00                                                      BADE   4
      P(1)=PK(1)*SOTA                                                   BADE   5
      PT(1)=-COTA                                                       BADE   6
      PK(2)=-3.D00*COTA                                                 BADE   7
      P(2)=PK(2)*SOTA                                                   BADE   8
      PT(2)=-3.D00*(COTA**2-SOTA**2)                                    BADE   9
      DO 101 I=3,N                                                      BADE  10
      PK(I)=(2*I-1)*COTA*PK(I-1)/DFLOAT(I-1)-I*PK(I-2)/DFLOAT(I-1)      BADE  11
      P(I)=PK(I)*SOTA                                                   BADE  12
  101 PT(I)=I*COTA*PK(I)-(I+1)*PK(I-1)                                  BADE  13
      RETURN                                                            BADE  14
      END                                                               BADE  15
```

```
      SUBROUTINE FIDAN(H1,H2,N,X)                                       FIDA   1
      IMPLICIT COMPLEX*16(A-H),REAL*8(O-Z)                              FIDA   2
      COMPLEX*16 H1(N),H2(N)                                            FIDA   3
      C=DCMPLX(DSIN(X)/X,DCOS(X)/X)                                     FIDA   4
      H2(1)=C*DCMPLX(1.D00/X,1.D00)                                     FIDA   5
      XX=X*X                                                            FIDA   6
      H2(2)=C*DCMPLX((3.D00-XX)/XX,3.D00/X)                             FIDA   7
      DO 101 I=3,N                                                      FIDA   8
  101 H2(I)=(2*I-1)*H2(I-1)/X-H2(I-2)                                   FIDA   9
      DO 102 I=1,N                                                      FIDA  10
  102 H1(I)=DCONJG(H2(I))                                               FIDA  11
      RETURN                                                            FIDA  12
      END                                                               FIDA  13
```

K*RMAX = 2.85
K*RMIN = 0.0
#OF SF = 2

P1 = 0.0 Z1 = 0.0 BMX1 = 90.00 BMN1 = -90.00 R1 = LONG BEAN OF FOCAL MAXIMA 0.6CM.
P2 = 0.0 Z2 = 2.21 BMX2 = 90.00 BMN2 = -90.00 R2 = SPHERE OF RADIUS 0.3195 CM

REAL(IE11)

0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0

IMAG(IE11)

-0.1514943766639D-01	0.1429783590106D-01	0.1271185216963D-01	0.4010127659350D-02	0.1089658140189D-02
0.1429783590106D-01	0.2147050045582D-01	0.1035135984882D-01	0.3734659796018D-02	0.9995905609435D-03
0.1271185216963D-01	0.1035135984882D-01	0.5376395127255D-02	0.1886586604034D-02	0.5066672363029D-03
0.4010127659350D-02	0.3734659796018D-02	0.1886586604034D-02	0.6637752376662D-03	0.1791115347504D-03
0.1089658140189D-02	0.9995905609435D-03	0.5066672363029D-03	0.1791115347504D-03	0.4867195991243D-04

REAL(IE12)

0.8333626329013D-01	0.2402406149341D-01	0.3306097188328D 00	-0.5685398011349D 01	0.5475951667236D 02
0.2402406149438D-01	-0.2100470069449D-01	-0.5806285292997D-02	0.4222369755642D 00	-0.7958433748937D 01
-0.2773928115717D-02	-0.5806285283654D-02	-0.1915152486241D-01	-0.6398226205365D-02	0.4550824743069D 00
0.1423798852917D-03	-0.4567549649191D-02	-0.6398226157873D-02	-0.1220912329235D-01	-0.3603078173463D-02
-0.1461693695675D-04	-0.1205778331374D-02	-0.2331119220658D-02	-0.3603079267493D-02	-0.8209725159791D-02

IMAG(IE12)

-0.1514943766639D-01	0.1429783590106D-01	0.1271185216963D-01	0.4010127659350D-02	0.1089658140189D-02
0.1429783590106D-01	0.2147050045582D-01	0.1035135984882D-01	0.3734659796018D-02	0.9995905609435D-03
0.1271185216963D-01	0.1035135984882D-01	0.5376395127255D-02	0.1886586604034D-02	0.5066672363029D-03
0.4010127659350D-02	0.3734659796018D-02	0.1886586604034D-02	0.6637752376662D-03	0.1791115347504D-03
0.1089658140189D-02	0.9995905609435D-03	0.5066672363029D-03	0.1791115347504D-03	0.4867195991243D-04

K*RMAX = 2.85
K*RMIN = 0.0
#OF SF = 2

P1 = 0.0 Z1 = 0.0 BMX1 = 90.00 BMN1 = -90.00 R1 = LONG BEAN OF FOCAL MAXIMA 0.6CM.
