AFRL-SN-RS-TR-1999-48, Vol II (of two)
Final Technical Report
May 1999

MCARM/STAP DATA ANALYSIS

Research Associates for Defense Conversion, Inc.

Braham Himed

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

AIR FORCE RESEARCH LABORATORY
SENSORS DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK

This report has been reviewed by the Air Force Research Laboratory, Information Directorate, Public Affairs Office (IFOIPA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

AFRL-IF-RS-TR-1999-48, Vol II (of two) has been reviewed and is approved for publication.

APPROVED: Todd B. Hale

TODD B. HALE, Capt, USAF
Project Engineer

FOR THE DIRECTOR: Robert G. Polce

ROBERT G. POLCE, Acting Chief
Rome Operations Office
Sensors Directorate

If your address has changed or if you wish to be removed from the Air Force Research Laboratory Rome Research Site mailing list, or if the addressee is no longer employed by your organization, please notify AFRL/SNRT, 26 Electronic Parkway, Rome, NY 13441-4514. This will assist us in maintaining a current mailing list.

Do not return copies of this report unless contractual obligations or notices on a specific document require that it be returned.

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 0704-0188*

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
May 1999

3. REPORT TYPE AND DATES COVERED
Final Apr 96 - Feb 98

4. TITLE AND SUBTITLE
MCARM/STAP DATA ANALYSIS, Volume II (of two)

5. FUNDING NUMBERS
C - F30602-96-C-0053
PE - 62702F
PR - 4506
TA - 11
WU - 2K

6. AUTHOR(S)
Braham Himed

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Research Associates for Defense Conversion, Inc.
10002 Hillside Terrace
Marcy NY 13403

8. PERFORMING ORGANIZATION REPORT NUMBER
N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Research Laboratory/SNRT (Formerly Rome Laboratory)
26 Electronics Parkway
Rome NY 13441-4514

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
AFRL-SN-RS-TR-1999-48, Volume II (of two)

11. SUPPLEMENTARY NOTES
Air Force Research Laboratory Project Engineer: Captain Todd Hale/SNRT/(315) 330-1896

12a. DISTRIBUTION AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

The purpose of this report is to assess the quality of the Multi-Channel Airborne Radar Measurements (MCARM) airborne data collected, evaluate several reduced rank Space-Time Adaptive Processing (STAP) algorithms, and provide guidance to researchers. The goal of MCARM is the development of STAP algorithms for enhanced target detection in severe environments through the use of real world multi-channel airborne data. The Rome Laboratory's MCARM program provides real data, monostatic and bistatic, for performance evaluation of existing and future STAP approaches.

This report evaluates the performance of several state of the art STAP algorithms using monostatic MCARM data. The algorithm list includes Factored Time Space (FTS), Extended FTS Approach (EFA), Cross Spectral Metric (CSM), Joint Domain Localized (JDL), Principal Components, Parametric Adaptive Matched Filter (PAMF), and others. As expected, adaptive processing results in significant detection performance improvements over conventional beamforming techniques. No single algorithm can be deemed "best" as several factors affect performance. Two primary factors are the sample support requirement for interference estimation and computational complexity. A significant finding from the work is the non-homogeneous nature of the clutter returns. This nature violates the assumption of independent, identically distributed data for interference estimation illustrating the need for Non-Homogeneity Detectors (NHDs). NHDs offer improved detection and false alarm performance through better selection of secondary data.

14. SUBJECT TERMS
MCARM, STAP, Adaptive Processing, Signal Processing, Data Processing

15. NUMBER OF PAGES
100

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	**18. SECURITY CLASSIFICATION OF THIS PAGE**	**19. SECURITY CLASSIFICATION OF ABSTRACT**	**20. LIMITATION OF ABSTRACT**
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

Table of Contents

List of Figures

List of Tables

1.0 Introduction

The second part of this report is devoted to the analysis of airborne data. It is well known that future airborne radars must detect targets in severe environments consisting of clutter and jamming signals. Space-Time Adaptive Processing (STAP) is a multi-dimensional adaptive filtering process which combines the signals from elements of an array of sensors and multiple pulses of a radar waveform, to suppress the interference (clutter, jamming and noise) and provide improved target detection. There have been several system design studies and computer simulations showing the benefits of available STAP algorithms. However, the development and performance evaluation of these algorithms has been limited due to a lack of real world multi-channel airborne data. The Rome Laboratory's Multi-Channel Airborne Radar Measurements (MCARM) program provides real data for performance evaluation of STAP approaches.

The thrust of this effort is the assessment of the quality of airborne data collected from the MCARM flight tests and evaluation of STAP algorithms. Algorithm evaluation is necessary to determine the quality of the data.

