EXTENSION AND UPDATING OF THE COMPUTER SIMULATION OF RANGE RELATIVE DOPPLER PROCESSING AND INVARIANT MAPPING FOR MM WAVE SEEKERS

VOLUME III

SIMULATION OF A DOPPLER BEAM SHAPENING RADAR FOR MM WAVE SEEKERS

Robert J. Polge
Bassem R. Mahafza
Jong G. Kim
College of Engineering
University of Alabama in Huntsville
Huntsville, AL 35899

OCTOBER 1990

Prepared for:
Advanced Sensors Directorate
Research, Development, and Engineering Center

Contract No. DAAH01-87-D-0021

Approved for public release; distribution is unlimited.
DISPOSITION INSTRUCTIONS

DESTROY THIS REPORT WHEN IT IS NO LONGER NEEDED. DO NOT RETURN IT TO THE ORIGINATOR.

DISCLAIMER

THE FINDINGS IN THIS REPORT ARE NOT TO BE CONSTRUED AS AN OFFICIAL DEPARTMENT OF THE ARMY POSITION UNLESS SO DESIGNATED BY OTHER AUTHORIZED DOCUMENTS.

TRADE NAMES

USE OF TRADE NAMES OR MANUFACTURERS IN THIS REPORT DOES NOT CONSTITUTE AN OFFICIAL ENDORSEMENT OR APPROVAL OF THE USE OF SUCH COMMERCIAL HARDWARE OR SOFTWARE.
**REPORT DOCUMENTATION PAGE**

- **REPORT SECURITY CLASSIFICATION**: Unclassified
- **SECURITY CLASSIFICATION AUTHORITY**: Unclassified
- **DECLASSIFICATION/DOWNGRADING SCHEDULE**: Approved for public release; distribution is unlimited.
- **PERFORMING ORGANIZATION REPORT NUMBER(S)**: TR-ER-AS-90-3, Vol III
- **NAME OF PERFORMING ORGANIZATION**: College of Engineering
- **ADDRESS (City, State, and ZIP Code)**: University of Alabama in Huntsville
  Huntsville, AL 35899
- **NAME OF FUNDING/SPONSORING ORGANIZATION**: RD&Z Center
- **ADDRESS (City, State, and ZIP Code)**: University of Alabama in Huntsville
  Huntsville, AL 35899
- **DATE OF REPORT**: October 1990
- **PAGE COUNT**: 78
- **TITLE**: Extension and updating of the computer simulation of range relative Doppler processing and invariant mapping for M wave seekers, Volume III, simulation of a Doppler beam sharpening radar for M wave seekers
- **PERSONAL AUTHOR(S)**: Robert J. Polge, Bassam R. Mahafza, Jong G. Kim
- **TYPE OF REPORT**: Interim
- **DATE COVERED**: FROM 6/3/88 TO 2/28/88
- **SOURCE OF FUNDING NUMBERS**: DAAB01-87-D-0021
- **ABSTRACT**: This report extends the application of Range Relative Doppler Processing "RRDP" and of the Invariant Mapping Technique to beam sharpening radars, where the goal is to map a large area on the ground. In this application the antenna follows a horizontal path while scanning a wide azimuth interval. Two programs are developed: DOPPF.FOR and DOPXY.FOR. The first detects targets within the sequence of footprints. The second program uses invariant mapping to generate a file which contains reflectivity data versus absolute coordinates for a selectable x-y window. An example illustrates the detection of 40 randomly distributed scatterers. The performance is quite good, except that the azimuth resolution is poor for footprints at very low azimuth angles. All the results are in complete agreement with the theory.

**DISTRIBUTION/AVAILABILITY OF ABSTRACT**

- **SAME AS REPORT**: Unclassified

**SUBJECT TERMS**: Range Relative Doppler Processing (RRDP), M wave seekers, Invariant Mapping

**ABSTRACT SECURITY CLASSIFICATION**: Unclassified

**NAME OF RESPONSIBLE INDIVIDUAL**: A. H. Green
  205-876-1728

---

**UNCLASSIFIED**

**SECURITY CLASSIFICATION OF THIS PAGE**: Unclassified
PREFACE


The main objective of Volume I is to extend and update the MM wave computer simulation developed for Contract No. DAAH01–87–D–0021, D.O. 18, documented in the final report dated February 1988, and entitled "Increasing Azimuth and Elevation Resolution of MM Wave Seeker Systems Using Coherent or Noncoherent Range Relative Doppler Processing (RRDP) with Constant or Linear Frequency Modulation and Invariant Mapping".

Volume II is a User Manual for the computer simulation documented in Volume I. With this manual, MICOM personnel should be able to: (1) install the Fortran software on an IBM PC/compatible, (2) duplicate the results presented in this report, and (3) run the simulation for different clutter maps and targets.

The main objective of Volume III is to develop Doppler Beam Sharpening for MM wave seekers. In this application the geometry is significantly different. That is, the seeker follows a straight horizontal trajectory while the antenna performs a forward near circular scan perpendicular to the trajectory.

Robert J. Polge

Huntsville, Alabama
March 1989
ACKNOWLEDGEMENT

The Principal Investigator wishes to take this opportunity to acknowledge the cooperation and guidance of Mr. Jim Mullins and Mr. A. H. Green both of the U. S. Army Missile Command. Special Thanks are due to Mrs. Janie Parent for her efforts in typing the manuscript.
Table of Contents

1. Introduction and Background .................................................. 1
2. Geometry and System Parameters for a Single Footprint .................. 3
3. Reflectivity versus Doppler and Range For a Single Footprint: DOPFFS .... 6
4. Reflectivity versus Absolute x–y Coordinates for a Single Footprint: DOPXYS 13
5. Sequence of Footprints for Scanning ....................................... 16
6. Predicted Resolution: azimuth and range ................................... 18
7. Overview of the Simulation of a Sequence of Footprints: DOPFP ............ 21
8. Overview of the Simulation with Invariant Mapping: DOPXY ............... 24
9. Inputs and Outputs for DOPFP and DOPXY .................................. 25
10. Demonstration of the DOPFP Simulation .................................... 27
11. Demonstration of the DOPXY Simulation .................................... 31
12. Summary, Conclusions and Recommendations for Future Work ............ 35

Appendices

Appendix A — Relevant Footprint Information for m = {0,d, h}. .................. 42
Appendix B — Relevant Information for a Scatterer C. .......................... 46
Appendix C — Listings for DOPFP and Associated Subprograms .................. 48
Appendix D — Listings of DOPXY and Associated Subprograms .................. 64

Accession For

| NTIS GRA&I |  |
| DTIC TAB |  |
| Unannounced |  |
| Justification |  |

By Distribution/ Availability Codes

<table>
<thead>
<tr>
<th>Dist</th>
<th>Avail and/or Special</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

v/(vi blank)
<table>
<thead>
<tr>
<th>Figure</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Geometry for a single footprint</td>
<td>4</td>
</tr>
<tr>
<td>3.1</td>
<td>Simulation of a single footprint: DOPFPS</td>
<td>8</td>
</tr>
<tr>
<td>4.1</td>
<td>Simulation of a single footprint: DOPXYS</td>
<td>14</td>
</tr>
<tr>
<td>5.1</td>
<td>Sequence of footprint centers for four cycles</td>
<td>19</td>
</tr>
<tr>
<td>5.2</td>
<td>Illustration of footprint overlaps</td>
<td>20</td>
</tr>
<tr>
<td>6.1</td>
<td>Azimuth resolution versus footprint azimuth</td>
<td>22</td>
</tr>
<tr>
<td>7.1</td>
<td>Simulation of a sequence of footprints: DOPFPG.GRD contains reflectivity data versus relative doppler and range</td>
<td>23</td>
</tr>
<tr>
<td>8.1</td>
<td>Simulation of a sequence of footprints: DOPXYG.GRD contains reflectivity data versus absolute x-y coordinates</td>
<td>26</td>
</tr>
<tr>
<td>10.1</td>
<td>3-D reflectivity plots versus relative doppler and range: (a) 16th footprint, (b) 61st footprint</td>
<td>29</td>
</tr>
<tr>
<td>10.2</td>
<td>Contour maps of reflectivity versus relative doppler and range: (a) 16th footprint, (b) 61st footprint</td>
<td>30</td>
</tr>
<tr>
<td>11.1</td>
<td>3-D reflectivity versus absolute x-y: (a) 16th footprint, (b) 61st footprint</td>
<td>32</td>
</tr>
<tr>
<td>11.2</td>
<td>Reflectivity contour maps versus x-y coordinates: (a) 16th footprint, (b) 61st footprint</td>
<td>33</td>
</tr>
<tr>
<td>11.3</td>
<td>Reflectivity versus absolute x-y: (a) 3-D plot, (b) contour map</td>
<td>35</td>
</tr>
<tr>
<td>11.4</td>
<td>Contour plot of the reflectivity versus absolute x-y for the 16th footprint</td>
<td>36</td>
</tr>
<tr>
<td>11.5a</td>
<td>Actual x-y coordinates for the 40 scatterers</td>
<td>37</td>
</tr>
<tr>
<td>11.5b</td>
<td>Detected x-y coordinates for the 40 scatterers</td>
<td>38</td>
</tr>
</tbody>
</table>
### LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>List of system parameters</td>
<td>7</td>
</tr>
<tr>
<td>3.1</td>
<td>Clutter data file CLMPI.INP for 40 scatterers</td>
<td>10</td>
</tr>
<tr>
<td>5.1</td>
<td>Geometry parameters for scanning: GEODOPI.INP</td>
<td>17</td>
</tr>
<tr>
<td>9.1</td>
<td>Final list of detected scatterers</td>
<td>28</td>
</tr>
</tbody>
</table>
1. Introduction and Background

Two new digital processing techniques [1–4] were developed to extend applications of SAR processing to nonlinear trajectories. They are entitled, "Range Relative Doppler Processing (RRDP)", and "Invariant Mapping".

Volume I [3] presents the most recent update for these techniques. The major contributions are: (1) a more general geometry where the trajectory is not restricted to the vertical plane, (2) an exact simulation for the dish antenna, (3) a more exact signal synthesis that includes the scatterer height, (4) the development of an interactive subroutine which allows online updating of the data file, (5) a much better compensation for antenna gain and range attenuation, (6) a reduction through frequency interpolation of the peak reflectivity variation due to FFT quantization, (7) a new, better, and more efficient invariant mapping algorithm, and (8) online graphics displays for the reflectivity maps (3-D and contours). Two main programs were developed, FRRDP and CRRDP. The first generates the footprint reflectivity map versus azimuth and range. The second program uses our Invariant Mapping Technique to create an absolute x–y reflectivity map. Examples show that the scatterers are detected at the proper locations and that impulse invariance has been achieved over one footprint within ±3%, and over the entire absolute map within ±5.5%.

Volume II [4] is a user manual for the computer simulation documented in Volume I [3]. With this manual MICOM personnel should be able to: (1) install the Fortran software on an IBM PC/compatible, (2) duplicate the results presented in Volume I, and (3) run the simulation for different clutter maps and targets. This manual contains: (1) hardware and software requirements, (2) user overview of the simulation, (3) demonstration diskettes, (4) installation procedure, (5) definition of inputs, (6) display information for FRRDP output, and (7) display information for CRRDP output.
The present report, which is denoted as Volume III, extends the application of RRDP and invariant mapping to Beam Doppler Sharpening Radars, where the goal is to map a wide area on the ground. In this application, the antenna platform follows a horizontal flight path with constant velocity, while the antenna is scanned to cover the desired azimuth interval.

Two main programs are developed: DOPFP.FOR and DOPXY.FOR. The first program detects scatterers in one or in all footprints. It also generates a file DOPFP.GRD which contains reflectivity data versus relative doppler and range for the selected footprint. Using SURFER [5] on the file DOPFP.GRD produces 3-D and contour plots of the reflectivity versus relative doppler and range. The second program includes invariant mapping and generates a file DOPXYG.GRD which contains reflectivity data versus absolute x–y coordinates for a selectable x–y window. Using SURFER on the file DOPXYG.GRD produces 3-D and contour plots of the reflectivity versus x–y coordinates for a selected x–y window. Again one can run the simulation for one or all footprints.

While many of the subroutines developed in Volume I are also used in DOPFP and DOPXY, several new complex subroutines were developed and the main programs are completely new. Problem areas include: (1) extending the simulation from one footprint to a sequence of footprints, (2) making the program work for any azimuth within the required interval, (3) defining the sequence of footprints for optimum coverage, (4) integrating the information for a sequence of footprints through invariant mapping, (5) developing subroutines to detect peaks and print for each peak: amplitude, footprint index, azimuth, range, and x–y coordinates, (6) developing low resolution displays for the very large area being mapped, and (7) developing high resolution 3-D or contour displays for the selected x–y window, within the limitation of MS–DOS.
The organization is as follows: (1) the geometry and system parameters for a single footprint are discussed, (2) the footprint simulation is extended to arbitrary azimuth, (3) the generation of a sequence of footprints for optimum scanning is presented, (4) the resolution is analyzed as a function of azimuth angle and time dwell, (5) a block diagram shows an overview of DOPFP, (6) another block diagram shows an overview of DOPXY, (7) the inputs and outputs for DOPFP and DOPXY are explained, (8) examples demonstrate DOPFP, and (9) the same examples demonstrate DOPXY. The last section contains conclusions and recommendations for future work.

2. **Geometry and System Parameters for a Single Footprint**

In Range Relative Doppler Processing (RRDP) the information of ground reflectivity is acquired sequentially, one footprint at a time. As in the earlier reports, the dwell interval is 20 ms which yields a frequency resolution of 50 Hz. The contour of the footprint is defined as the 3 dB contour at central dwell time. As usual the footprint is first resolved in range by range gating online where the range cell is the inverse of the system bandwidth. Then each range cell is divided into FFT cells on the base of relative doppler frequency, where zero doppler corresponds to the center of the range cell. For a typical antenna azimuth angle there are about 40 non-interpolated azimuth cells within a range bin. Thus, the azimuth resolution is improved by a factor of about 40. The center and boundary of each FFT cell can be mapped onto the x–y absolute coordinates. It follows that a peak amplitude in an FFT cell corresponds to a peak reflectivity for which the x–y coordinates can be computed. It has been shown that the relative azimuth [1–3] (measured with respect to the ground line of sight GLOS) varies nearly linearly with frequency. Therefore the display of reflectivity versus range and FFT index is very useful. Using Invariant Mapping, one can map each azimuth cell onto the x–y plane so as to obtain a reflectivity map versus absolute coordinates. For more details on RRDP for a single footprint one should refer to Volumes I and II. Figure 2.1 shows the geometry for a
Figure 2.1 Geometry for a single footprint.
single footprint. In our application, the trajectory is an horizontal line in the vertical plane with height \( h = 300 \text{ m} \), and constant velocity \( v = 170 \text{ m/s} \). The elevation angle \( \beta \) of the antenna, which is the complement of the depression angle, is equal to \( 75^\circ \). The azimuth angle \( \alpha \) is within the interval \(-30^\circ \) to \( 30^\circ \).

In Fig. 2.1 the position of the antenna at zero absolute time is

\[
\hat{\mathbf{a}}_0 = (0, 0, h).
\]

Then the position of the antenna at central dwell time \( t_c \) is

\[
\hat{\mathbf{a}}(t_c) = \hat{\mathbf{m}} = (0, y_m = v t_c, h).
\]

Given the azimuth and elevation angles \((\alpha, \beta^*)\) and the central dwell time \( t_c \), one can easily compute the coordinates of the footprint center;

\[
x_q = h \tan \beta^* \sin \alpha \tag{2.3a}
\]

\[
y_q = h \tan \beta^* \cos \alpha + v t_c. \tag{2.3b}
\]

Appendix A contains all the relevant footprint information. This Appendix is very similar to Appendix A of Volume I [3], except that the maximum range is computed differently because of the small depression angle. The same notation is used as in Volume I: \( \Delta r \) is the range bin width and the footprint center is given by,

\[
\hat{\mathbf{q}}^* = \hat{\mathbf{q}}(t_c) : \{x_q, y_q, 0\}. \tag{2.4}
\]

The relevant information for the \( i \)th scatterer \( \hat{\mathbf{C}}_i \) is in Appendix B. The absolute coordinates of the scatterer \( \hat{\mathbf{C}}_i \) and of its ground projection \( \hat{\mathbf{C}}_{gi} \) are:

\[
\hat{\mathbf{C}}_i = \{x_i, y_i, h_i\} \tag{2.5a}
\]

\[
\hat{\mathbf{C}}_{gi} = \{x_i, y_i, 0\}. \tag{2.5b}
\]

As in Volume I, the relative azimuth and elevation for \( \hat{\mathbf{C}}_i \), are denoted \( (\mu_i, \epsilon_i) \) and they are computed with respect to \( \hat{\mathbf{C}}_{gi} \). More precisely, the azimuth and elevation, with respect to \( \hat{\mathbf{m}} \), of a scatterer \( \hat{\mathbf{C}}_i \) in the \( j \)th range bin are,

\[
\alpha_i = \alpha + \mu_i \tag{2.6a}
\]

\[
\beta_i = \beta^*_j + \epsilon_i \tag{2.6b}
\]

where \( \beta^*_j \) is the elevation angle for the center of the \( j \)th range cell.
The position of the antenna within the observation interval $D_{ob}$, is given by

$$\hat{a}(t_a) - \hat{a}(t + t_c) = \{0, v t_a = v(t+t_c), b\}$$  \hspace{1cm} (2.7a)

$$-\frac{D_{ob}}{2} \leq t \leq \frac{D_{ob}}{2}$$  \hspace{1cm} (2.7b)

where $t_a$ denotes absolute time, and $t$ denotes relative time.