P2 = 0.0 Z2 = 2.21 BMX2 = 90.00 BMN2 = -90.00 R2 = SPHERE OF RADIUS 0.3195 CM

REAL(IE21)

-0.8333626329013D-01	-0.2402406149438D-01	0.2773928115717D-02	-0.1423798852917D-03	0.1461693695675D-04
-0.2402406149341D-01	0.2100470069449D-01	0.5806285283654D-02	0.4567549649191D-02	0.1205778331374D-02
-0.3306097188328D 00	0.5806285292997D-02	0.1915152486241D-01	0.6398226157873D-02	0.2331119220658D-02
0.5685398011349D 01	-0.4222369755642D 00	0.6398226205365D-02	0.1220912329235D-01	0.3603079267493D-02
-0.5475951667236D 02	0.7958433748937D 01	-0.4550824743069D 00	0.3603078173463D-02	0.8209725159791D-02

IMAG(IE21)

-0.1514943766639D-01	0.1429783590106D-01	0.1271185216963D-01	0.4010127659350D-02	0.1089658140189D-02
0.1429783590106D-01	0.2147050045582D-01	0.1035135984882D-01	0.3734659796018D-02	0.9995905609435D-03
0.1271185216963D-01	0.1035135984882D-01	0.5376395127255D-02	0.1886586604034D-02	0.5066672363029D-03
0.4010127659350D-02	0.3734659796018D-02	0.1886586604034D-02	0.6637752376662D-03	0.1791115347504D-03
0.1089658140189D-02	0.9995905609435D-03	0.5066672363029D-03	0.1791115347504D-03	0.4867195991243D-04

REAL(IE22)

0.3601372333488D-17	-0.9652649874878D-12	0.3333836469485D 00	-0.5685540391235D 01	0.5475953128930D 02
0.9652455328351D-12	-0.5285485590867D-18	-0.9342924810272D-11	0.4268045252134D 00	-0.7957227970606D 01
-0.3333836469485D 00	0.9342939175951D-11	0.3364414866494D-17	-0.4749158178671D-10	0.4574135935275D 00
0.5685540391235D 01	-0.4268045252134D 00	0.4749159148354D-10	-0.9757819552370D-18	0.1094030710263D-08
-0.5475953128930D 02	0.7957227970606D 01	-0.4574135935275D 00	-0.1094030720671D-08	0.1036768327439D-17

IMAG(IE22)

0.7033073968588D 00	0.5507193917475D 01	0.1320345297834D 02	0.3536691311666D 03	0.7386570153228D 03
0.5507193917475D 01	0.1789742237258D 02	0.2797234878330D 03	0.1419310786827D 04	0.4157704479768D 05
0.1320345297834D 02	0.2797234878330D 03	0.1613502957742D 04	0.3624219282889D 05	0.2938912048720D 06
0.3536691311666D 03	0.1419310786827D 04	0.3624219282889D 05	0.3160680670886D 06	0.8964882295077D 07
0.7386570153228D 03	0.4157704479768D 05	0.2938912048720D 06	0.8964882295077D 07	0.1084992196875D 09

K*RMAX = 2.85
K*RMIN = 0.0
#OF SF = 2

P1 = 0.0 Z1 = 0.0 BMX1 = 90.00 BMN1 = -90.00 R1 = LONG BEAN OF FOCAL MAXIMA 0.6CM.