Another objective of this effort is to briefly demonstrate the impact of non-homogeneous environments on adaptive filter performance through specific examples using measured MCARM data.

This report is organized as follows. Chapter 2 gives a description and background of the MCARM program. Chapter 3 discusses methods of analyzing the data before making it available for further processing. Chapter 4 analyzes issues relevant to the MCARM database. Chapter 5 performs a comparative analysis with several STAP techniques. Chapter 6 concludes with remarks and recommendations for future work.

2.0 Background

The MCARM program collected multi-channel clutter data from an airborne platform. The Westinghouse owned BAC 1-11 was used as a platform for the L-Band radar data collection system that was built on a combination of Westinghouse internal funds and MCARM program funds. The system consists of an L-band active aperture antenna, multiple IF receivers, high dynamic range A/Ds, a high speed magnetic tape recorder, a signal/data processor, a multi-mode controller, and operator console. The data was collected at a variety of pulse repetition frequencies (PRFs) over various terrains including mountains, rural, urban, and land/sea interface clutter. Westinghouse IR&D also collected Ground Moving Target Indication (GMTI) data with and without foliage to evaluate FOPEN. Monostatic data was collected at PRFs of 7 kHz (High PRF), 2 kHz (Medium PRF), and 500 Hz (Low PRF). Bistatic data was collected at 23 kHz (High PRF) and at 313 Hz (Low PRF). Westinghouse collected the MCARM data during several Delmarva and East Coast fly-overs that terminated in Florida [1].

Eleven flight tests collected multi-channel data. They are labeled sequentially from 1 to 11. The last two, 10 and 11, were actually done on the same physical flight, but were given different names because flight 10 was the last bistatic flight and flight 11 was a monostatic land/sea interface flight. Two safety of flight tests did not collect data, although the radar was tested during the second. These two flights successfully evaluated the performance and flight envelope of the BAC 1-11 with the external MCARM radome/antenna.

Acquisition data files are numbered sequentially starting with number 1. Cycle numbers have been introduced to denote a system failure. In this case, a new cycle number is introduced and the system is restarted with acquisition number 1.

During the MCARM flight test, data was collected simultaneously from a multi-channel sub-aperture architecture and a low sidelobe sum and difference, (also known as Σ and Δ), analog beamformer. The multi-channel architecture was tested using two separate sub-aperture configurations each having twenty-two (22) degrees of freedom. The analog beamformer is useful for comparisons to adaptive beamforming as well as mainlobe plus auxiliary configurations. One of the sub-aperture configurations optimized the number of azimuthal degrees of freedom (16) as well as giving up to two (2) elevation degrees of freedom. The other configuration was used for all the bistatic data measurement.

Preliminary monostatic data reduction has shown mainbeam clutter to noise ratios over 80 dB and outstanding adaptive cancellation ratios of up to 70 dB. While preliminary results indicate up to 20 dB of residual uncancelled clutter in some of the analyses, good calibration data was collected on a ground range and in the air at the Patuxent (Pax) River test range. This calibration data showed excellent array performance and can be used to evaluate possible limitations to adaptive cancellation such as channel to channel stability and aircraft effects. This is precisely why the MCARM program was necessary as adaptive sensors are emerging to meet the challenges of future air surveillance threats. The ground range data showed up to -58 dB RMS sidelobe performance for an analog beamformer and localized closed loop adaptive nulls of -90 dB with respect to peak. This data was augmented by airborne pattern measurements, which showed -47 dB RMS sidelobes for the analog beamformer. These measurements show the degradation due to aircraft effects that STAP techniques should be able to overcome.

2.1 MCARM Database

Pre-processed airborne radar datacubes form the MCARM database. A datacube consists of J channels, N pulses, and K range cells. Each acquisition data file consists of a single coherent processing interval (CPI). Before making these data files available through the Internet, they have undergone the following processing steps:

- baseband conversion
- decimation
- digital in-phase and quadrature phase (I/Q) conversion
- pulse compression
- channel balancing.

Each processed datacube, and its associated "header" variables, are provided in MATLAB format. Header variables include transmit pointing direction, number of interpulse periods, pulse repetition frequency, aircraft platform inertial navigation data, global positioning system (GPS) data, etc. In addition, header variables have been added (e.g. PRF) to facilitate processing and maintain compatibility with CREST and RLSTAP. This information was gathered and stored in a file called Beam Steering Computer (BSC). The information extracted from the BSC log files allows one to, for example, obtain the complete flight trajectory for each flight number/flight cycle, target Doppler information (if there is any), etc. A complete description of these variables can be found in Appendix A.