As explained in Volume I the gain pattern for a circular dish of diameter $D$ is $[6, 7]$

$$G(sD/\lambda) = 2A \frac{J_1(\pi sD/\lambda)}{\pi sD/\lambda}$$  \hspace{1cm} (2.8)

where $J_1$ is a Bessel function of the first kind of the first order, and $A$ is the area of the circular dish aperture. The normalized dish pattern is in Fig. 3.1 of Volume I.

The list of system parameters is in Table 2.1. These parameters are different significantly for those of Volume I in three areas: (1) lower height, (2) lower depression angle, and (3) horizontal velocity.

3. Reflectivity Versus Relative Doppler and Range for a Single Footprint: DOPFPS

A main program DOPFP.FOR has been developed to detect scatterers in one or in all footprints. This program also generates a file DOPFP.GRD which contains reflectivity data versus relative doppler and range for the selected footprint. Using SURFER [5] on the file DOPFP.GRD produces 3-D and contour plots of the reflectivity versus relative doppler and range. Since the program DOPFP.FOR is quite complex, this section explains only a simplified program operating on a single footprint. This fictitious single footprint program is denoted as DOPFPS.FOR.

The block diagram for DOPFPS.FOR is in Fig. 3.1. This program differs from RRDPF.FOR of Volume I as follows: (1) the simulation is extended to negative azimuth angles, (2) the antenna gain compensation has been improved, (3) the clutter map is now defined for an arbitrary number of footprints, yet the signal simulation for a range cell involves only the relevant scatterers, (4) subroutines were developed to detect and print the peaks and the associated information after processing, and (5) the structure of the
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TIMDWL</td>
<td>0.02000000</td>
<td>observation interval</td>
</tr>
<tr>
<td>DELRAN</td>
<td>1.00000000</td>
<td>range bin</td>
</tr>
<tr>
<td>XMMNPR</td>
<td>20.0000000</td>
<td>threshold to print XM(KFF)</td>
</tr>
<tr>
<td>WPAR(1)</td>
<td>0.75000000</td>
<td>rectangular grid increment @XY</td>
</tr>
<tr>
<td>WPAR(2)</td>
<td>1.00000000</td>
<td>mapping threshold in FP or XY</td>
</tr>
<tr>
<td>WPAR(3)</td>
<td>287.5000000</td>
<td>window(xmin-abs.) @XY</td>
</tr>
<tr>
<td>WPAR(4)</td>
<td>1276.2500000</td>
<td>window(ymin-abs.) @XY</td>
</tr>
<tr>
<td>WPAR(5)</td>
<td>357.2500000</td>
<td>window(xmax-abs.) @XY</td>
</tr>
<tr>
<td>WPAR(6)</td>
<td>1346.0000000</td>
<td>window(ymax-abs.) @XY</td>
</tr>
<tr>
<td>WPAR(7)</td>
<td>102.0000000</td>
<td>window(xsize) @XY</td>
</tr>
<tr>
<td>WPAR(8)</td>
<td>187.0000000</td>
<td>window(ysize) @XY</td>
</tr>
<tr>
<td>XIFXMT</td>
<td>0.00000000</td>
<td>transmitter phase</td>
</tr>
<tr>
<td>DETIPR</td>
<td>4.00000000</td>
<td>FFT window</td>
</tr>
<tr>
<td>SELWIN</td>
<td>2.00000000</td>
<td>to print PRXYSC-1 @FP</td>
</tr>
<tr>
<td>SELDIS</td>
<td>1.00000000</td>
<td>define dish simulation</td>
</tr>
<tr>
<td>WINIPR</td>
<td>300.0000000</td>
<td>(0.,1.) window (WPAR,center FP)</td>
</tr>
<tr>
<td>HEIPF</td>
<td>-300.0000000</td>
<td>ref-height for RANCOF(init-value)</td>
</tr>
<tr>
<td>BETREF</td>
<td>-999.0000000</td>
<td>ref-beta for RANCOF (init-value)</td>
</tr>
<tr>
<td>SHOSCA</td>
<td>-999.0000000</td>
<td>scatterer-data in FP @FP</td>
</tr>
<tr>
<td>BUGLEV</td>
<td>140.0000000</td>
<td>(0. for noPRT);((1.,2.) to debug)</td>
</tr>
<tr>
<td>FPXMB</td>
<td>1.00000000</td>
<td>bound on # footprint processed</td>
</tr>
<tr>
<td>SURFIN</td>
<td>1.00000000</td>
<td>1. to write the file DOPFGP.GRD</td>
</tr>
</tbody>
</table>
1 Initialization

2 Enter Scatterer Information : CLUMAP

3 Enter Footprint Information

4 Compute Relevant Footprint Parameters : BCEDOP

5 Information on Smoothed Scatterers for all range cells in Footprint : CLUFPT

6 Specify Range Cell

7 (Loop KTHSCA)
   Transfer Scatterer Information from SMOMAP to WTSMO.
   Compute Antenna Gain, Synthesize Signal in JRANth Range Cell : Array (XD,XQ)

8 Electronic Processing

9 Scale and Store Reflectivity in SYNMAP and Prepare for Detection

10 Detect Scatterers within a Footprint

Figure 3.1 Simulation of a single footprint : DOPFPS
program has been greatly improved. The listings of DOPFP and of the new or modified associated subroutines are in Appendix C. For the other subroutines, refer to Volume I.

In the initialization block, the system parameters are entered interactively by reading, displaying on the screen, and updating the system input parameter file SYPAI.INP (see Table 2.1). In SYPAI.INP the parameters that are only for debugging are specified by the symbol "@@". Parameters only used in DOFPF or only in DOPXY are indicated respectively by, "@FP" and "@XY". The subroutines WINPUX is used to enter the parameters interactively. WINPUX is the same as WINPUT, which was explained in Volume I, except6 that the data is displayed on the screen before and after the changes. The arrays that are independent of the footprint, are also computed. These include:

\{WTIME ⇔ relative sampling time\}
\{WFREQ ⇔ relative doppler frequency\}
\{WIND ⇔ Kaiser window for FFT\}
\{LNPPP ⇔ translated FFT index\}.

The purpose of \{LNPPP\} is to use the index KNPPP which varies from 1 to NFFT from minimum to maximum doppler, rather than the FFT index KFF which does not correspond to a continuous doppler variation. The correspondence is: \{KNPPP = 1 ⇔ KFF−1 = (NFFT/2 + 1)\}, where KFF is modulo NFFT.

The scatterer information is entered in block 2 by reading the clutter data file CLMPI.INP. Table 3.1 shows a clutter file for 40 scatterers. The format is as follows: (1) number of scatterers, and (2) one row of data for each scatterer: {amplitude, phase, x-coordinate, y-coordinate, z-coordinate, scatterer index}. The scatterer data is read, and displayed on the screen.

In block 3 the footprint information is entered by reading one row of the file FPSQI.INP. The eight parameters defining the footprint are:
Table 3.1. Clutter data file CLMPI.INP for 40 scatterers

<p>| | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>40.000</td>
<td>.000</td>
<td>-447.409</td>
<td>1083.000</td>
<td>.000</td>
<td>1.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>-511.018</td>
<td>1204.927</td>
<td>.000</td>
<td>2.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>-460.390</td>
<td>1121.403</td>
<td>.000</td>
<td>3.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>-530.195</td>
<td>1228.854</td>
<td>.000</td>
<td>4.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>-450.643</td>
<td>1092.814</td>
<td>.000</td>
<td>5.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>-349.354</td>
<td>1194.242</td>
<td>.000</td>
<td>6.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>-320.981</td>
<td>1214.717</td>
<td>.000</td>
<td>7.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>-355.527</td>
<td>1276.781</td>
<td>.000</td>
<td>8.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>-372.186</td>
<td>1320.675</td>
<td>.000</td>
<td>9.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>-347.018</td>
<td>1267.020</td>
<td>.000</td>
<td>10.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>-320.981</td>
<td>1214.717</td>
<td>.000</td>
<td>11.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>-411.497</td>
<td>1324.242</td>
<td>.000</td>
<td>12.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>-320.981</td>
<td>1214.717</td>
<td>.000</td>
<td>13.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>-222.056</td>
<td>1300.950</td>
<td>.000</td>
<td>14.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>-229.101</td>
<td>1346.910</td>
<td>.000</td>
<td>15.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>-61.271</td>
<td>1246.853</td>
<td>.000</td>
<td>16.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>-62.473</td>
<td>1266.286</td>
<td>.000</td>
<td>17.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>-61.264</td>
<td>1430.723</td>
<td>.000</td>
<td>18.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>-43.413</td>
<td>1369.523</td>
<td>.000</td>
<td>19.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>-50.957</td>
<td>1262.079</td>
<td>.000</td>
<td>20.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>77.345</td>
<td>1411.096</td>
<td>.000</td>
<td>21.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>72.384</td>
<td>1418.710</td>
<td>.000</td>
<td>22.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>45.316</td>
<td>1275.881</td>
<td>.000</td>
<td>23.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>70.514</td>
<td>1300.239</td>
<td>.000</td>
<td>24.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>62.073</td>
<td>1446.496</td>
<td>.000</td>
<td>25.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>256.499</td>
<td>1434.587</td>
<td>.000</td>
<td>26.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>193.773</td>
<td>1324.643</td>
<td>.000</td>
<td>27.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>190.213</td>
<td>1270.189</td>
<td>.000</td>
<td>28.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>192.693</td>
<td>1307.827</td>
<td>.000</td>
<td>29.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>215.186</td>
<td>1324.218</td>
<td>.000</td>
<td>30.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>331.811</td>
<td>1277.989</td>
<td>.000</td>
<td>31.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>353.076</td>
<td>1333.173</td>
<td>.000</td>
<td>32.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>318.932</td>
<td>1292.184</td>
<td>.000</td>
<td>33.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>367.597</td>
<td>1349.559</td>
<td>.000</td>
<td>34.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>314.725</td>
<td>1250.479</td>
<td>.000</td>
<td>35.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>514.790</td>
<td>1283.882</td>
<td>.000</td>
<td>36.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>499.218</td>
<td>1298.021</td>
<td>.000</td>
<td>37.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>505.042</td>
<td>1274.220</td>
<td>.000</td>
<td>38.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>502.415</td>
<td>1350.866</td>
<td>.000</td>
<td>39.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>.000</td>
<td>450.037</td>
<td>1250.946</td>
<td>.000</td>
<td>40.000</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
VELOM $\Rightarrow$ Missile velocity

ALFM $\Rightarrow$ Azimuth angle for $\hat{q}(t_c)$

BETM $\Rightarrow$ Elevation angle for $\hat{q}(t_c)$

XM $\Rightarrow$ $x$–coordinate for missile at central time

YM $\Rightarrow$ $y$–coordinate for missile at central time

ZM $\Rightarrow$ $z$–coordinate for missile at central time $= h$

XFP $\Rightarrow$ $x$–coordinate for footprint center

YFP $\Rightarrow$ $y$–coordinate for footprint center

Block 4 is the computation of all the relevant footprint parameters. This is accomplished by subroutine BCEDOP, which implements the formulas given in Appendix A.

In block 5 the information for the scatterers that fall within the footprint is computed using subroutine CLUFPT. As well known, a point scatterer affects three consecutive range cells because of the finite bandwidth of the system. This effect which is denoted as range smoothing is simulated using subroutine RANGSMOO, which splits each scatterer into three equivalent scatterers. All the necessary scatterer information is passed to the main program through the arrays SMOMAP, WADRLO, WADRUP, and WADSMO.

The heart of DOPFPS in "loop JRAN" which corresponds to "DO 35" in the program. In this loop the range cells are processed sequentially: JRAN = 1, JRANMX.

In block 6 a check for the presence of smoothed scatterers within the current JRANth cell is performed. If WADRLO(JRAN) = 0 there are no scatterers and "loop JRAN" has been completed.

Block 7 represents "loop KTHSCA" which corresponds to "DO 30" in the program. Its purpose is to synthesize the signal \( \{X_D(KDQ), X_Q(KDQ), KDQ = 1, NFFT\} \) in the
JRANth range cell. The contributing scatterers are defined by \( \{ \text{WADRLO}(\text{JRAN}) < \text{KTHSCA} < \text{WADRUP}(\text{JRAN}) \} \). The information for the KTHSCA scatterer is transferred from SMOMAP into the array WTSMO. The antenna gain and the contribution to the signal due to KTHSCA scatterer are computed using subroutines SCAGAN and TSIGNAL.

The electronic processing of the JRANth cell is in block 8. It consists of the following steps: (1) deramping, (2) windowing, and (3) fast Fourier transformation (FFT). The subroutines DERAMP, WINDOWS, and FFT are used for this purpose. Note that the FFT is performed on the extended arrays \( \{ \text{XD}(k), \text{XQ}(k), k = 1, 2*\text{NFFT} \} \) which are padded with \( \text{NFFT} \) zeros. This corresponds to the implementation of frequency interpolation which was discussed in section 6 of Volume I.

Block 9 is concerned with the scaling of the FFT cells, the synthesis of the reflectivity array SYNMAP, and the preparation for detection. First, the array WAGNCL which contains the antenna gains at the center of the FFT cells is computed, ("DO 39" in the program). Then, both the scaling and the computation of SYNMAP are performed in "DO 41". There are two important subroutines: SURFMP and DETFPT, which are listed in Appendix C. The subroutine SURFMP uses the Common /SURFP/, and has three entry points. The first entry point, in block 1, is for initialization of the array SYNMAP. The second entry point, in block 9, is for storing reflectivity data in SYNMAP. The subroutine DETFPT includes the Common /DETECT/ and has also three entry points. The first entry point is used in block 9 to store the peak values above the specified threshold within the current range cell.

Block 10 is the last step in the processing of a single footprint. Subroutine SURFMP is called in the third entry mode to generate DODFPG.GRD from SYNMAP in the format specified by SURFER [5]. DETFPT in the second entry mode selects the peaks within the entire footprint.
4. Reflectivity Versus Absolute x–y Coordinates for a Single Footprint: DOPXYS

A main program DOPXY.FOR has been developed to detect scatterers in one or all footprints. This program includes invariant mapping and generates a file DOPXYG.GRD which contain reflectivity data versus absolute x–y coordinates for a selectable x–y window. Using SURFER on the file DOPXYG.GRD produces 3–D and contour plots. Again one can run the simulation for one or all footprints. In the case of overlapping footprints, the reflectivity information is accumulated and smoothed through invariant mapping. The listings of DOPXY, MAPRGD, MAPAZD are in Appendix D.

Since the program DOPXY.FOR is quite complex, this section explains only a simplified program operating on a single footprint. This fictitious single footprint program is denoted DOPXYS.FOR.

The block diagram for DOPXYS.FOR is in Fig. 4.1. The first eight blocks of Fig. 4.1 are the same as in DOPFPS.FOR and will not be discussed further. Both DOPFPS and DOPXYS use the same input files.

In DOPXY the x–y window can be selected in two ways: (1) manual specification, and (2) automatic specification. In the first case the window parameters in the input file SYPAI.INP are set as follows:

WINIWR = 0.

WPAR(3) = x–for minimum corner
WPAR(4) = y–for minimum corner
WPAR(5) is computed
WPAR(6) is computed
WPAR(7) = size of window along x–coordinate
WPAR(8) = size of window along y–coordinate.

In the second case, WINIWR is set to equal (1.), and the window of size (WPAR(7), WPAR(8)) is automatically centered on the center of the selected footprint. The window
1 Initialization

2 Enter Scatterer Information : CLUMAP

3 Enter Footprint Information

4 Compute Relevant Footprint Parameters : BCEDOP

5 Information on Smoothed Scatterers for all range cells in Footprint : CLUFPT

6 Specify Range Cell

7 (Loop KTHSCA)
Transfer Scatterer Information from SMOMAP to WTSMO.
Compute Antenna Gain, Synthesize Signal in JRANth Range Cell : Array (XD,XQ)

8 Electronic Processing

9 Compute Scaled Reflectivity XM and Apexes of the Azimuth Cells

10 Invariant Mapping : Accumulate Numerator and Denominator of (4.1)

11 Complete Invariant Mapping and Generate DOPXYG.GRD in Format [5]

Figure 4.1 Simulation of a single footprint : DOPXYS
is rounded up to integer values of \( \text{DELMAP} = \text{WPAR}(1) \) using the subroutine \text{MAPWIN} with entry 1. The automatic window is set by \text{MAPWIN} with entry 4. The subroutine \text{MAPWIN} is listed in Appendix D.