P2 = 0.0 Z2 = 2.21 BMX2 = 90.00 BMN2 = -90.00 R2 = SPHERE OF RADIUS 0.3195 CM

REAL(IM11)

0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0

IMAG(IM11)

-0.11530305230650 00	0.22219131690510-01	0.42390831441900-02	0.25329620828130-03	0.88819003072520-04
0.22219081862120-01	-0.93708346458840-02	-0.18571324471550-02	-0.13355279264280-02	-0.49851106909470-03
0.42391458980310-02	-0.18572589303230-02	-0.30238839952180-02	-0.13123820033180-02	-0.43153942749390-03
0.25313426192510-03	-0.13354125025580-02	-0.13123692800610-02	-0.63628042650480-03	-0.20698601962180-03
0.88932927461050-04	-0.49849060976820-03	-0.43137336742410-03	-0.20698151603990-03	-0.69084962778270-04

REAL(IM12)

-0.18604221678280-00	-0.28021457166810 00	0.21369787760860 01	-0.14380110634700 02	0.31163365494750 03
-0.54805747296680-01	-0.40513676730090 00	-0.30916787148490 00	0.19280841733480 01	-0.39837201094900 02
-0.11948194381070-01	-0.19057282691150-01	-0.49179732322790 00	-0.22252840834270 00	0.50366348778740 00
-0.33447280557790-02	-0.62530479845240-02	-0.10399884153170-01	-0.52250424820640 00	-0.10916770203230 00
-0.10227867589380-02	-0.10073197515260-02	-0.27704768602970-02	-0.77325604061090-02	-0.52817197385130 00

IMAG(IM12)

-0.11530305230650 00	0.22219131690510-01	0.42390831441900-02	0.25329620828130-03	0.88819003072520-04
0.22219081862120-01	-0.93708346458840-02	-0.18571324471550-02	-0.13355279264280-02	-0.49851106909470-03
0.42391458980310-02	-0.18572589303230-02	-0.30238839952180-02	-0.13123820033180-02	-0.43153942749390-03
0.25313426192510-03	-0.13354125025580-02	-0.13123692800610-02	-0.63628042650480-03	-0.20698601962180-03
0.88932927461050-04	-0.49849060976820-03	-0.43137336742410-03	-0.20698151603990-03	-0.69084962778270-04

K*RMAX = 2.85
K*RMIN = 0.0
#OF SF = 2

P1 = 0.0 Z1 = 0.0 BMX1 = 90.00 BMN1 = -90.00 R1 = LONG BEAN OF FOCAL MAXIMA 0.6CM.
P2 = 0.0 Z2 = 2.21 BMX2 = 90.00 BMN2 = -90.00 R2 = SPHERE OF RADIUS 0.3195 CM

REAL(IM21)

-0.81395778321740 00	0.54805796830310-01	0.11948154815290-01	0.33447615610300-02	0.10227298697450-02
0.28021435246760 00	-0.59486323270410 00	0.19057343564670-01	0.62528853421250-02	0.10072731007500-02
-0.21369824247250 01	0.30916725037260 00	-0.50820267680620 00	0.10399839032450-01	0.27710544075840-02
0.14381736886400 02	-0.19281048705950 01	0.22252679199210 00	-0.47749575058290 00	0.77313780512060-02
-0.31178090343640 03	0.39845773637650 02	-0.50376696755600 00	0.10916660582010 00	-0.47182883733800 00

IMAG(IM21)

-0.11530305230650 00	0.22219131690510-01	0.42390831441900-02	0.25329620828130-03	0.88819003072520-04
0.22219081862120-01	-0.93708346458840-02	-0.18571324471550-02	-0.13355279264280-02	-0.49851106909470-03
0.42391458980310-02	-0.18572589303230-02	-0.30238839952180-02	-0.13123820033180-02	-0.43153942749390-03
0.25313426192510-03	-0.13354125025580-02	-0.13123692800610-02	-0.63628042650480-03	-0.20698601962180-03
0.88932927461050-04	-0.49849060976820-03	-0.43137336742410-03	-0.20698151603990-03	-0.69084962778270-04

REAL(IM22)

-0.10000000000000 01	-0.22540877483780 00	0.21489269309010 01	-0.14376765873140 02	0.31163467767740 03
0.22540860517090 00	-0.10000000000050 01	-0.29011052792020 00	0.19343370586900 01	-0.39836193781800 02
-0.21489306191060 01	0.29010996768140 00	-0.10000000000340 01	-0.21212856931030 00	0.50643454216500 00
0.14378392158350 02	-0.19343579185800 01	0.21212269078380 00	-0.99999999987890 00	-0.10143632398110 00
-0.31178192622320 03	0.39844766317900 02	-0.50653744441630 00	0.10143404541400 00	-0.10000004111890 01

IMAG(IM22)

0.18600734647030 01	0.11595244465530 02	0.12484909336720 03	0.10206396066490 04	0.18967254957140 05
0.11595321804080 02	0.17942420222330 03	0.16776674660900 04	0.27196858926750 05	0.33637075517120 06
0.12484488660190 03	0.16776891881550 04	0.31041442506270 05	0.43018138384880 06	0.90460238864560 07
0.10207855423310 04	0.27196435421690 05	0.43017871215970 06	0.97006793972870 07	0.17912135622010 09
0.18964765907160 05	0.33635582199280 06	0.90465844908600 07	0.17910666091980 09	0.47933491644260 10

K*RMAX = 2.85
K*RMIN = 0.0
#OF SF = 2

P1 = 0.0 Z1 = 0.0 BMX1 = 90.00 BMN1 = -90.00 R1 = LONG BEAN OF FOCAL MAXIMA 0.6CM.