As an example, let us look at the BSC log files of flight 5. In this particular case, we use data file ulog5a.dat and bwi.latlong. These are ASCII type files and include information about the whole flight profile. The following figures show the flight paths of both the BAC 1-11 and in this case the Sabreliner.

Figure 1 shows the flight path of the BAC 1-11 starting at the Baltimore-Washington International (BWI) airport. Acquisitions shown with a "+" sign have been collected.

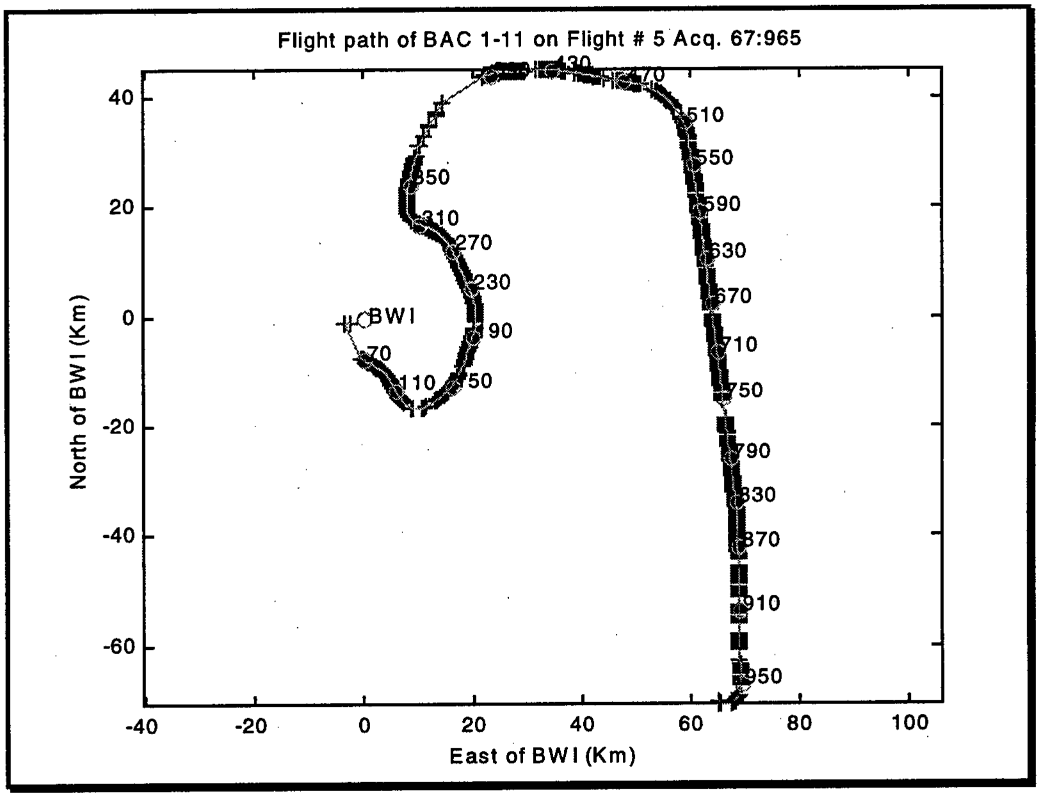

Figure 1. Flight path of BAC 1-11 on Flight 5, Acq. 67:965.

The flight path of the Sabreliner (target) is shown in Fig. 2. Figure 3 shows both the path of the BAC 1-11 and the target while Fig. 4 is similar to Fig. 3. The only difference is that at each acquisition, the dashed line connecting the two paths shows where the mainbeam hits the target's path.