Block 9 is concerned with scaling of the FFT cells and computation of trapezoid apexes coordinates. First, the array \text{WAGNCL} which contains the antenna gains at the center of the FFT cells is computed, ("DO 39") in the program). Then, scaling for range attenuation is performed in "DO 41". Also, subroutine \text{APEXTRAP}, which was discussed in Volume I, computes the apex coordinates for the trapezoidal FFT cells in the current range cell.

As explained in Volume I the invariant mapping algorithm implements the following formula

\[
\text{RFR}(k) = \frac{\sum_{m} \text{OVL}(m,k) \text{RFT}(m)}{\sum_{m} \text{OVL}(m,k)}
\]  

(4.1)

where: \( m \) defines all the trapezoids which overlap the \( k \)th rectangle, and

\( \text{TRP}(m) \Leftrightarrow \text{mth trapezoidal cell} \)

\( \text{RFT}(m) \Leftrightarrow \text{reflectivity of the mth trapezoidal cell} \)

\( \text{REC}(k) \Leftrightarrow \text{kth absolute rectangular cell} \)

\( \text{RFR}(k) \Leftrightarrow \text{estimated reflectivity of the kth rectangular cell} \)

\( \text{OVL}(m,k) \Leftrightarrow \text{overlap area between mth trapezoid and kth rectangle.} \)

Block 10 uses subroutine \text{MAPRGD} to compute the array \text{SYNMAP} and \text{SUMARA} which represent respectively, the numerator and denominator of (4.1). The accumulation in \text{SYNMAP} and \text{SUMARA} is performed by subroutine \text{MAPAZD} which is called by \text{MAPRGD}. Communication between \text{DOPXYS} and \text{MAPAZD} is through common /\text{GRDMAP}/.
Block 11 corresponds to the two "DO 60" in the program. First, the reflectivity map SYNMAP is computed according to (4.1). Then, the reflectivity map versus absolute x and y coordinates is written on the file DOPXYG.GRD using the format recommended by [5].

5. Sequence of Footprints for Scanning

In our application the antenna platform moves at a constant speed along a horizontal path as explained in section 2. The absolute coordinates of the platform at absolute time \( t_a \) are,

\[
a(t_a) = \{0, v \cdot t_a, h\}.
\]  

(5.1)

The elevation angle \( \beta \) is constant and equal to 75°. For each azimuth angle \(-30° < \alpha < 30°\) one can compute the coordinates of the footprint center \((x_q, y_q)\) as explained in Appendix A.

Scanning can be defined as selecting a sequence of azimuth values \( \{\alpha\} \) which will provide a complete ground coverage including some overlap. Subroutine FPIDOP, which is listed in Appendix D, is used to compute the sequence of angles \( \alpha \), and to write the footprint information for each \( \alpha \) value. The assumptions are as follows: (1) the initial value of \( \alpha \) is about \(-30°\), (2) each dwell interval for every footprint is 20 msec, (3) the time interval between consecutive footprints is greater than 20 msec to allow step scanning, (4) mapping is performed on increasing \( \alpha \) from about \(-30°\) to \(+30°\) with fast return from \(-30°\) to \(+30°\), and (5) there should be sufficient overlap between consecutive footprints and scans in order to guarantee detection.

The inputs of FPIDOP are read in the file GEODOPI.INP. The parameters used in our examples are in Table 5.1.

The subroutine FPIDOP writes the file FPSQI.INP which contains the footprint information. More precisely, one row in FPSQI.INP defines for each dwell time: (1) the velocity vector, (2) the antenna position, and (3) the center of the footprint. There are
Table 5.1. Geometry parameters for scanning: GEODOPI.INP.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VELM</td>
<td>170.0000000</td>
<td>Seeker velocity</td>
</tr>
<tr>
<td>ALFM</td>
<td>0.0000000</td>
<td>Seeker azimuth angle</td>
</tr>
<tr>
<td>BETM</td>
<td>1.5707960</td>
<td>Seeker elevation angle</td>
</tr>
<tr>
<td>WBRUN(1)</td>
<td>0.0000000</td>
<td>X-beginning of run</td>
</tr>
<tr>
<td>WBRUN(2)</td>
<td>300.0000000</td>
<td>Y-beginning of run</td>
</tr>
<tr>
<td>WBRUN(3)</td>
<td>1.3089970</td>
<td>Z-beginning of run</td>
</tr>
<tr>
<td>BETFP</td>
<td>1.0472000</td>
<td>Footprint elevation angle</td>
</tr>
<tr>
<td>ALFCY</td>
<td>1.4300000</td>
<td>Cycle azimuth coverage</td>
</tr>
<tr>
<td>OVRLAP</td>
<td>4.0000000</td>
<td>Footprint overlap (1,1.5)</td>
</tr>
<tr>
<td>CYNB</td>
<td>0.0100000</td>
<td># cycles in run</td>
</tr>
<tr>
<td>TIMGAP</td>
<td>-0.5235988</td>
<td>Time gap between cycles (sec)</td>
</tr>
<tr>
<td>ALFSTR</td>
<td>Initial ALFFP for a run</td>
<td></td>
</tr>
</tbody>
</table>
nine components per row: VELOM, ALFM, BETM, XM, YM, ZM, XFP, YFP, and ALFFP. The last component ALFFP is used only for debugging. The main formulas in FPIDOP are

\[
\begin{align*}
\text{NBCY} &= \# \text{ of cycles} = \text{input} \\
\text{ANGCY} &= \text{angle per cycle from } \hat{m} \\
&= \cos^{-1}(\sin^2(\text{BETFP}) + \cos(\text{ALFCY}) + \cos^2(\text{BETFP})) \\
\text{NBFPCY} &= \text{footprints per cycle} = 2 \int \left[ \frac{\text{ANGCY} \times \text{OURLAP} \times 0.5 + \epsilon}{\theta_{3dB}} \right] \\
\text{DLALF} &= \alpha \text{ increment in FP sequence} \\
\text{ALFCY} &= \frac{\text{ALFCY}}{\text{NBFPCY}} \\
\text{FPLNG} &= \text{FP length} = h \left[ \tan(\text{BETFP} + \theta_H) - \tan(\text{BETFP} - \theta_H) \right] \\
\text{PERCY} &= \text{period per cycle} = \frac{\text{FPLNG}}{V_y \times \text{OVRLAP}} \\
\text{DLTISQ} &= \text{time increment in FP sequence} = \frac{\text{PERCY} - \text{TIMGAP}}{\text{NBFPCY}}
\end{align*}
\]

Figure 5.1 shows the sequence of footprint centers for the first four cycles. The unit for \(x\) and \(y\) are meters. Figure 5.2 shows that the footprint overlap in both azimuth and range to guarantee complete coverage.

6. Predicted Resolution: azimuth and range

The relative doppler in the \(j\)th range cell for a horizontal flight path is given by

\[
\text{rdop} = -\sin \beta_j^* \cdot v(\cos(\alpha^* + \mu) - \cos \alpha^*)
\]

(6.1)

where \(\beta_j^*\) is the elevation angle of the \(j\)th range cell and \(\mu\) defines the position of the scatterer with respect to the ground line of sight (GLOS). This formula is in complete
Figure 5.2 Illustration of footprint overlaps.
agreement with the relative doppler computed from the Taylor expansion or by direct dot product. During the sequence of footprints, the dwell time per footprint remains constant and it is equal to 20 ms. It follows that the doppler resolution in any range cell or footprint is equal to $\Delta f = 50$ Hz. However, the azimuth resolution is not constant. The azimuth resolution at specified $\left( \alpha^*, \beta_j^* \right)$ can be computed from (6.1). More precisely, the angular azimuth resolution is given by

$$
\mu = \cos^{-1} \left[ \frac{\lambda \text{ rdop}}{2v \sin \beta_j} + \cos \alpha^* \right] - \alpha^*,
$$

(6.2)

where $\text{rdop} = 50$ Hz. Figure 6.1 shows the variation of the azimuth resolution for the central range cell $\beta_j^* = 75^\circ$, when the absolute value of $\alpha$ varies from $0^\circ$ to $45^\circ$. One observes that the azimuth resolution is 10 times better at $\alpha^* = 30^\circ$ than at $\alpha^* \approx 2.5^\circ$. On the other hand, the azimuth resolution is quite good for $\alpha > 8^\circ$. We will show that all the targets can be detected, even for very small $\alpha^*$.

Range resolution, which is the inverse of the system bandwidth, is assumed constant and equal to 1 m.

7. **Overview of the Simulation of a Sequence of Footprints: DOPFP**

The purpose of DOPFP is to simulate the detection of targets within an azimuth interval from $-30^\circ$ to $+30^\circ$, using RRDP to increase the azimuth resolution. A sequence of footprints is generated to insure a complete ground coverage, including a sufficient azimuth and range overlap. The processing of a single footprint has been explained in section 3 using the fictitious program DOPFPS.

Figure 7.1 gives an overview of DOPFP. Since DOPFP is an extension of DOPFPS only the additional features need to be explained. The initialization block in DOPFP includes an online prompt which is self explanatory:
Figure 6.1. Azimuth resolution versus footprint azimuth.
Figure 7.1 Simulation of a sequence of footprints: DOPFPFG.GRD contains reflectivity data versus relative doppler and range.
"SELECT KFPSGN: (1) -999 = Abort, (2) positive to process and display only footprint #KFPSGN, and (3) negative to process all footprints and display footprint #|KFPSGN|"

"K = ?"

While the ABORT command is instantaneous, the actual selection of one or all footprints is implemented within Loop KFP ("DO 1" in the program), just after computing all the footprint parameters with BCEDOP. For example if "16" is entered, then only the 16th footprint is processed, all the others being skipped. Targets within the 16th footprint are detected. At the same time DOPFP generates the file DOPFPG.GRD which contains reflectivity data versus relative doppler and range, for later display by SURFER. On the other hand if "-16" is entered, then all the footprints are processed. In this case all the ground targets should be detected. Again DOPFPG.GRD contains only the reflectivity information for the 16th footprint.

Target detection is implemented using subroutine DETFPT which was introduced in Section 3. As earlier the first two entries points are used: (1) to store the peak values above the specified threshold within the current range cell (block 9), (2) to select the peaks within a footprint (block 10). The third entry point is used for the final peak selection (block 11).

The generation of the sequence of footprints for the selected scanning scheme is performed by subroutine FPIDOP (block 3) which was discussed in Section 5.

8. Overview of the Simulation with Invariant Mapping: DOPXY

The objective of DOPXY is to generate an x–y reflectivity map for the selected sequence of footprints using both RRDP and invariant mapping. It follows that DOPXY is very similar to DOPFP. The main difference is in the output files: DOPFPG.GRD contains the reflectivity data versus relative doppler and range for the |KFPSGN|th footprint, while DOPXYG.GRD contains the reflectivity data versus absolute x–y
coordinates for the KFPSGNth footprint if KFPSGN>0 or for all footprints if KFPSGN<0. Another difference is that the exact determination of reflectivity peaks using subroutine DETECT is implemented in DOPFP but not in DOPXY, since this would be a duplication. Figure 8.1 gives an overview of DOPXY.

The two programs DOPFP and DOPXY use the same input files and provide complementary information. The main output of DOPFP is the list of detected scatterers while the main output of DOPXY is the file DOPXYG.GRD which contains the reflectivity data versus x−y. Invariant mapping [1–3] performs an incoherent integration on the footprint information. This can be demonstrated by selecting an x−y window and running DOPXY two ways: (1) for a single footprint within the window, and (2) for all the footprints. One can see the smoothing effect due to integration. Also, additional scatterers detected within other footprints appear in the x−y window.

9. Inputs and Outputs for DOPFP and DOPXY

DOPFP and DOPXY use the same input files:

SYPAI.INP ⇔ System parameters file
CLMPI.INP ⇔ Clutter data file
GEODOPI.INP ⇔ Geometry parameter file for scanning.

An example of SYPAI.INP is listed in Table 2.1 and discussed in section 3. An example of CLMPI.INP is in Table 3.1 and is discussed in section 3. The file GEODOPI.INP is discussed in section 5 and an example is in Table 5.1.

The simulation DOPFP has two output files:

DOPFP.PRT ⇔ printout file used for debugging, also contains the final list of detected scatterers.
DOPFP.GRD ⇔ reflectivity data versus relative doppler and range using SURFER format.

25
1 Initialization and Selection of one or all Footprints: KFPSGN

2 Enter Scatterer Information: CLUMAP

3 Generate Footprint Sequence: FPIDOP

4 Specify Footprint: KFP

5 Compute Relevant Footprint Parameters: BCEDOP

   YES  
   KFPSGN > 0 and .NE. KFP

   ND

6 Information on Smoothed Scatterers for all range cells in Footprint: CLUFPT

7 Specify Range Cell

   YES  
   WADRLO(JRAN) = 0

8 (Loop KTHSCA)
   Transfer Scatterer Information from SMOMAP to WTSMO.
   Compute Antenna Gain, Synthesize Signal in JRANth Range Cell: Array (XD,XQ)

9 Electronic Processing

10 Compute Scaled Reflectivity XM and Apexes of the Azimuth Cells

11 Invariant Mapping: Accumulate Numerator and Denominator of (4.1)

12 Generate DOPXYG.GRD in Format [5]

Figure 8.1 Simulation of a sequence of footprints: DOPXYG.GRD contains reflectivity data versus absolute x-y coordinates.
Table 9.1 shows the final list of detected scatterers obtained when using the clutter input file from Table 3.1. The columns of Table 9.1 are as follows: {peak index, debugging index, footprint, range bin, doppler index, peak amplitude, x-coordinate, y-coordinate}.

The simulation DOPXY also has two output files:

DOPXYP.PRT => printout file used for debugging purposes.
DOPXYG.GRD => reflectivity data versus absolute x-y coordinates in SURFER format.

10. Demonstration of the DOPFP Simulation

The performance of DOPFP is demonstrated by detecting (40) forty randomly distributed scatterers on the ground being scanned. The input files for this example are: (1) SYPAI.INP in Table 2.1, (2) CLMPI.INP in Table 3.1, and (3) GEODOPI.INP in Table 5.1. The HP 7475A plotter was used to obtain hard copies.

Running DOPFP with a negative value for KFPSGN will generate the list of detected targets given in Table 9.1. Comparison of the position of the detected peaks to the actual scatterer location in Table 3.1 shows that the detection error is quite small (within ±0.5 m in each coordinate). The detected peaks should theoretically be equal. However, the detected peak amplitude varies within ±9% of the mean. Running DOPFP with a positive value for KFPSGN will generate the list of the scatterers detected within the (KFPSGN)th footprint.

In addition to the list of detected peaks, DOPFP generates the file DOPFPG.GRD which contains the reflectivity data versus relative doppler and range for the |KFPSGN|th footprint. In this file the range index varies from 1 to JRANMX, and the relative doppler index varies from 1 to 128, where 65 corresponds to zero relative doppler (center of range cell).

Figures 10.1a and 10.1b shows a 3-D plot of reflectivity for the 16th and 61th footprints, respectively. Figures 10.2a and 10.2b show the corresponding contour maps.
### Table 9.1. Final list of detected scatterers.