P2 = 0.0 Z2 = 2.21 BMX2 = 90.00 BMN2 = -90.00 R2 = SPHERE OF RADIUS 0.3195 CM

ALPHA EI(1)

0.31970706161550D 01	0.73923842966250D 01
-0.78098111627330D-01	0.68799346293530D 01
-0.99285282446640D 01	-0.16362490952160D 00
-0.10237391416180D 01	-0.10553761386710D 02
0.88810106604510D 01	0.38503103467520D 00

ALPHA MI(1)

0.17316585260740D 01	0.50941289351600D 01
-0.80534372919690D 01	0.18696967590790D-01
-0.10776787262960D 00	-0.87999568417510D 01
0.10625309738810D 02	0.92032339436320D 00
0.30791567374920D 00	0.14232348045150D 02

ALPHA EI(2)

0.15339444778620D 01	-0.38752557918740D 01
-0.10560569734000D 00	0.34581128571000D 00
-0.59419027234640D-02	0.10768291204920D-01
0.52226350796490D-03	-0.16424537551560D-02
-0.71843072229380D-05	0.32552041784010D-04

ALPHA MI(2)

-0.76219712613420D 00	0.17763644689200D 00
-0.46290849378900D-01	0.13210031736450D 00
0.10467773495500D-01	-0.11899046649130D-01
-0.17484367018050D-03	0.16754122293150D-04
-0.63590911477300D-05	0.10947673436520D-04

ALPHA ES

-0.14170538304800D 01	0.37113458860730D 01
0.15034761601440D 00	-0.65211338107490D 00
0.79812781435190D-02	-0.14638256343630D 00
-0.59118312271600D-02	-0.43132872712620D-01
-0.18377588266730D-02	-0.13317008031730D-01

ALPHA MS

0.93487664848380D 00	-0.87673735829270D 00
-0.19482169498210D 00	-0.11215876638400D 00
-0.14151916630320D 00	0.22540856261650D-01
-0.40273994781200D-01	0.13338398047470D-02
-0.11426952785000D-01	0.14649688968290D-02

K*RMAX = 2.85
K*RMIN = 0.0
#OF SF = 2

P1 = 0.0 Z1 = 0.0 BMX1 = 90.00 BMN1 = -90.00 R1 = LONG BEAN OF FOCAL MAXIMA 0.6CM.