**DETFPT+LSTPIK:** KPIK, KAD, KFP, JRAN, KNPPP, XM, XSCA, YSCA

<table>
<thead>
<tr>
<th>DETFPT-!</th>
<th>1</th>
<th>36</th>
<th>54</th>
<th>165</th>
<th>69</th>
<th>27.4051</th>
<th>75.8491</th>
<th>1411.1520</th>
</tr>
</thead>
<tbody>
<tr>
<td>DETFPT-!</td>
<td>2</td>
<td>23</td>
<td>45</td>
<td>111</td>
<td>70</td>
<td>27.3461</td>
<td>-220.8387</td>
<td>1308.6890</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>3</td>
<td>20</td>
<td>42</td>
<td>132</td>
<td>44</td>
<td>27.2710</td>
<td>-355.7807</td>
<td>1276.7160</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>4</td>
<td>42</td>
<td>61</td>
<td>27</td>
<td>54</td>
<td>27.1940</td>
<td>314.5651</td>
<td>1250.4350</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>5</td>
<td>61</td>
<td>83</td>
<td>71</td>
<td>71</td>
<td>27.1860</td>
<td>-60.5324</td>
<td>1430.6290</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>6</td>
<td>30</td>
<td>50</td>
<td>35</td>
<td>65</td>
<td>26.8870</td>
<td>-49.2655</td>
<td>1262.1770</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>7</td>
<td>54</td>
<td>67</td>
<td>96</td>
<td>86</td>
<td>26.8115</td>
<td>505.1594</td>
<td>1274.2750</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>8</td>
<td>26</td>
<td>46</td>
<td>124</td>
<td>63</td>
<td>26.7918</td>
<td>-198.6389</td>
<td>1322.5990</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>9</td>
<td>68</td>
<td>92</td>
<td>62</td>
<td>58</td>
<td>26.7390</td>
<td>226.7033</td>
<td>1434.6640</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>10</td>
<td>12</td>
<td>37</td>
<td>47</td>
<td>75</td>
<td>26.6955</td>
<td>-460.5746</td>
<td>1120.0560</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>11</td>
<td>37</td>
<td>141</td>
<td>60</td>
<td>26.6356</td>
<td>-511.0810</td>
<td>1205.0560</td>
<td></td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>12</td>
<td>46</td>
<td>62</td>
<td>130</td>
<td>68</td>
<td>26.5043</td>
<td>367.7256</td>
<td>1349.3830</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>13</td>
<td>31</td>
<td>50</td>
<td>21</td>
<td>62</td>
<td>26.4431</td>
<td>-63.0861</td>
<td>1246.8710</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>14</td>
<td>1</td>
<td>4</td>
<td>124</td>
<td>97</td>
<td>26.4358</td>
<td>-447.1980</td>
<td>1082.9330</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>15</td>
<td>50</td>
<td>66</td>
<td>112</td>
<td>30</td>
<td>26.3119</td>
<td>515.0131</td>
<td>1283.9090</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>16</td>
<td>14</td>
<td>37</td>
<td>18</td>
<td>70</td>
<td>26.2371</td>
<td>-450.4071</td>
<td>1092.8320</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>17</td>
<td>17</td>
<td>41</td>
<td>124</td>
<td>90</td>
<td>26.0349</td>
<td>-346.6365</td>
<td>1229.0770</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>18</td>
<td>69</td>
<td>101</td>
<td>44</td>
<td>49</td>
<td>25.9598</td>
<td>502.4506</td>
<td>1350.6440</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>19</td>
<td>15</td>
<td>37</td>
<td>170</td>
<td>50</td>
<td>25.9259</td>
<td>-530.0738</td>
<td>1229.0770</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>20</td>
<td>18</td>
<td>41</td>
<td>58</td>
<td>61</td>
<td>25.8600</td>
<td>-349.7366</td>
<td>1194.1560</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>21</td>
<td>64</td>
<td>87</td>
<td>71</td>
<td>63</td>
<td>25.8340</td>
<td>61.5157</td>
<td>1446.3220</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>22</td>
<td>21</td>
<td>42</td>
<td>65</td>
<td>60</td>
<td>25.6760</td>
<td>-321.0694</td>
<td>1214.9520</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>23</td>
<td>5</td>
<td>12</td>
<td>178</td>
<td>66</td>
<td>25.6440</td>
<td>-203.0483</td>
<td>1242.7950</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>24</td>
<td>38</td>
<td>57</td>
<td>84</td>
<td>62</td>
<td>25.3274</td>
<td>193.1102</td>
<td>1324.8380</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>25</td>
<td>65</td>
<td>88</td>
<td>41</td>
<td>68</td>
<td>25.2679</td>
<td>71.6483</td>
<td>1418.4600</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>26</td>
<td>60</td>
<td>78</td>
<td>32</td>
<td>78</td>
<td>24.9477</td>
<td>-228.7078</td>
<td>1347.2560</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>27</td>
<td>51</td>
<td>66</td>
<td>56</td>
<td>101</td>
<td>24.8499</td>
<td>449.6770</td>
<td>1250.8830</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>28</td>
<td>10</td>
<td>23</td>
<td>161</td>
<td>72</td>
<td>24.7814</td>
<td>189.5856</td>
<td>1270.5690</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>29</td>
<td>47</td>
<td>62</td>
<td>111</td>
<td>78</td>
<td>24.6996</td>
<td>353.0634</td>
<td>1333.5400</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>30</td>
<td>39</td>
<td>57</td>
<td>68</td>
<td>60</td>
<td>24.5755</td>
<td>193.2785</td>
<td>1308.0180</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>31</td>
<td>63</td>
<td>84</td>
<td>8</td>
<td>66</td>
<td>24.5306</td>
<td>-42.1248</td>
<td>1369.9460</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>32</td>
<td>44</td>
<td>61</td>
<td>67</td>
<td>63</td>
<td>24.4731</td>
<td>319.0387</td>
<td>1292.5420</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>33</td>
<td>34</td>
<td>53</td>
<td>37</td>
<td>66</td>
<td>24.2831</td>
<td>43.3615</td>
<td>1276.3230</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>34</td>
<td>45</td>
<td>61</td>
<td>57</td>
<td>45</td>
<td>24.1248</td>
<td>331.3734</td>
<td>1277.8450</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>35</td>
<td>33</td>
<td>50</td>
<td>40</td>
<td>62</td>
<td>23.9599</td>
<td>-64.2520</td>
<td>1266.5800</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>36</td>
<td>35</td>
<td>53</td>
<td>61</td>
<td>61</td>
<td>23.7105</td>
<td>69.8309</td>
<td>1299.8780</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>37</td>
<td>28</td>
<td>46</td>
<td>112</td>
<td>70</td>
<td>23.6840</td>
<td>-184.9778</td>
<td>1312.3290</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>38</td>
<td>41</td>
<td>58</td>
<td>84</td>
<td>70</td>
<td>23.4100</td>
<td>215.4423</td>
<td>1324.6640</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>39</td>
<td>53</td>
<td>66</td>
<td>117</td>
<td>62</td>
<td>22.8162</td>
<td>499.1581</td>
<td>1297.4350</td>
</tr>
<tr>
<td>DETFPT-!</td>
<td>40</td>
<td>59</td>
<td>74</td>
<td>60</td>
<td>75</td>
<td>22.4777</td>
<td>-371.7148</td>
<td>1320.2300</td>
</tr>
</tbody>
</table>
Figure 10.1. 3-D reflectivity plots versus relative doppler and range:
(a) 16th footprint, (b) 61st footprint.
Figure 10.2. Contour maps of reflectivity versus relative Doppler and range: 

(a) 1st footprint, (b) 61st footprint.
One can observe that impulse invariance has been practically achieved when plotting reflectivity versus doppler and range, even though footprint #16 corresponds to $a^* \approx 2.5^\circ$. Note however, that the azimuth resolution does depend upon $a$ as indicated in Fig. 6.1.

11. Demonstration of the DOPXY Simulation

The performance of DOPXY is demonstrated by displaying the output x–y reflectivity map for the same input files as in section 10. The HP 7475A plotter was used to obtain hard copies.

Running DOPXY with a positive value for KFPSGN will generate the file DOPXYG.GRD which contains the reflectivity data versus x–y coordinate for the selected footprint if the x–y window is correctly centered. There are two ways to define the x–y window, a manual setting and an automatic setting. For manual setting enter WINIWR = 0, and define the minimum corner coordinates and the window size in SYPAI.INP. For automatic setting enter WINIWR = 1, then the window is automatically centered on the center of the footprint by subroutine MAPWIN with entry level 4. Automatic setting is recommended because non overlapping window and footprint will produce an empty map. Of course, the reflectivity map will also be empty if there are no scatterers within the selected footprint. The maximum peak within the window is displayed on the screen. If there is one or more targets within the window, the peak should be much greater than zero. A peak equal to zero means that the window is empty, and the use of SURFER results into a warning "Plannar Grid File". Figures 11.1a and 11.1b show the 3–D reflectivity for the 16th and 61th footprints, respectively. Figures 11.2a and 11.2b show the corresponding contour maps. It is very obvious for both the 3–D plots and contour maps that the azimuth resolution is much better at large $a$. Yet we are able to detect targets anywhere within the scanned strip.

Running DOPXY with a negative value for KFPSGN will generate the file DOPXYG.GRD which contains the reflectivity data versus absolute x–y coordinates for all the footprints within the x–y window centered on the $|KFPSGN|$th footprint. Running
Figure 11.1. 3-D reflectivity versus absolute x-y: (a) 16th footprint, (b) 61st footprint.
Figure 11.2. Reflectivity contour maps versus x-y coordinates: (a) 16th footprint, (b) 61st footprint.
DOPXY for KFPSGN = -61 and displaying with SURFER produces Fig. 11.3a and 11.3b. Comparison of Fig. 11.3 with Figs. 11.2a and 11.2b show the appearance of new targets detected in footprints other than 61. Also, the scatterers that are both in 61 and some other footprints have a smoother shape due to the invariant mapping incoherent integration. Running DOPXY for KFPSGN = -16 yields the contour map shown in Fig. 11.4. Comparison to Fig. 11.2a shows again the presence of new detected scatterers and the incoherent integration smoothing effect. As already noted for the single footprint detection, azimuth resolution is poor when \( \alpha \) is very small.

We were limited by "DOS" (the execution file is about 550 kByte) to a map of size: \( x = 102 \) and \( y = 187 \) in DELMAP units. Since DELMAP = 0.75m then the size of the window in meters is \( x = 76.5 \text{m}, y = 140.25 \text{m} \). A rectangular window was selected to better match the shape of the footprint. Therefore, our approach is to provide an overall low resolution display, and to use DOPXYG.GRD for high resolution display of a selected window. Figure 11.5a shows a low resolution display for the actual clutter map, where the data is from file CLMPI.INP. Figure 11.5b shows a low resolution display for the detected clutter map where the data is as in Table 9.1. One can visualize the very close agreement between the actual and the detected clutter maps. One can also, check that the center of gravity for the high resolution contour maps corresponds closely to the actual scatterer location.

12. Summary, Conclusions and Recommendations for Future Work

This report extends the application of RRDP and invariant mapping to beam doppler sharpening radar, where the goal is to map a wide area on the ground. The antenna platform follows a horizontal flight path while the antenna scans the specified azimuth interval. Two main programs were developed: DOPFP.FOR and DOPXY.FOR. The first program detects scatterers in one or in all footprints, and also generates a file which contains the reflectivity data versus relative doppler and range for one footprint. The second program includes invariant mapping and generates a file which contains
Figure 11.3: Reflectivity versus absolute x-y:
(a) 3-D plot, (b) contour map.
Figure 11.4. Contour plot of the reflectivity versus absolute x-y for the 16th footprint.
Figure 11.5a. Actual x-y coordinates for the 40 scatterers.
Figure 11.5b. Detected x-y coordinates for the 40 scatterers.
reflectivity data versus absolute x–y coordinates for a selectable window. Our examples illustrate the detection of (40) forty scatterers randomly distributed on the ground.

The main contributions of this report are: (1) extending the simulation to a sequence of footprints, (2) making the programs work for positive/negative and very small azimuth, (3) generating a sequence of footprints which guarantees complete ground coverage, (4) developing subroutines to detect peaks, first within a single footprint and then within a sequence of footprints, (5) integrating the information for a sequence of footprints through invariant mapping, (6) implementing an automatic window selection to facilitate the use of the program, and (7) presenting both low resolution and high resolution displays to illustrate the results.

The main conclusions are:

(1) all the (40) forty scatterers where detected at the proper location, with a peak amplitude variation of about ±9%, around the mean,
(2) the range resolution and the relative doppler resolution remain constant,
(3) a formula which expresses the azimuth resolution as a function of the footprint azimuth $\alpha^*$, and of the elevation $\beta_j$ for the range cell is presented,
(4) the azimuth resolution is a nonlinear function of the footprint azimuth $\alpha^*$. In our application it is quite good for $|\alpha^*| > 8^\circ$, but much worse around $|\alpha^*| \approx 2^\circ$,
(5) the invariant mapping technique performs very well for any single footprint within the azimuth coverage. Very little resolution is lost through this mapping.
(6) invariant mapping performs a noncoherent integration over the sequence of footprints as expected.

Recommendations for future work include: (1) to generate a sequence of footprints which takes into account the mechanical limitations of the seeker, (2) to avoid as much as
possible small azimuth angles, where the azimuth resolution is much lower (for example, it
will be better to scan from (5° to 65°) rather than (−30° to +30°)), (3) to extend doppler
beam sharpening to nonlinear trajectories, and (4) to extend the doppler sharpening
technique to linear arrays for a 3-D scatterer detection.
References


APPENDIX A

Relevant Footprint Information For $\mathbf{m} = \{d_x, d_y, h\}$

A.1 Footprint center $q^* = q(t_c)$, see Fig. A.1.

Rectangular coordinates of footprint center = $\{x_q, y_q, 0\}$

range: $r^* = \sqrt{h^2 + (x_q - d_x)^2 + (y_q - d_y)^2}$

azimuth angle: $\alpha^* = \tan^{-1}[(x_q - d_x)/(y_q - d_y)]$ (A.1)

elevation angle = $\beta^* = \cos^{-1}\left(\frac{h}{r^*}\right)$

A.2 Range bounds and indexing, see Fig. A.1.

Minimum range: $R_{\text{Ranmn}} = \frac{h}{\cos(\beta^* - \theta_H)}$

Maximum range: $R_{\text{Ranmx}} = \frac{h}{\cos(\beta^* + \theta_H)}$ (A.2)

$\theta_H =$ half beamwidth

Range index for $q^*$: $j = \text{INT}[(r^* - R_{\text{Ranmn}} + \Delta r/2)/\Delta r]$

Maximum Range index: $J_{\text{RANMX}} = \text{INT}((R_{\text{Ranmx}} - R_{\text{Ranmn}} + \Delta r/2)/\Delta r)$

A.3 Information for the $j$th range cell

median range: $r_{j}^* = r^* + (j - j^*) \Delta r$

elevation angle: $\beta_{j}^* = \cos^{-1}\left(\frac{h}{r_{j}^*}\right)$ (A.3)

maximum relative elevation (see Fig. A.2): $\epsilon_{mx} = \beta_{j}^* - \cos^{-1}\left[\frac{h}{r_{j}^* - \Delta r/2}\right]$

maximum relative azimuth (see Fig. A.3): $\mu_{mx} = \cos^{-1}\left[\frac{\cos\theta_H - \cos\beta_j^* \cos\beta^*}{\sin\beta_j^* \sin\beta^*}\right]$
Figure A.1 Footprint center, range bounds, and jth range cell.
Figure A.2 Maximum relative elevation $\varepsilon_{mx}$ ($\Delta = \Delta r/2$).
Figure A.3  Maximum relative azimuth angle $\mu_{\text{max}}, \theta_H =$ half beamwidth.
APPENDIX B

Relevant Information for a Scatterer $\hat{C}_i$ (see Fig. B.1).

B.1 Computation of $\{r_{gi}, r_i, \alpha, \beta, \mu\}$ from $\{m, \hat{C}_i, \alpha^*\}$

$(\hat{m}_g, \hat{C}_g)$ are the vertical projections of $(\hat{m}, \hat{C}_i)$

ground range: $GR = |\hat{C}_g - \hat{m}_g| = \sqrt{[(x_i - d_x)^2 + (y_i - d_y)^2]}

range to $\hat{C}_g = r_{gi} = |\hat{C}_g - \hat{m}| = \sqrt{[GR^2 + h^2]}

slant range to $\hat{C}_i$: $r_i = |\hat{C}_i - \hat{m}| = \sqrt{[GR^2 + (h - h_i)^2]}

range cell index: $j = \text{INT}[(r_i - r^* + \Delta r/2)/\Delta r] + j^*$

(B.1)
elevation angle: $\beta = \tan^{-1}\left(\frac{GR}{h}\right)$

azimuth angle: $\alpha = \tan^{-1}\left[(x_i - d_x)/(y_i - d_y)\right]$

relative azimuth increment: $\mu = \alpha^* - \alpha$

B.2 Antenna Gain for $\hat{C}_i$ at time $t$.

Antenna angle to $\hat{C}_i$: $\text{Ang} = \text{Angle}(q^* - a(t), \hat{C}_i - a(t))$

let $w1 = \{a_x(t), a_y(t), a_z(t)\}$

$w2 = \{q_x, q_y, q_z\}$

and $w3 = \{C_{ix}, C_{iy}, C_{iz}\}$

$s = \sin(\text{Ang}) = \sqrt{1 - \text{DOTCOS}(w1,w2,w3)**2}$

(B.2a)

where the function DOTCOS is listed in Appendix E.

One-way Antenna gain = DISH(s,2,DOVL)

(B.2b)

where the function DISH is listed in Appendix C, and DOVL is the dish diameter in wavelength units.

The gain is computed at three times

$\{t = -\frac{D_{ob}}{2}, t = 0, t = \frac{D_{ob}}{2}\}$

(B.2c)
Figure B.1 Fictitious scatterer \( \mathbf{C}_j^* \) and \( \text{ith} \) scatterer \( \mathbf{C}_i \) in the \( \text{jth} \) range cell: \( \alpha_i = \alpha^* + \mu_i \).
Appendix C

Listings for DOPFP.FOR and Associated Subroutines

Program DOPFP.FOR

$LARGE

CHARACTER*240 TITLE
REAL W(23), TIMCNT, TIMDWL, DELRAN, XMMNPR, WPAR(8), XIFXMT, SELWIN,
& DETIPR, SELDIS, BITINT, WSQ(12), SHOSCA, BUGLEV
REAL SMOMAP(540), TXD(64), TXQ(64), WANGN9(265), WAGNCL(128),
& WBET9(265), WDOM(3), WDOVL(3), WEPSMX(265), WIND(64)
REAL VELOM, ALFM, BETM, WMPOSB(3), WMPOS(6), WPOSE(3), WQPOSB(3),
& WQPOS(6), WQPOS(3), WREQ(128), WMVELO(6)
REAL WMU3DB(265), WMUMDR(266), WRA9N(265), WTU9A(25), WTIME(64),
&WTSMO(5), XD(128), XM(128), XQ(128), WADRL0(265), WADRUP(265),
&WADSMO(90), WAGAIN(3), HEIREF, BETREF, RANREF
INTEGER LNPPP(128), NBSCA, KFPDIS, NW, LEVBUG, NBFP, NBCSCN, JRSCMN,
& JRSCMX, IWZMX, JZWZMX, KFPSCN, ISURF, IPRED, NBINCNU, NBI, NBINS, NPIXFP
REAL DISH, DOTCOS, FREQMFU, FRECO, COSANT, GRDMAP, TMU3DB, WINIWR,
& FPMXB, RELDOP, BET, SURFIN
REAL CLIGHT, DELFIN, DELFUT, DELTIM, DOVL, EPS, FMU, FRECO, GDSH2,
& GDISH3, GRC, HDELRA, HTHETA, HTIMDW, RANCOF, OMEGO, SCAFA,
& SINANG, THETA, TMBET, TPI, V1, V2, X, XSCA, YSCA, Z, FXM, SCANEW
INTEGER IDDISH, IFIPE, INDX, INTCOF, IW, J, X, XSCA, Y, YSCA, Z, Y
& KASMO, KDQ, KI, KJ, KF, KNPPP, KSCALO, KSCAUP, KT, KTM, KTHSCA,
& NBI, NBTFX, NNF, NFFTH, NFFFX, NFFXH, NNPMMN, NNPMPX, NCSFAC,
& NBFPMX, KFP
EXTERNAL YMU3DB, FRTOMU
INTEGER MNIW, MNJW, MXIW, MXJW, NBSCDT, IWFPT, IWRRAN, IWFRQ, JWFPT,
& JWRAN, JWFRQ
REAL ZM, SYNMAP, WSCDTX, WSCDTY, WSCDTM, VX, VY, VM
COMMON /DETECT/ NBSCDT, WSCDTX(400), WSCDTY(400), WSCDTM(400),
& IWFPT(400), IWRRAN(400), IWFRQ(400), NBINCNU, VX(129), VY(129)
& VM(129), JWRAN(129), JWFRQ(129), JWFPT(129), NBINOL
COMMON /SURFP/ SYNMAP(128,200), MNIW, MNJW, MXIW, MXJW, ZMX, NPNFP
EQUIVALENCE (W(1), TIMDWL), (W(2), DELRAN), (W(3), XMMNPR), (W(4),
& WPAR(1), W(12), XIFXMT), (W(13), SELWIN), (W(14), DETIPR), (W(15),
& SELDIS), (W(16), WINIWR), (W(17), SCANEW), (W(18), HEIREF),
& (W(19), BETREF), (W(20), SHOSCA), (W(21), BUGLEV), (W(22), FPMXNB),
& (W(23), SURFIN)
C OUTPUT PARAMETERS IN WPAR= (DELMAP, THRESH, XMNXMAP, YMNMAP)
C , XMMNMAP, YMMNMAP, XSZ, YSZ) ****************************
C WVELO=(XVELO, YVELO, ZVELO, VELOM, {2/WAVLNGH}, DELFUT) **************
C WMPOS=(XM, YM, ZM, HTTHETA, RANMN, RANMX) ****************************
C WQPOS=(XP=QX9, YFP=YQ9, ZFP=ZQ9, ALFQ9, BETQ9, RANQ9) **************
C DIMENSION OF SMOMAP = NBSCA*15 (SEE SUB. CLUFPT) **************
FRQOFMU(V1, FMU, V2, V3)=-V1*(V2*FMU+(V3/2.)*FMU**2) **************
C I/O FILES : U9=SYPAI.INP(main); U3=DOPFP.PRT(main); U11=CLMPI.INP

48
C (clumap), U10 = FPSQI.INP (made in fpidop, used in bcedop), U12 = GEODOPI
C &INP(fpidop); U7 = DOPPFG GRD(main) 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
C HALF 3db BEAMWIDTH = 0.015 gives DOVL = 14.83343168 = DISH DIAMETER IN
C &LAMBDA UNITS (2-WAY GAIN = -3.00 DB means DISH = 0.84139514)

DATA THETA, FREQ, CLIGHT, TPI, EPS, WDOVL, NW, BITINT/0.043, 94, E9, 3, E8,
& 6, 28318531, 0.001, 17.15094, 17.15094, 14, 80884, 23.1, /
DATA WQPOSB(3), WQPOSC(3), WQPOSE(3)/0, 0, 0, /
NBSCDT = 0
SINANG = SIN(0.0215)
GDISH2 = DISH(SINANG, 2, WDOVL(2))
GDISH3 = DISH(SINANG, 3, WDOVL(3))
OPEN(UNIT = 3, FILE = 'DOPPFG.PRT', STATUS = 'OLD')
WRITE(3, *) GDISH2, GDISH3, GDISH3
PRINT*, 'WPAR not used in DOPFP except for WPAR(2) = SURF-threshold'
TITLE = 'TIMDWL = 1; DELRAN = 2; XMNMNR = 3; WPAR1 = 4 = DELMAP. WPAR2 = THRESH. WPAR
& 3 = XMN; WPAR4 = YMN; WPAR5 = XMX; WPAR6 = YMX; WPAR7 = XSIZE; WPAR8 = YSIZE; XIFX
& = 12; SELWIN = 13; DETIPR = 14; SELDIS = 15; WINIWR = 16; SCANEW = 17; HEIREF = 18; BE
& TREF = 19; SHOSCA = 20; BUGLEV = 21; FMXNB = 22; SURFIN = 23'
OPEN(UNIT = 9, FILE = 'SYPAI.INP', STATUS = 'OLD')
CALL WINPUX(NW, W, 9, TITLE)
IDDISH = INT(SELDIS + 0.0001)
LEVDUG = INT(BUGLEV + 0.0001)
ISURF = INT(SURFIN + EPS)
IPRDET = INT(DETIPR + EPS)
NBFPMX = INT(FMXNB + EPS)
DOVL = WDOVL(IDDISH)
OMETG = TPI*FREQ
HDELRA = DELRAN*0.5
HTHETA = THERA*0.5
WMVPOSC(4) = HTHETA
NBITFF = 6
NBFX = NBITFF + INT(BITINT + EPS)
NFFT = 2**NBFX
NFFT1 = NFFT/2
NFFX = 2**NBFX
NFFX1 = NFFX/2
DELFIN = TIMDWL
INTCOF = INT(TIMDWL + EPS)
DELFUT = DELFIN/FLOAT(INTCOF)
WMVELO(6) = DELFUT
DELTIM = TIMDWL/NFFT
HTIMDWL = TIMDWL*0.5

C COMPUTE RANGE REFERENCE TO USE IN RANCOF ****************************
RANREF = HEIREF/COS(BETREF)
CALL CLUMAP(HTHETA, LEVBUG)
OPEN(UNIT = 10, FILE = 'FPSQI.INP', STATUS = 'OLD')
PRINT* ****************************
PRINT*, 'SELECT FP TO DISPLAY: KFPSGN, ABORT = -999; ALL FP = -KFPSGN'
CALL KINPUT(KFPSGN)
IF(KFPSGN EQ -999) STOP 'ABORT'
KFDPIS = IABS(KFPSGN)
CALL FPIDOP(NBFPMX, NBFP, HTHETA, TIMDWL, KFDPIS)
PRINT* ****** KFDPIS =; NBFPMX =; NBFP =; , KFDPIS, NBFP , ******
C COMPUTE ARRAY OF RELATIVE TIME WTIME *******
    DO 2 KTIM=1,NFFT
2    WTIME(KTIM)=DELTIM*(KTIM-NFFTH-0.5)
C INDEX KNPPP=1,NFX FOR FREQ=[-NFFX+1)*DELFUT] ******
C COMPUTE ARRAY LNPPP,THEN KFF=LNPPP(KNPPP) & WFREQ(KNPPP) ********
    DO 9 KNPPP=1,NFFX
9    WFREQ(KNPPP)=DELFUT*FLOAT(KNPPP-NFFTH-1)/FREQO
     INDX=KNPPP+NFFX
     IF(INDX.GT.NFFX) INDX=INDX-NFFX
C COMPUTE ARRAY OF RELATIVE TIME WTIME *******
    DO 1 KFP=1,NBFP
1    CALL BCEDO(W,NFFT,WTIME,VELOM,ALFM,BETM,WMP0SB,WMP0SC,WMP0E.
       &WQPOSB,WQPOSC,WQPOSE,WMVELO,JRANQ9,JRANMX,KFPDIS,KFP)
    IF(KFP.NE.KFPDIS.AND.KFPSGN.GT.0) GO TO 1
1    WITE(*,*) 'VELOM, ALFM, BETM, XM, YM, ZM, XFP, YFP'
C COMPUTE 3DB FOOTPRINT LIMITS : (NPPPMN,NPPPMX) ********
C TO LIMIT # CELLS BEING PROCESSED TO KNPPP=NPPPMN,NPPPMX *********
7    TMU3DB(JRAN)=SIGN(WMU3DB(JRAN),WQPOSC(4))
     NPPPMN=FRQOFMU(FREQO,TMU3DB(JRAN),WTAU(6))/DELFUT-NFFX-5
     NPPPMX=FRQOFMU(FREQO,-TMU3DB(JRAN),WTAU(6))/DELFUT+NFFX+5
     IF(ABS(WQPOSC(4)).LT.WMU3DB(JRAN)) NPPPMX=FRQOFMU(FREQO,-WQPOSC(4))
& WTAU(6), WTAU(10))/DIELUT+NFFXH+5.
C COMPUTE MU AT CENTER OF CELLS FOR INCREASING DOPPLER **************
DO 17 K=NPPPMN-1,NPPPMX+1
17 WMUMDR(K)=SIGN(1.,WQPOSC(4))*FRUMU(WTAU,WFreq(K))
C COEFFICIENT FOR RANGE NORMALIZATION: RANCOF ***************
RANCOF=(WRAN9(JRAN)/RANREF)**4
KSCAUP=WADRUP(JRAN)+EPS
DO 30 KTHSC=KSCALO,KSCAUP
KADSMO=WADSMO(KTISCA)+EPS
C INFORMATION ON SCATTERER, WTS=AMPSCA, FASSC=MUSCA,JRAN-SCA, ZSCA
DO 31 KTHSC=KSCALO,KSCAUP
31 WTSO(K)=SOMA(KADSMO-44-KT)
C ANTIMEG(T)=WAGAIN(1)+WAGAIN(2)*t+WAGAIN(3)*t-2 ***************
C WAGAIN IS OUTPUT OF SCAGAN, INPUTS ARE DOVL,HITMDW, WMPOS=(BEG. & CENT. END), WQPOSC, WTSO=(AMP, FAS, MU, JRAN, ZSCA), JRANQ9, DELRAN
CALL SCAGAN(IDDISH, DOVL, HTIMDW, WMPOS, WTSO, WQPOSC, WQPOSE, WMPOSB, WQPOSC, & WQPOB, TSQPO, JRANQ9, DELRAN, WAGAIN)
CALL TSIGNAL(OMEGO, NFFT, RANCOF, XIFXMT, WTSO, WTAU, WTIME, WAGAIN, TXD & TXQ)
DO 33 KDO=1,NFFT
XD(KDO)=XD(KDO)+TXD(KDO)
33 XQ(KDO)=XQ(KDO)+TXQ(KDO)
C CONTINUE
DO 37 K=1,NFFT
XD(K)=XD(K)*WIND(K)
37 XQ(K)=XQ(K)*WIND(K)
CALL FFT(-NBITFX, 1, 1, 1, XD, XQ)
C COMPUTE ARRAY OF ANTENNA GAIN AT CENTER OF CELLS **************
DO 39 KNPPP=NPPPMN,NPPPMX
39 WMUMDR(K)=DSQRT(1.+EPS-DSIN(WQPOSC(4))*DSIN(WBET9(JRAN))*DCOS(WMUMDR(KNPPP))+DCOS(WQPOSC(5))*DCOS(WBET9(JRAN)))**2
WAGNCL(KNPPP)=DISH(SINANG, IDDISH, DOVL)**2
C (XD,XQ) NEED TO BE SCALLED FOR RANGE AND ANTENNA GAIN AT KNPPP
DO 41 KNPPP=NPPPMN,NPPPMX
41 KFF=LNPPP(KNPPP)
SACFAC=1./(RANCOF*WAGNCL(KNPPP))
XD(KFF)=XD(KFF)*SACFAC
XQ(KFF)=XQ(KFF)*SACFAC
XM(KFF)=SQR(XD(KFF)**2+XQ(KFF)**2)
IF(XM(KFF).GE.XMMNPR) CALL DETFT(KFF, JRAN, KNPPP, WUMDR(KNPPP), & JRANQ9, DELRAN, WMPOS, WQPOSC(1), IPREDET)
IF(ISUFR, EQ. 1. AND. KFPDIS.EQ. KFP) CALL & SURFMP(KNPPP, JRAN, XM(KFF), JRANMX, 2)
41 CONTINUE
35 CONTINUE
IF(NBINCU.GT.0) CALL DETFT(KFF, JRAN, KNPPP, WDUM(1), WDUM(2), & JRANQ9, DELRAN, WMPOS, WQPOSC(2))
1 CONTINUE
CALL DETFT(KFF, JRAN, KNPPP, WDUM(1), WDUM(2), JRANQ9, DELRAN, & WMPOS, WQPOSC(3))
CLOSE (3)
51
SUBROUTINE BCEDOP(W,NFFT,WTIME,VELOM,ALFM,BETM,WMPOSB,WMPOSC, & WMPOSE,WQPOSB,WQPOSC,WQPOSE,WMVELO,JRANQ9,JRANMX,KFPDIS,KFP) 
INTEGER K,K1,K2,JRANQ9,JRANMX,KFPDIS,KFP,NFFT 
REAL W(1),VELOM,ALFM,BETM,WMPOSB(1),WMPOSC(1),WMPOSE(1),WQPOSB(1), & WQPOSC(1),WQPOSE(1),WMVELO(1),WTIME(1),GRC,RANMN,RANMX,ALFQ9,RANQ9 
C WQPOSC=(XFP,YFP,ZFP,ALFQ9,BETQ9,RANQ9) 
C UPDATING FOOTPRINT INFO. BY READING FILE FPSQI.INP 
IF(KFP.EQ.KFPDIS. .AND. KFPDIS. .LT. 999) PRINT*, '@@@ SUB-BCEDOP reads KF 
& KFPDIS=',KFPDIS,' row of FPSQI.INP & writes WMPOS(B,C,E),WQPOS(B,C, & E),WMVELO(x,y,z) @@@' 
IF(KFPDIS.EQ.999) PRINT*, '@@@ SUB-BCEDOP IN FP=',KFP 
READ(10,18) VELOM,ALFM,BETM,(WMPOSC(K1),K1=1,3),(WQPOSC(K2),K2=1,2, & & K2=1,2),ALFQ9 
C COMPUTE MISSILE INFORMATION 
******************S 
WMVELO(1)=VELOM*SIN(BETM)*SIN(ALFM) 
WMVELO(2)=VELOM*SIN(BETM)*COS(ALFM) 
WMVELO(3)=-VELOM*COS(BETM) 
WMVELO(4)=VELOM 
DO 21 K=1,3 
WMPOSB(K)=WMPOSC(K)+WTIME(1)*WMVELO(K) 
21 WMPOSE(K)=WMPOSC(K)+WTIME(NFFT)*WMVELO(K) 
C COMPUTE FOOTPRINT INFORMATION AT WQPOSC 
GRC=SQRT((WQPOSC(1)-WMPOSC(1))**2+(WQPOSC(2)-WMPOSC(2))**2) 
WQPOSC(4)=DATAN2((WQPOSC(1)-WMPOSC(1)),(WQPOSC(2)-WMPOSC(2))) 
WQPOSC(5)=DATAN2(GRC,WQPOSC(3)) 
WQPOSC(6)=DSQRT(GRC**2+WMPOSC(3)**2) 
WQPOSB(1)=WMPOSB(3)*DTAN(WQPOSC(5))*DSIN(WQPOSC(4))+WMPOSB(1) 
WQPOSB(2)=WMPOSB(3)*DTAN(WQPOSC(5))*DCOS(WQPOSC(4))+WMPOSB(2) 
WQPOSE(1)=WMPOSE(3)*DTAN(WQPOSC(5))*DSIN(WQPOSC(4))+WMPOSE(1) 
WQPOSE(2)=WMPOSE(3)*DTAN(WQPOSC(5))*DCOS(WQPOSC(4))+WMPOSE(2) 
RANMN=WMPOSC(3)/DCOS(WQPOSC(5))-WMPOSC(4)) 
RANMX=WMPOSC(3)/DCOS(WQPOSC(5)+WMPOSC(4)) 
RANQ9=WMPOSC(3)/DCOS(WQPOSC(5)) 
WMPOSC(5)=RANMN 
WMPOSC(6)=RANMX 
JRANQ9=INT((WQPOSC(6)-RANMN+0.5*W(2))/W(2)) 
JRANMX=INT((RANMX-RANMN+0.5*W(2))/W(2)) 
18 FORMAT(F10.3,2F10.6,5F10.3,F10.6) 
IF(KFP.NE.KFPDIS) RETURN 
PRINT*, 'VELOM,ALFM,BETM,(WMPOSC(K1),K1=1,3),(WQPOSC(K2),K2=1,2)' 
PRINT*, 'VELOM,ALFM,BETM,(WMPOSC(K1),K1=1,3),(WQPOSC(K2),K2=1,2)' 
PRINT*, 'YM,ALFQ9,WQPOSC4,RANQ9,RANMN,RANMX,JRANQ9,JRANMX' 
52
Subroutine CLUFPT.FOR