P2 = 0.0 Z2 = 2.21 BMX2 = 90.00 BMN2 = -90.00 R2 = SPHERE OF RADIUS 0.3195 CM

THETA<DEG>	FOR<PHI=0>	FOR<PHI=90>
0	0.3035762581D 00	0.3035762581D 00
5	0.3003537909D 00	0.3034772163D 00
10	0.2908064324D 00	0.3032241824D 00
15	0.2752988026D 00	0.3029388182D 00
20	0.2544498749D 00	0.3027905449D 00
25	0.2291380866D 00	0.3029532423D 00
30	0.2004887500D 00	0.3035568553D 00
35	0.1698321377D 00	0.3046435813D 00
40	0.1386266253D 00	0.3061380278D 00
45	0.1083505796D 00	0.3078388796D 00
50	0.8037660347D-01	0.3094363275D 00
55	0.5584997017D-01	0.3105550895D 00
60	0.3559336084D-01	0.3108179206D 00
65	0.2006082912D-01	0.3099200205D 00
70	0.9349239538D-02	0.3077018505D 00
75	0.3267818479D-02	0.3042074686D 00
80	0.1452195674D-02	0.2997176151D 00
85	0.3500734283D-02	0.2947516793D 00
90	0.9106150369D-02	0.2900366654D 00
95	0.1815577908D-01	0.2864472211D 00