SUBROUTINE CLUFPT(DELRAN,JRANQ9,JRANMX,WQ9,WMPOSC,NSCAFP,SMOMAP,
&WADRLO,WADRUP,WADSMO)
C CREATED ON 1/24/1989, 1300 hr ****************************************
REAL WMPOSC(1),WQ9(1),WQPOS(2),SMOMAP(1),
&GRDMAP,WADRLO(1),WADRUP(1),WJRAN(200),WADSMO(1),W(34)
REAL COSANT,HDELRA,DOTCOS,EPS
INTEGER NSCA,KSMO,KSCA,KX,JRANON,KSCAT,KINC,K,J
COMMON /GRDCLU/NSCA,GRDMAP
PRINT*,100000a SUB-CLUFPT: uses GRDCLU to compute SMOMAP @@@@
C WQ9=(XQ,YQ,ZQ,ALFQ9,BETQ9,RANQ9) *
C SMOOTHED OUTPUT FOR CURRENT FOOTPRINT IS SMOMAP, ONE REAL
C &SCATTERER GIVES 15-COMPONENTS **********************************
C SMOMAP=(EARLY,SCAFAS,MU,JRANON-1,ZSCA,ON,SCAFAS,MU,JRANON,ZSCA,LAT
C &E,SCAFAS,MU,JRANON+1,ZSCA) ****************************************
KSMO=0
HDELRA=DELRAN/2.
C WSCA=(X-SCA,Y-SCA,Z-SCA,AMP-SCA,FAS-SCA) **************************
DO 5 KS CA=1,NBSCA
KX=(KSCA-1)*5+3
WSCA(1)=GRDMAP(KX)
WSCA(2)=GRDMAP(KX+1)
WSCA(3)=GRDMAP(KX+2)
WSCA(4)=GRDMAP(KX-2)
WSCA(5)=GRDMAP(KX-1)
IF(WSCA(4).LT.0.001) GO TO 5
IF(DOTCOS(WMPOSC,WQ9,WSCA).LT.COSANT) GO TO 5
C REPLACE (X,Y) BY (MU,4N3CA) ***************************************
WSCA(6)=WSCA(1)
WSCA(7)=WSCA(2)
WSCA(1)=ATAN2(WSCA(6)-WMPOSC(1),WSCA(7)-WMPOSC(2))-WQ9(6)
WSCA(2)=SQRT((WSCA(6)-WMPOSC(1))**2+(WSCA(7)-WMPOSC(2))**2+
&(WSCA(3)-WMPOSC(3))**2)
C RANGE SMOOTHING:1-SCATTERER GIVES WMSCA=(EARLY,ON,LATE) ********
JRANON=((WSCA(2)-WQ9(6)+HDELRA)/DELRAN)+JRANQ9
WRITE(3,*),JRANON,' JRANON=',JRANQ9
CALL RANGSMOO(DELRAN,JRANON,JRANQ9,WQ9(6),WMPOSC,WSCA,WMSCA)
KSCAT=0
DO 7 KINC=2,12,5
SMOMAP(KSMO+KINC)=WSCA(5)
SMOMAP(KSMO+KINC+1)=WSCA(1)
KSCAT=KSCAT+1
53
SMOMAP(KSMO+KINC-1)=WMSCA(KSCAT)
SMOMAP(KSMO+KINC+2)=JRANON-2+KSCAT
7 SMOMAP(KSMO+KINC+3)=WSCA(3)
KSMO=KSMO+15
5 CONTINUE
EPS=0.00001
NSCAFP=0
DO 10 K=4,KSMO,5
NSCAFP=NSCAFP+1
WADSMO(NSCAFP)=K
10 WJRAN(NSCAFP)=SMOMAP(K)
CALL SORTTRI(WJRAN,WADSMO,-NSCAFP)
DO 12 J=1,JRANMX
WADRLO(J)=0.
WADRUP(J)=0.
DO 14 K=NSCAFP,1,-1
14 IF(ABS(WJRAN(K)-FLOAT(J)).LE.EPS) WADRLO(J)=FLOAT(K)
12 CONTINUE
PRINT*,...EXIT SUB-CLUFT.......
RETURN
END

Subroutine CLUMAP.FOR

SUBROUTINE CLUMAP(HTHETA,LEVBUG)
C CREATED 02/09/2200  
CHARACTER*240 TITLE
REAL W(34),GRDMAP,HTHETA,COSANT,W1,XI,YI,WGRD(6),SCANB
INTEGER KSCA,KK,NBSCA,K,LEVBUG
COMMON /GRDCLU/NBSCA,GRDMAP(501),COSANT
PRINT*,...SUBROUTINE CLUMAP.......
OPEN(UNIT=11,FILE='CLMPI.INP',STATUS='OLD')
C CLUTTER INFO. IS IN CLMPI.INP FORMATTED AS FOLLOWS:
C FIRST ROW=NBSCA
C OTHER ROWS {AMP--SCA, FAS--SCA, X--SCA, Y--SCA, Z--SCA}
READ(11,50) SCANB
NBSCA=INT(SCANB+.0001)
40 FORMAT(6(1X,F10.3))
42 FORMAT(1X,' GRDMAP(K),K=1,5) '/5(1XF10.3)
50 FORMAT(6F10.3)
COSANT=COS(HTHETA)
KK=0
IF(LEVBUG.GE.1)PRINT*,'AMP--SCA, FAS--SCA, X--SCA, Y--SCA, Z--SCA,SCA#'
DO 3 KSCA=1,NBSCA
READ(11,50) (WGRD(K),K=1,6)
IF(LEVBUG.GE.1) WRITE(*,40) (WGRD(K),K=1,6)
SUBROUTINE CLUMAP(HTHETA,LEVBUG)
DO 5 K=1.5
    KK=KK+1
3 CONTINUE
CLOSE (11)
IF(LEVBUG.EQ.2) WRITE(*,42) (GRDMAP(K),K'=1,5)
PRINT *,**** EXIT CLUMAP LEVBUG=',LEVBUG
RETURN
END

Subroutine DETFPT.FOR

SUBROUTINE DETFPT(KFP ,JRAN,KFRQ ,FMUCEL,XM,JRANQ9,DELRAN,WMP OSC.
&WQPOSC,IN123,IPRT1)
REAL ALFQ9,RANSCA,BETSCA,FMUCEL,WQPOSC(1),WMPOSC(1),XSCA,
&YSCA,ALFSCA,GRNSCA,RANMN,RANMX,RANQ9,DELTRAN,XM
REAL WSCDTX,WSCDTY,WSCDTM INTEGER KFP ,JRAN,KFRQ,JRANQ9,IN123,K,NBSCDT,IWFPT,IWRAN,IWFRQ,
&IPRT1,JWFPT,JWRAN,JWFRQ,NBINCU,NBINOL
REAL WM(400),WX(400),WY(400),VM,VX,VY
INTEGER NW,LOCBEG(400),LOCEND(400),NPIIC,LOCPIK(200),KAD,ICOMPR
COMMON /DETECT/NBSCDT,WSCDTX(400),WSCDTY(400),WSCDTM(400),
&WQPOSC(400),IWFPT(400),IWRAN(400),IWFRQ(400),NBINCU,VX(129),VY(129)
,&VM(129),JWFPT(129),JWRAN(129),JWFRQ(129),JWFPT(129),NBINOL
4 FORMAT(I1X,'DETFPT-FOR4,3(XI4),2X,3(1XF11.3))
GO TO (1,2,3),1N123
1 NBINCU=NBINCU+1
    ALFQ9=WQPOSC(4)
    RANQ9=WQPOSC(6)
    RANMN=WMPOSC(5)
    RANMX=WMPOSC(6)
    RANSCA=RANQ9+DELRAN*FLOAT(JRAN-JRANQ9)
    BETSCA=ACOS(WMPOSC(3)/RANSCA)
    GRNSCA=WMPOSC(3)*TAN(BETSCA)
    ALFSCA=ALFQ9+FMUCEL
    XSCA=GRNSCA*SIN(ALFSCA)+WMPOSC(1)
    YSCA=GRNSCA*COS(ALFSCA)+WMPOSC(2)
    VX(NBINCU)=XSCA
    VY(NBINCU)=YSCA
    VM(NBINCU)=XM
    JWFPT(NBINCU)=KFP
    JWRAN(NBINCU)=JRAN
    JWFRQ(NBINCU)=KFRQ
9 FORMAT('DETFPT-1B: '4(1XI3),1X,3(1XF10.4))
IF (IPRT1.EQ.1) WRITE(*,9) KFP,JRAN,KFRQ,NBINCU,XM,XSC \,YSCA
NBINOL=NBINCU
RETURN

55
SUBROUTINE FPIDOP(NBFPMX,NBFPT,HTHETA,TIMDWL,KFPDIS)
INTEGER KFP,NBFPT,K,NBCY,NBFPCY,KA,KB,I,NBFPMX
REAL ANGCY,DLALF,FPLNG,PERCY,DLTISQ,HDLTIS,ALFONE,ALFFP,TIMINC
REAL XFP,YFP,VELOM,ALFM,BETM,XM,YM,ZM,SALFM,CALFM,SBETM,CBETM,
      &SBETFP,CALFCY,CBETFP,GR
REAL V(12),WMESSO(3),WMPOSC(3)

56
CHARACTER*240 TITLE
PRINT*: '****** SUB-FPIDOP: information for FP--sequence ******'
PRINT*: 'GEODOLP.INP; STATUS='OLD')
OPEN(UNIT=12,FILE='GEODOLP.INP',STATUS='OLD')
TITLE='VZLOM=1; ALFM=2; BETM=3; WMBRUN=4-6; BETFP=7; ALFCY=8 /
&OVRLAP=g; CYNB=10; TIMGAP=11; ALFSTR=12',
CALL WINPUX(12,V,12,TITLE)
CLOSE(12)
SALFM=SIN(V(2))
CALFM=COS(V(2))
SBETM=SIN(V(3))
CBETM=COS(V(3))
SBETFP=SIN(V(7))
CBETFP=COS(V(7))
NBCY=INT(V(10)+0.001)
CALFCY=COS(V(8))
ANGCY=ACOS(SBETFP**2*CALFCY+CBETFP**2)
NBFPACY=INT((ANGCY**V(9))/(4.*HTHETA) + 0.5) * 2
DLALF=V(8)/FLOAT(NBFPACY)
FPLNG=V(6)*TAN(V(7)+HTHETA)-TAN(V(7)-HTHETA))
WMVELO(1)=V(1)*SBETM*SALFM
WMVELO(2)=V(1)*SBETM*CALFM
WMVELO(3)=-V(1)*CBETM
PERCY=FPLNG/V(9))WMVELO(2)
DLTISQ=(PERCY*V(11))/FLOAT(NBFPACY)
HDLTIS=0.5*DLTISQ
PRINT*: 'NBCY, ANGCY, NBFPACY, DLALF, FPLNG, PERCY, DLTISQ',
WRITE(*,*) NBCY, ANGCY, NBFPACY, DLALF, FPLNG, PERCY, DLTISQ
IF(DLTISQ.LT.TIMDWL) STOP 'ERROR--FPIDOP'
NBFP=NBFPCY=NBCY
IF(NBFP.GT.NBFPMX) NBFP=NBFPMX
DO 10 K=1,3
10 WMPOSC(K)=V(K+3)+WMVELO(K)*HDLTIS
ALFONE=-0.5*(V(8)-DLALF)
ALFFP=ALFONE
GR=WMPOSC(3)*TAN(V(7))
XFP=GR*SIN(ALFFP)+WMPOSC(1)
YFP=GR*COS(ALFFP)+WMPOSC(2)
REWIND 10
WRITE(10,1)(V(KA),KA=1,3),(WMPOSC(KB),KB=1,3),XFP,YFP,ALFFP
1 FORMAT (F10.3,2F10.6,5F10.3,F10.6)
IF(NBFP.EQ.1) GO TO 6
DO 12 KFP=2,NBFP
I=MOD((KFP-1),NBFPACY)
DO 14 K=1,3
TIMINC=DLTISQ
IF(I.EQ.0) TIMINC=V(11)+DLTISQ
14 WMPOSC(K)=WMPOSC(K)+WMVELO(K)*TIMINC
ALFFP=ALFFP+DLALF
IF(I.EQ.0) ALFFP=ALFONE
GR=WMPOSC(3)*TAN(V(7))
XFP=GR*SIN(ALFFP)+WMPOSC(1)
YFP=GR*COS(ALFFP)+WMPOSC(2)

WRITE(10,1)(V(KA),KA=1,3),(WMPOSC(KB),KB=1,3),XFP,YFP,ALFFP

CONTINUE

REWIND 10
PRINT *, 'VELOM ALFM BETM XM YM ZM XFP YFP ALFFP'
DO 16 K=1,NBFP
READ(10,1) VELOM, ALFM, BETM, XM, YM, ZM, XFP, YFP, ALFFP
IF(K.EQ.KFPDIS) WRITE(*,2) VELOM, ALFM, BETM, XM, YM, ZM,
& XFP, YFP, ALFFP
CONTINUE 2
FORMAT(9(1X,F8.3))
REWIND 10
PRINT *, '000@0@ EXIT SUB- FPIDOP @@@@'
RETURN
END

Subroutine FPTPIK.FOR

SUBROUTINE FPTPIK(WM,WX,WY,LOCBEG,LOCEND,NW,JRNDIS,JAZDIS,NPIK,IPRT)
INTEGER LOCBEG(1),LOCEND(1),NW,NPIK,K,IPRT
REAL WM(1),WX(1),WY(1),XS,YS
CALL SOR3RL(WM, WX, WY, LOCBEG, NW)
DO 20 K=1,NW
20 IF(IPRT.EQ.1)PRINT*, 'FPTPIK+SORT', K, WM(K), WX(K), WY(K)
IDSCA=0
DO 10 K=NW,1,-1
IF(LocEnd(K).NE.0) GO TO 10
IDSCA=IDSCA+1
LOCEND(K)=IDSCA
KS=K
XS=WX(K)
YS=WY(K)
IF(KS.EQ.1) GO TO 10
DO 13 KK=KS-1,1,-1
IF(LOCEND(KK).NE.0) GO TO 13
IF(ABS(XS-WX(KK)).LT.FLOAT(JRNDIS).
&ABS(YS-WY(KK)).LT.FLOAT(JAZDIS)) LOCEND(KK)=IDSCA
13 CONTINUE
CONTINUE
NPIK=IDSCA
DO 25 KID=1,NPIK
KFIND=0
DO 35 K=NW,1,-1
IF(KFIND.EQ.1.OR.LOCEND(K).NE.KID) GO TO 35
KFIND=1
LOCPIK(KID)=LOCBEG(K)
35 CONTINUE
CONTINUE
IF(IPRT.EQ.1)PRINT*,’FPTPIK–END: KPIK, LOCPIK, LOCBEG’
DO 37 K=1,NPIK
IF(IPRT.EQ.1)WRITE(*,4) K,LOCPIK(K),LOCBEG(K)
37 CONTINUE
4 FORMAT(1X, ’FPTPIK–END’,3(2X,I5))
RETURN
END

Function FRTOMU.FOR

FUNCTION FRTOMU(WTAU,FRQCEL)
REAL FRTOMU,WTAU(‘),FRQCEL
DOUBLE PRECISION TEMP,ATAU6,TMU
ATAU6=ABS(WTAU(6))
TEMP=ATAU6**2-2.*WTAU(10)*FRQCEL
IF( TEMP.GT.1.E-16) TMU=(-ATAU6+DSQRT(TEMP))/WTAU(10)
IF(TEMP.LE.1.E-16) TMU=-FRQCEL/ATAU6
FRTOMU=TMU
RETURN
END

Subroutine LSTPIK.FOR

SUBROUTINE LSTPIK(WM,WX,WY,LOCBEG,LOCEND,NW,DISPIK,LOCPIK,NPIK, &IPRT)
INTEGER LOCBEG(1),LOCEND(1),NW,LOCPIK(1),NPIK,K,IDSCA,KS,KK,KID, &KFIND,IPRT
REAL WM(1),WX(1),WY(1),DISPIK,XS,YS
CALL SOR3RL(WM,WX,WY,LOCBEG,NW)
DO 20 K=1,NW
20 IF(IPRT.EQ.1)PRINT*,’LSTPIK+SORT’,K,WM(K),WX(K),WY(K)
  IDSCA=0
  DO 10 K=NW,1,-1
  IF(LOCEND(K).NE.0) GO TO 10
  IDSCA=IDSCA+1
  LOCEND(K)=IDSCA
  KS=K
  XS=WX(K)
  YS=WY(K)
  IF(KS.EQ.1) GO TO 10
  DO 13 KK=KS-1,1,-1
  IF(LOCEND(KK).NE.0) GO TO 13
  IF(ABS(XS–WX(KK)).LT.DISPIK.AND. ABS(YS–WY(KK)).LT.DISPIK)
&LOCEND(KK)=IDSCA
13 CONTINUE
10 CONTINUE
NPIK=IDSCA
DO 25 KID=1,NPIK
KFIND=0
DO 35 K=1,NW,-1
IF(KFIND.EQ.1.OR.LOCEND(K).NE.KID) GO TO 35
KFIND=1
LOCPIK(KID)=LOCBEG(K)
35 CONTINUE
25 CONTINUE
IF(IPRT.EQ.1)PRINT*,LSTPIK- END: KPIK, LOCPIK, LOCEEG
DO 37 K=1,NPIK
IF(IPRT.EQ.1)WPITE(*,4) K,LOCPIK(K),LOCBEG(K)
37 CONTINUE
4 FORMAT(1X,LSTPIK- END,3(2XI5))
RETURN
END

Subroutine MAPWIN.FOR

SUBROUTINE MAPWIN(WPAR,IWIN,WCNWIN,WMNWIN,XFPT,YFPT,MNXXMAP,MNXYMAP,
&NXMAP,NYMAP)
REAL WPAR(8),WCNWIN(2),WMNWIN(2),XFPT,YFPT,XCEN,YCEN
INTEGER IWIN,K,NXMAP,NYMAP,MXXMAP,MXYMAP
C IWIN=(1,2,3,4):1=no-reset,2=center WCNWIN,3=min-corner WMNWIN,4=  
C center (XFPT,YFPT); Size of window is:(WPAR7,WPAR8)**************
GO TO (1,2,3,4),IWIN
1 DO 9 K=3,4
9 WPAR(K)=WPAR(1)*FLOAT(INT(WPAR(K)/WPAR(1)+0.5))
PRINT*,MAPWIN-1:readjust WPAR3-4 to int# DELMAP
GO TO 14
2 DO 7 K=1,2
7 WPAR(K+2)=WPAR(1)*FLOAT(INT((WCNWIN(K)-WPAR(K+6)*.5)/WPAR(1)+0.5))
PRINT*,MAPWIN-2:compute WPAR3-4 from center WCNWIN
GO TO 14
3 DO 6 K=1,2
6 WPAR(K+2)=WPAR(1)*FLOAT(INT((WMNWIN(K)/WPAR(1)+0.5))
PRINT*,MAPWIN-3:compute WPAR3-4 from min-corner WMNWIN
GO TO 14
4 WCNWIN(1)=XFPT
WCNWIN(2)=YFPT
DO 11 K=1,2
11 WPAR(K+2)=WPAR(1)*FLOAT(INT((WCNWIN(K)-WPAR(K+6)*.5)/WPAR(1)+0.5))
PRINT*,MAPWIN-4:compute WPAR3-4 from center (XFPT,YFPT)
CONTINUE
NXMAP=INT(0.5+WPAR(7)/WPAR(1))
NYMAP=INT(0.5+WPAR(8)/WPAR(1))
IF(NXMAP.GT.MXMAP.OR.NYMAP.GT.MYMAP)STOP 'size window GT dimension of SYNMAP & SUMAR

XCEN=WPAR(3)+0.5*FLOAT(NXMAP)*WPAR(1)
YCEN=WPAR(4)+0.5*FLOAT(NYMAP)*WPAR(1)
WPAR(5)=WPAR(3)+FLOAT(NXMAP)*WPAR(1)
WPAR(6)=WPAR(4)+FLOAT(NYMAP)*WPAR(1)
WPAR(7)=WPAR(5)-WPAR(3)
WPAR(8)=WPAR(6)-WPAR(4)
PRINT*, 'MAPWIN: XMN, YMN, XMX, YMX, XSIZE, YSIZE, XCEN, YCEN, NXMAP, NYMAP'
WRITE(*,20)(WPAR(K),K=3,8),XCEN,YCEN,NXMAP,NYMAP
20 FORMAT(8(1XF8.2),2(lXI3))
RETURN
END

Subroutine SOR3RL FOR

SUBROUTINE SOR3RL(VAL, VX, VY, LOC, N)
REAL VAL(1), VX(1), VY(1), VALS, VXS, VYS
INTEGER LOC(1), N, LOCS, K, M, I, II
7 M=N
1 CONTINUE
M=M/2
IF(M.EQ.0) RETURN
K=N-M
J=1
2 CONTINUE
I=J
3 CONTINUE
II=I+M
IF(VAL(I).LT.VAL(II)) GO TO 4
VALS=VAL(I)
VXS=VX(I)
VYS=VY(I)
LOCS=LOC(I)
VAL(I)=VAL(II)
VX(I)=VX(II)
VY(I)=VY(II)
LOC(I)=LOC(II)
VAL(II)=VALS
VX(II)=VXS
VY(II)=VYS
LOCS=LOCS
I=I-M
IF(I.GE.1) GO TO 3
4 CONTINUE
J=J+1
IF(J.GT.K) GO TO 1
GO TO 2

61
Subroutine SURFMP.FOR

SUBROUTINE SURFMP(KFRQ,KRAN,Z,JRANMX,IN123)
INTEGER MNIW,MNJW,MXIW,MXJW,KFRQ,KRAN,IN123,IW,JW,NPI-NFP,JRANM.X
REAL ZMX,SYNMAP,Z
COMMON /SURFP/ SYNMAP( 128,200),MNIW,MNJW,MXIW,MXJW,ZMX,NPINFP
GO TO (1,2,3), IN123
1 MNIW=999
MNJW=999
MXIW=-999
MXJW=-999
ZMX=0.
NPINFP=0
DO 5 IW=1,128
DO 5 JW=1,200
5 SYNMAP(IW,JW)=0.
RETURN
2 NPINFP=NPINFP+1
MNIW=1
MNJW=1
MXIW=128
MXJW=JRANMX
ZMX=A MAX( ZMX,Z)
SYNMAP(KFRQ,KRAN)=Z
RETURN
3 OPEN(UNIT=7,FILE='DOPFP.GRD',STATUS='OLD')
WRITE(7,'(DSAA)')
WRITE(7,'(I5,1X,I5)')MXIW-MNIW+1,MXJW-MNJW+1
WRITE(7,'(E12.5,1X,E12.5)')FLOAT(MNIW),FLOAT(MXIW)
WRITE(7,'(E12.5,1X,E12.5)')FLOAT(MNJW),FLOAT(MXJW)
WRITE(7,'(E12.5,1X,E12.5)')0.,ZMX
DO 69 JW=MNJW,MXJW
WRITE(7,'()') (SYNMAP(IW,JW),IW=MNIW,MXIW)
WRITE(7,'()')
69 CONTINUE
62 FORMAT(10(1XF8.3))
CLOSE(7)
STOP
END

Subroutine WINPUX.FOR
SUBROUTINE WINPUX(NW,W,IDFILE,TITLE)
  CHARACTER*240 TITLE
  CHARACTER*55 WCHAR(51)
  REAL W(1)
  INTEGER KW,K
  REWIND IDFILE
  DO 7 KW=1,NW
  7 READ(IDFILE,11) W(KW),WCHAR(KW)
  11 FORMAT(F16.7,A55)
  PRINT*, 'ARRAY W BEFORE UPDATE'
  WRITE(*,12) TITLE
  WRITE(2) (W(K),K=1,NW)
  2 FOR.MAT(4( 2XF16.7))
  CALL WCHANGE (W)
  REWIND IDFILE
  DO 8 KW=1,NW
  8 WRITE(IDFILE,'(F16.7,A55)') W(KW),WCHAR(KW)
  REWIND IDFILE
  DO 17 KW=1,NW
  17 READ IDFILE,11) W(KW),WCHAR(KW)
  PRINT*, 'ARRAY W AFTER UPDATE'
  WRITE(*,2) (W(K),K=1,NW)
  RETURN
END

Function YMU3DB.FOR

FUNCTION YMU3DB(HTHETA,BETQ9,BETSCA,IPRT)
  REAL YMU3DB,BETQ9,BETSCA,HTHETA
  DOUBLE PRECISION COSMU,HTHETD,BETQ9D,BETSCD
  HTHETD=HTHETA
  BETQ9D=BETQ9
  BETSCD=BETSCA
  COSMU=(DCOS(HTHETD)-DCOS(BETQ9D)*DCOS(BETSCD))/(DSIN(BETQ9D)*
&DSSN(BETSCD))
  IF(DABS(COSMU).GT.1.D0) GO TO 2
  YMU3DB=DACOS(COSMU-.000000001D0)
  IF(IPRT.EQ.0)RETURN
  2 PRINT*,HTHETA, BETQ9, BETSCA, COSMU, YMU3DB
  PRINT*,HTHETA, BETQ9, BETSCA, COSMU, YMU3DB
  IF(DABS(COSMU).GT.1.D0) STOP 'ABS(COSMU) > 1.'
  RETURN
END
Appendix D
Listings for DOPXY.FOR, MAPAZD.FOR, and MAPRGD.FOR

Program DOPXY.FOR

C CREATED 02 26 9 2300hr
CHARACTER*240 TITLE
REAL W(23),TIMCNT,TIMDWL,DELTRAN,XMMNPR,WPAR(8),XIFXMT,SELWIN,
&DELPTR,SELDIS,BITINT,WSQ(12),SHOSCA,BUGLEV,DUM1,DUM2,WINIWR
REAL SMOMAP(450),TXD(64),TXQ(64),WANG9(265),WAGNCL(128),
&WBET9(265),WDOM3(3),WDQVL3(3),WEPSMX(265),WIND(64)
REAL VELOM,ALFM,BETM,WMPOSB(3),WMPOSE(3),WQPOSB(3),
&WQPOS(6),WQPOSE(3),WFREQ(128),WMVELO(6),WWAPEX(128,9)
REAL WMU3DB(265),WMUMDR(265),WRAN9(265),WTAU(25),WTIME(64),
&WTSMO(5),XD(128),XM(128),XQ(128),WADRL0(265),WADRUP(265),
&WADSNO(90),WAGAIN(3),HEIREF,BETREF,RANREF
INTEGER LNPPP(128),NBSCA,KFPDIS,NW,LEVBUG,NBFP,NBSCIN,JRSCMN,
&JRSCMX,IWZMX,JWZMX,KFPSGN,ISURF,IPRDET,NBINCIJ,NBINOL,IWMX,JWMX
REAL DISH,DOTCOS,FRQOBU,FREQ,COSANT,GRDMAP,TMU3DB,
&FPmxNB,RELDOP,BET,SURFIN
REAL CLIGHT,DELFIN,DELFIN,DOVL,DOVL,DOVL,DOVL,DOVL,DOVL,DOVL,DOVL,
&GDISH3,GRCHDELRTA,HHTHETA,HTIDMDW,RCNOF,OMEGO,SCAFAC,
&SINANG,THTHETA,TPBET,TPI,V1,V2,V3,X,XSCA,XSQA,Y,Z,FZMX,SCANEW
INTEGER IDDISH,IFILE,INDEX,INTCOF,IP,JFSAK,NBFP,NBFP
EXTERNAL YMU3DB,FRTOMU
REAL SY.NMAP(0:137,0:250),SUMARA(0:137,0:250)
COMMON /GRDMAP/SYNMAP,STJMAPARA
INTEGER NBSCDT,IWFPT,IWRAN,IWFRQ,JWFPT,JWRAN,JWFRQ
REAL WSCDTX,WSCDTX,WSCDTX,WSCDTX,WSCDTX,WSCDTX,WSCDTX,WSCDTX,WSCDTX,
&IWFTP(400),IWRAN(400),IWFRQ400),NBINCU,XV(129),VY(129)
&VM129),JWRAN(129),JWFRQ(129),JWFTP(129),NBINOL
EQUIVALENCE (W1,TIMDWL),(W2,DELTRAN),(W3,XMMNPR),(W4,
&WPAR1),(W12,XIFXMT),(W13,SELWIN),(W14,DELPTR),(W15,
&SELDIS),(W16,WINIWR),(W17,SCANEW),(W18,HEIREF),
&(W19,BETREF),(W20,SHOSCA),(W21,BUGLEV),(W22,FPmxNB),
&(W23,SURFIN)
C PARAMETERS FOR OUTPUT MAP ARE IN WPAR=(DELMAP,THRESH,XMMNPR,YNMAP
C,XMMAP,YMMAP,XYSIZE,YSIZE)**********************************************
C WMVELO=(XVELO,YVELO,ZVELO,VELOM,2/WAVLNGH),DELFUT)***************
C WMPOSC=(XM,YM,ZM,HTHETA,RANMN,RANMX)*------------------------------
C WQPOSC=(XQF,YQP,YQF,ZQF,ALFQF,BETQF,RANQF)*************************
C DIMENSION OF SMOMAP = NBSCA*15 (SEE SUB. CLUFPT)**********************
FRQOFMU(V1,FMU,V2,V3)=V1*(V2*FMU+(V3/2.)*FMU**2)