100	0.3077901548D-01	0.2849228529D 00
105	0.4732968871D-01	0.2863708710D 00
110	0.6830323647D-01	0.2915646078D 00
115	0.9420173794D-01	0.3010475115D 00
120	0.1293716128D 00	0.3150545973D 00
125	0.1618461718D 00	0.3334626601D 00
130	0.2032257412D 00	0.3557704753D 00
135	0.2486208764D 00	0.3811693084D 00
140	0.2966706300D 00	0.4085329063D 00
145	0.3456315139D 00	0.4365965354D 00
150	0.3935183261D 00	0.4640288921D 00
155	0.4302694095D 00	0.4895463899D 00
160	0.4779081338D 00	0.5120020796D 00
165	0.5106787315D 00	0.5304379020D 00
170	0.5354451864D 00	0.5441169852D 00
175	0.5502528336D 00	0.5525248083D 00
180	0.5553603362D 00	0.5553603362D 00

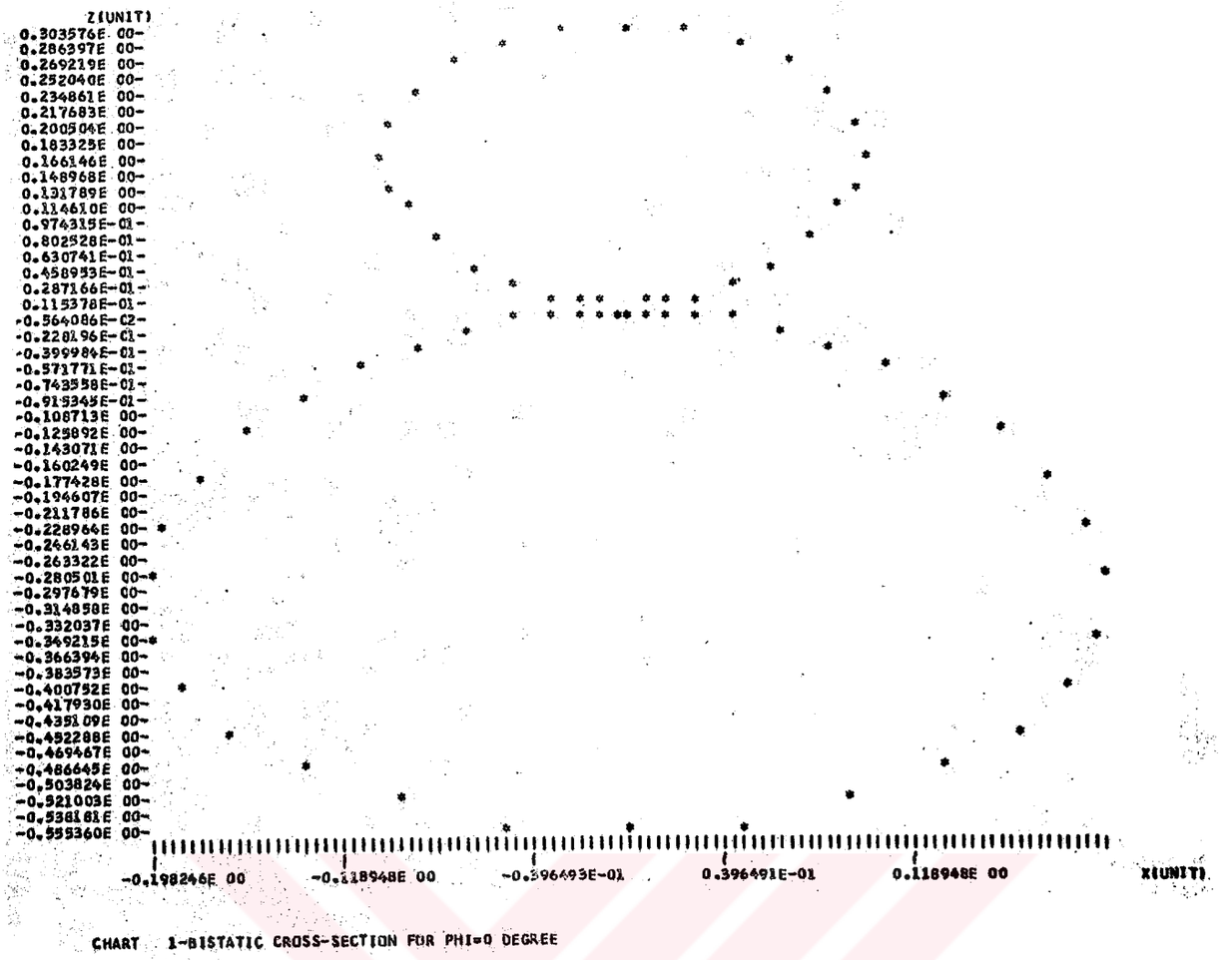

CHART 1-BISTATIC CROSS-SECTION FOR PHI=0 DEGREE

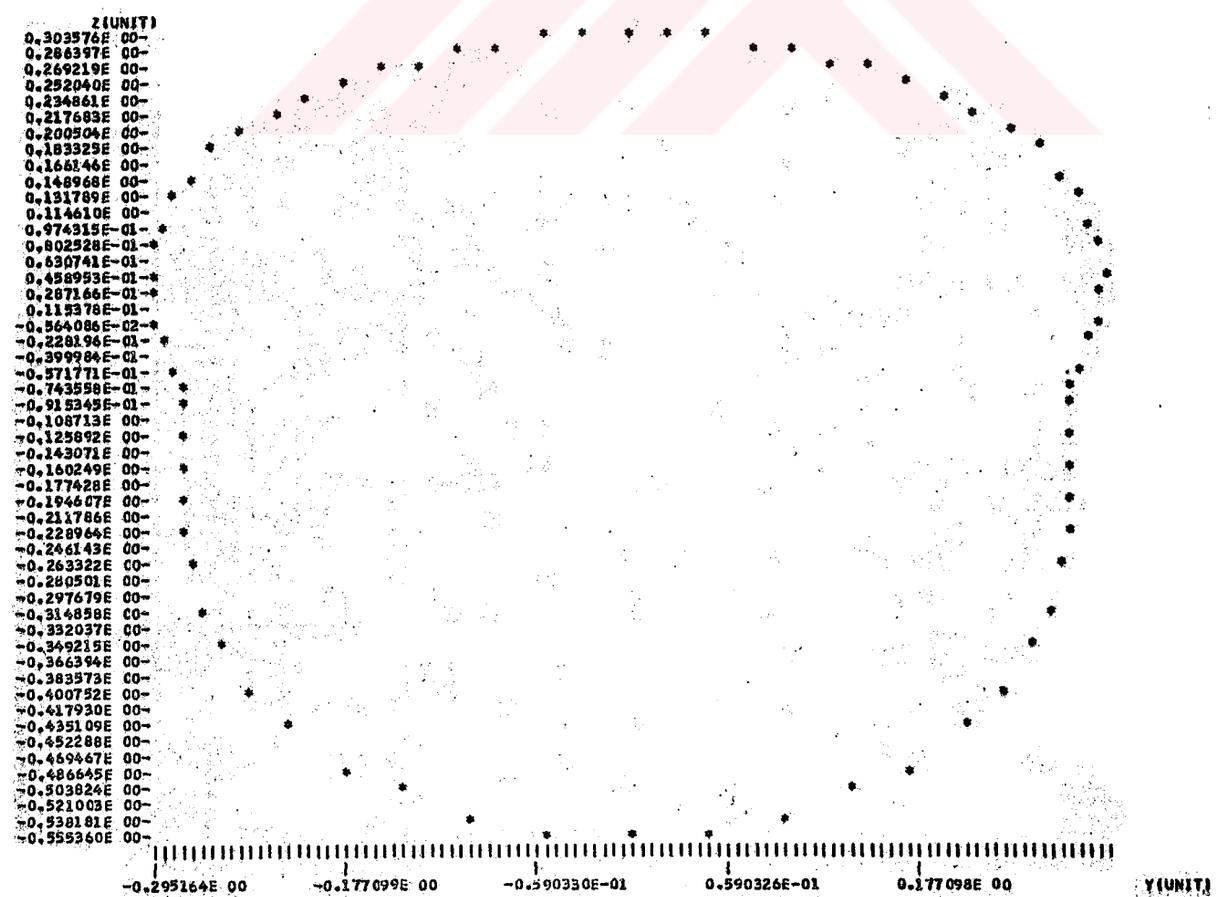

CHART 2-BISTATIC CROSS-SECTION FOR PHI=90 DEGREES