64
C I/O FILES: U9=SYPAI.INP(main);U3=DOPFP.PRT(main);U11=CLMPI.INP
C (clumap);U10=FPSQI.INP(made in fpidop, used in bcedop);U12=GEODOPI.
C &INP(fpidop);U7=DOPFP.GRD(main)  
C HALF 3db BEAMWIDTH=.0215 gives DOVL=14.83343168=DISH DIAMETER IN
C &LAMBDA UNITS (2-WAY GAIN = -3.00 DB means DISH=0.84139514)*****
C DATA THETA,FREQO,CLIGHT,TPI,EPS,WDOVL,NW,BITINT/,.043,94.E9,3.E8,
&6.28318531,.0001,17.15094,17.15094,14.80884,23.1,/
C DATA WQPOSB(3),WQPOSC(3),WQPOSE(3)/0.,0.,0./
C NBSCDT=0
C SINANG=SIN(0.0215)
C GDISH2=DISH(SINANG,2,WDOVL(2))
C GDISH3=DISH(SINANG,3,WDOVL(3))
C OPEN(UNIT=3,FILE='DOPFP.PRT',STATUS='OLD')
C WRITE(3,*)IGDISH2,GDISH3*,GDISH2,GDISH3
C PRINT*,'WINIPX: 0. window by WPAR, 1. window centered at
&(XFP,YFP) of footprint KFP=KFPDIS'
C TITLE='TIMDWL= 1;DELRAN=2;XMNMNPR=3;WPAR1=4=DELMAP;WPAR2=THRESH;WPAR
&3=XMN;WPAR4=YMN;WPAR5=XM/;WPAR6=YM/X;WPAR7=XSIZE;WPAR8=YSIZE;XIFX
&=12;SELWIN=13;DETPR=14;SELDIS=15;WINIPX=16;SCANW=17;HEIREF=18;BE
&TREF=19;SHOSCA=20;BUGLEV=21;FPMXN=22;SURFIN=23'
C OPEN(UNIT=9,FILE='SYPAI.INP',STATUS='OLD')
C CALL WINIPX(NW,W,9,TITLE)
C PRINT*,'WINIPX specify (1,2,3,4)&(7,8), then (5,6) are computed,'
C PRINT*,'in MAPWIN last 2--arguments are DIM of SYNMAP,DUM=dummy'
C CALL MAPWIN(WPAR,1,WDUM,WDUM,DUM1,DUM2,137,250,NXMAP,NYMAP)
C IWRWIN=INT(SURFIN+EPS)
C IDDISH=INT(SELDIS+.00001)
C LEVBUG=INT(BUGLEV+.00001)
C ISURF=INT(SURFIN+EPS)
C IPRDET=INT(DETPR+EPS)
C NBFP=INT(FPMXN+BEPS)
C WDOVL=WDOVL(IDDISH)
C OMEG O=TPI*FREQO
C HDELRA=DELRA*0.5
C HTHETA=THETA*0.5
C WMPOSE(4)=HTHETA
C NBITFF=6
C NBITFX=NBITFF+INT(BITINT+EPS)
C NFFT=2**NBITFF
C NFFTH=NFFT/2
C NFFX=2**NBITFX
C NFFXH=NFFX/2
C DELFIN=1./TIMDWL
C INTCOF=2**INT(BITINT+EPS)
C DELFUT=DELFIN/FLOAT(INTCOF)
C WMVELO(6)=DELFUT
C DELTIM=TIMDWL/NFFT
C HTMDW=TIMDWL*.5
C COMPUTE RANGE REFERENCE TO USE IN RANCOF **********************
C RANREF=HEIREF/COS(BETREF)
C CALL CLUMAP(HTHETA,LEVBUG)
C OPEN(UNIT=10,FILE='FPSQI.INP',STATUS='OLD')
PRINT*,'SELECT FP TO DISPLAY: KFPSGN, ABORT=-999; ALL FP=-KFPSGN'
CALL KINPUT(KFPSGN)
IF(KFPSGN.EQ.-999) STOP 'ABORT'
KFPDIS=ABS(KFPSGN)
CALL FPIDOP(NBFPMX, NBFP, HTHETA, TIMDWL, KFPDIS)
PRINT*,'**** KFPDIS=; NBFPMX=; NBFP=; KFPDIS, NBFPMX, NBFP, *****'
DO 13 IW=0,185
DO 13 JW=0,185
SUMARA(IW,JW)=0.
13 SYNMAP(IW,JW)=0.
C COMPUTE ARRAY OF RELATIVE TIME WTIME
DO 2 KTIM=1,NFFT
WTIME(KTIM)=DELTIM*(KTIM-NFFTH-0.5)
C INDEX KNPPP =1,NFFX FOR FREQ= (-NFFXH*DELFUT,(NFFXH-1)*DELFUT)
C COMPUTE ARRAY LNPPP, THEN KFF =LNPPP(KNPPP) & WFREQ(KNPPP)
DO 9 KNPPP=1,NFFX
WFREQ(KNPPP)=DELFUT*FLOAT(KNPPP-NFFXH-1)/FREQO
INDEX=KNPPP+NFFXH
IF(INDEX.GT.NFFX) INDEX=INDEX-NFFX
9 LNPPP(KNPPP)=INDEX
CALL WINDOWS(NFFT,4,3.14159,WIND)
IF(ISURF.EQ.1) CALL SURFMP(KFF,JRAN,Z,1)
DO 1 KFF=1,NBFP
CALL BCDOP(W,NFFT,WTIME,VELOM,ALFM,BETM,WMPOSB,WMPOSC,WMPOSE.
&WQPOSB,WQPOSC,WQPOSE,WMVELO,JRANQ9,JRMNX,KFPDIS,KFF)
IF(KFF.NE.KFPDIS.AND.KFPSGN.GT.0) GO TO 1
WRITE(*,'(**)') VELOM, ALFM, BETM, XM, YM, ZM, XFP, YFP'
WRITE(*,'(**)') VELOM, ALFM, BETM, (WMPOSC(K),K=1,3),(WQPOSC(IW),IW=1,2)
WMVELO(5)=2.*FREQO/CLIGHT
IF(KFF.EQ.KFPDIS.AND.ABS(WINIW-1.).LT.EPS) PRINT*, 'window centered
& at (XFP, YFP)
IF(KFF.EQ.KFPDIS.AND.ABS(WINIW-1.).LT.EPS) CALL MAPWIN(WPAR,4,
&WDOM, WDOM, WQPOSC(1), WQPOSC(2), 137, 250, NXMAP, NYMAP)
C STORE RANGE, BET, EPSMX, AND MU-3DB FOR FICTITIOUS SCATTERERS
DO 7 JRAN=5,JRANMX-9
WRAN9(JRAN)=WQPOSC(6)+(JRAN-JRANQ9)*DELWAN
WBET9(JRAN)=ACOS(WMPOSC(3)/WRAN9(JRAN))
TMPBET=ACOS(WMPOSC(3)/(WRAN9(JRAN)-HDELRA))
WFPSM9(JRAN)=WBET9(JRAN)-TMPBET
WMU3DB(JRAN)=WMU3DB(HTHETA,WQPOSC(5),WBET9(JRAN),0)
7 THIS IN PREPARATION FOR MAPPING
C ENTER CLUTTER INFORMATION FOR FOOTPRINT AT WQPOSC & MAP PARAMETERS
WRITE(3,'(*)') JRANMX=, JRANMX
CALL CLUFPT(DERAN,JRANQ9,JRMNX,WQPOS,WMPOS,NSCAFP,SMOMAP,
&WADRLO,WADRUP,WADMOS)
C PROCESS RANGE CELLS SEQUENTIALLY AND COMPUTE REFLECTIVITY MAP
DO 35 JRAN=5,JRANMX-9
KSCALO=WADRLO(JRAN)+EPS
IF(KSCALO.EQ.0) GO TO 35
C INITIALIZING QUADRATURE ARRAYS=(XD,XQ)
DO 11 K=1,NFFX
XD(K) = 0.
11 XQ(K) = 0.
WDUM(3) = 0.
C COMPUTE TAYLOR COEFFICIENTS WTAU=(TAU,TAUt,TAUe,TAUtt,TAUee,TAUtm,
C TAUte,TAUttt,TAUtte) ************
CALL TAYLOR(VELOM,BETM,ALFM,WQPOSC(4),WQPOSC(5),WMPOSC(3),HTHETA,
&DELTRAN,JRAN,WTAU)
WRITE(3,*)'ARRAY WTAU'
CALL PRTXY(WTAU,1,25,119,3)
C PREPARATION FOR MAPPING: COMPUTE (NPPPMN,NPPPMX) ON 3DB FOOTPRINT
C TO LIMIT # CELLS BEING MAPPED TO KNPPP=NPPPMN,NPPPMX **************
TMU3DB=SIGN(WMU3DB(JRAN),WQPOSC(4))
NPPPMN=FRQOFMU(FREQO,TMU3DB,WTAU(6),WTAU(10))/DELFUT+NFFXH-5.
NPPPMX=FRQOFMU(FREQO,-TMU3DB,WTAU(6),WTAU(10))/DELFUT+NFFXH+5.
IF(ABS(WQPOSC(4)).LT.WMU3DB(JRAN)) NPPPMX=FRQOFMU(FREQO,-WQPOSC(4)
&.WTAU(6),WTAU(10))/DELFUT+NFFXH+5.
C COMPUTE MU AT CENTER OF CELLS FOR INCREASING DOPPLER **************
DO 17 K=NPPPMN-1,NPPPMX+1
17 WMUMDR(K)=SIGN(1.,WQPOSC(4))*FRMTUMU(WTAU,WFREQ(K))
C COEFFICIENT FOR RANGE NORMALIZATION: RANCOF ***************
RANCOF=(WRA9(JRAN))/RANREF)**4
KSCAUP=WADRUP(JRAN)+EPS
DO 30 KTHSCA=KSCALO,KSCAUP
KADSMO=WADSMO(KADSMO)+EPS
C INFORMATION ON SCATTERER,WTSMO=(AMPSCA,FASSCA,MUSCA,JRAN-SCA.ZSCA)
DO 31 KT=1,5
31 WTSMO(KT)=SMOMNAP(KADSMO-4+KT)
C ANTGAIN(t)=WAGAIN(1)+WAGAIN(2)*t+WAGAIN(3)*t-2 ***************
C WAGAIN IS OUTPUT OF SCAGAN, INPUTS ARE DOVL,HTIMDW,WMPOS=(BEG.
C &CENT,END),WQPOSC,WTSMO=(AMP,FAS, MU,JRAN,ZSCA),JRANQ9,DELRAN
CALL SCAGAN(IDDISH,DOVL,HTIMDW,WMPOSB,WMPOSC,WMPOSE,WQPOSE,WTSMO,JRANQ9,DELRAN
&.WAGAIN)
CALL TSIGNAL(OMEGO,NFFT,RANCOF,XIFXMT,WTSMO,WTAU,WTIME,WAGAIN,TXD.
& TXQ)
1 DO 33 KDQ=1,NFFT
33 XQ(KDQ)=XQ(KDQ)+TXQ(KDQ)
33 CONTINUE
C COMPUTE ARRAY OF ANTENNA GAIN AT CENTER OF CELLS ********************
DO 39 KNPPP=NPPPMN,NPPPMX
WNANG=DSQRT(1.+EPS-(DSIN(WQPOSC(5))^2+DSIN(WBET9(JRAN))^2)*DCOS(WMUMDR
&.KNPPP)+DCOS(WQPOSC(5))^2*DCOS(WBET9(JRAN))^2)**2
39 WANGCL(KNPPP)=DSIN(WNANG,IDDISH,DOVL)**2
C WWAPEX NEED TO BE SCALLED FOR RANGE AND ANTENNA GAIN AT KNPPP ****
C APEXTRAP COMPUTES WWAPEX AT CENTERS OF CELLS,*****************
C WWAPEX=(KNPPP,X1,Y1,X2,Y2,X3,Y3,X4,Y4,AREA) ********************
CALL APEXTRAP(NFFX,WBET9(JRAN),DELTRAN,WQPOSC(4),WEPMSX(JRAN),
&NPPPMN,NPPPMX,WTAU,WMPOSC,WMUMDR,WWAPEX)
67
DO 41 KNPPP=NPPPMN,NPPPMX
KFF=LNPPP(KNPPP)
SCAFAC=1./(RANCOF*WAGNCL(KNPPP))
XD(KFF)=XD(KFF)*SCAFAC
XQ(KFF)=XQ(KFF)*SCAFAC
41 XM(KFF)=SQRT(XD(KFF)**2+XQ(KFF)**2)
C FPT CELLS ARE MAPPED EXCEPT (BELOW THRESH or. OUT FOOTPR) KNPPP **
C GOES:(NPPPMN,NPPPMX) ie:(MUMX,MUMN); FOR FFT USE KFF=LNPPP(KNPPP) *
CALL MAPRSD(NXMAP,NYMAP,NPPPMN,NPPPMX,XM,WPAR,LNPPP,WWAPEX)
35 CONTINUE
1 CONTINUE
PRINT*,NXMAP,NYMAP,NXMAP,NYMAP
FZMX=0.
DO 60 IW=0,NXMAP-1
DO 60 JW=0,NYMAP-1
IF(SUMARLA(IW,JW).LT.EPS) GO TO 60
SYNMAP(IW,JW)=SYNMAP(IW,JW)/SUMARA(IW,JW)
IF(SYNMAP(IW,JW).LT.FZMX) GO TO 60
FZMX=SYNMAP(IW,JW)
IWMX=IW
JWMX=JW
60 CONTINUE
PRINT*,FZMX=; IWMX=; JWMX=; ',FZMX,IWMX,JWMX
OPEN(UNIT=7,FILE='DOPXYG.GRD',STATUS='OLD')
WRITE(7,14)NXMAP,NYMAP
WRITE(7,14)NXMAP,NYMAP
WRITE(7,14)NXMAP,NYMAP
WRITE(7,14)NXMAP,NYMAP
WRITE(7,14)NXMAP,NYMAP
WRITE(7,62) (SYNMAP(IW,JW),IW=0,NXMAP-1)
WRITE(7,'(_)')
69 CONTINUE
62 FORMAT(10(1XF8.3))
18 FORMAT(8F10.3)
CLOSE(3)
CLOSE(7)
CLOSE(9)
CLOSE(10)
STOP
END

Subroutine MAPAZD.FOR

SUBROUTINE MAPAZD(NXMAP,NYMAP,XMVAL,WAX,WAY,WPAR)
REAL WAX(1),WAY(1),WPAR(6),WWAREA(0:31,0:31)
REAL SYNMAP(0:137,0:250),SUMARA(0:137,0:250)
REAL AREPOL
INTEGER I,IIO,IFAIL,IMX,IW,J,JJO,JMX,JW

...
COMMON /GRDMAP/SYNMAP,SUMARA
CALL POLONFRA(4,WAX,WAY,WPAR,II0,JJO,IMX,JMX,AREPOL,WWAREA,0,
 &IFAIL)
C 3--SYSTEMS OF COORDINATES: WPAR define ABS--location of window:****
C (IW,JW)=WINDOW--coordinates ; (I,J)=FRAME--coordinates ***********
DO 43 I=0,IMX-1
DO 43 J=0,JMX-1
IF(WWAREA(I,J).LT.0.0002) GO TO 43
IW=II0+I
JW=JJO+J
IF(IW.LT.0.OR.JW.LT.0) GO TO 43
IF(IW.GT.NXMAP.OR.JW.GT.NYMAP) GO TO 43
SYNMAP(IW,JW)=SYNMAP(IW,JW)+WWAREA(I,J)*XMVAL
SUMARA(IW,JW)=SUMARA(IW,JW)+WWAREA(I,J)
43 CONTINUE
RETURN
END

Subroutine MAPRGD.FOR

SUBROUTINE MAPRGD(NXMAP,NYMAP,NPPPMN,NPPPMX,XM,WPAR,LNPPP,WWAPEX)
REAL WPAR(6),WWAREA(0:15,0:15) WAX(4),WAY(4),WWAPEX(128,9),
&XM(1),WAXMN,WAXMX,WAYMN,WAYMX
INTEGER KA,KAPX,KFF,KNPPP,KMNMX,LNPPP(l)
DO 45 KNPPP=NPPPMN,NPPPMX
KFF=LNPPP(KNPPP)
IF(XM(KFF).LT.WPAR(2)) GO TO 45
KA=0
DO 40 KAPX=1,7,2
KA=KA+1
WAX(KA)=WWAPEX(KNPPP,KAPX)
40 WAY(KA)=WWAPEX(KNPPP,KAPX+1)
C CALL POLONFRA ONLY IF TRAPEZOIDS ARE WITHIN THE SPECIFIED WINDOW *
WAXMN=WAX(1)
WAYMN=WAY(1)
WAXMX=WAX(1)
WAYMX=WAY(1)
DO 5 KMNMX=1,4
WAXMN=AMIN1(WAXMN,WAX(KMNMX))
WAXMX=AMAX1(WAXMX,WAX(KMNMX))
WAYMN=AMIN1(WAYMN,WAY(KMNMX))
WAYMX=AMAX1(WAYMX,WAY(KMNMX))
5 IF(WAXMN.LE.WPAR(3).OR.WAXMX.GE.WPAR(5)) GO TO 45
IF(WAYMN.LE.WPAR(4).OR.WAYMX.GE.WPAR(6)) GO TO 45
IF(XM(KFF).GE.10.) CALL MAPAZD(NXMAP,NYMAP,XM(KFF),WAX,WAY,WPAR)
45 CONTINUE
RETURN
END

69/(70 blank)
<table>
<thead>
<tr>
<th>Organization</th>
<th>No. of Copies</th>
</tr>
</thead>
<tbody>
<tr>
<td>U.S. Army Materiel System Analysis Activity</td>
<td>1</td>
</tr>
<tr>
<td>ATTN: AMXSY-MP (erbert Cohen)</td>
<td>1</td>
</tr>
<tr>
<td>Aberdeen Proving Ground, MD 21005</td>
<td></td>
</tr>
<tr>
<td>IIT Research Institute</td>
<td>1</td>
</tr>
<tr>
<td>ATTN: GACIAC</td>
<td></td>
</tr>
<tr>
<td>10 W. 35th Street</td>
<td></td>
</tr>
<tr>
<td>Chicago, IL 60616</td>
<td></td>
</tr>
<tr>
<td>AMSMI-RD</td>
<td>1</td>
</tr>
<tr>
<td>AMSMI-RD-CS-R</td>
<td>15</td>
</tr>
<tr>
<td>AMSMI-RD-CS-T</td>
<td>1</td>
</tr>
<tr>
<td>AMSMI-RD-AS</td>
<td>1</td>
</tr>
<tr>
<td>AMSMI-RD-AS, A.H. GREEN</td>
<td>4</td>
</tr>
<tr>
<td>AMSMI-GC-IP, Mr. Fred Bush</td>
<td>1</td>
</tr>
</tbody>
</table>