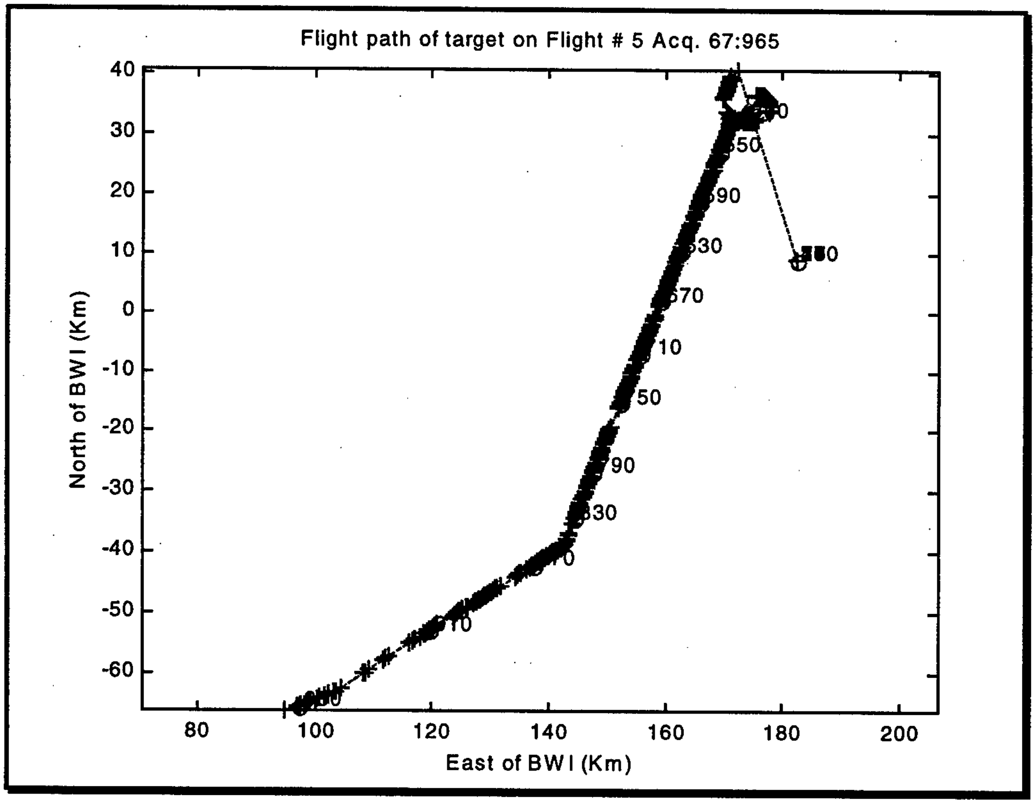

Figure 2. Flight path of target on Flight 5, Acq. 67:965.

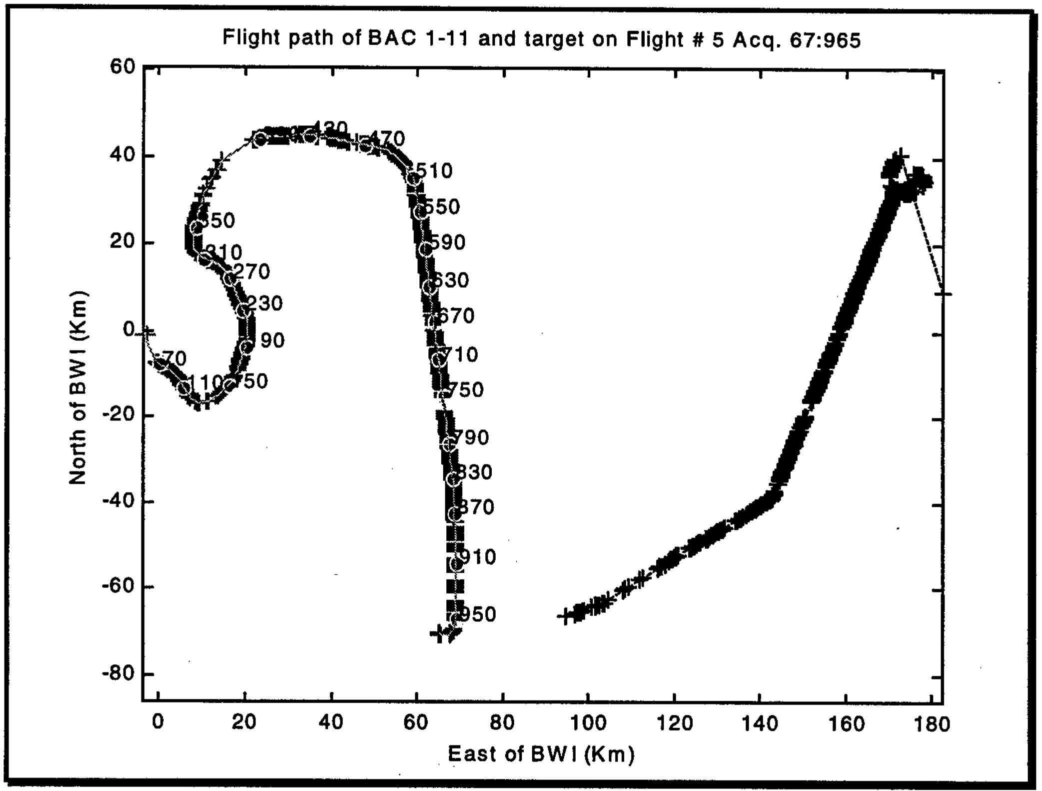

Figure 3. Flight path of BAC 1-11 and target on Flight 5, Acq. 67:965.

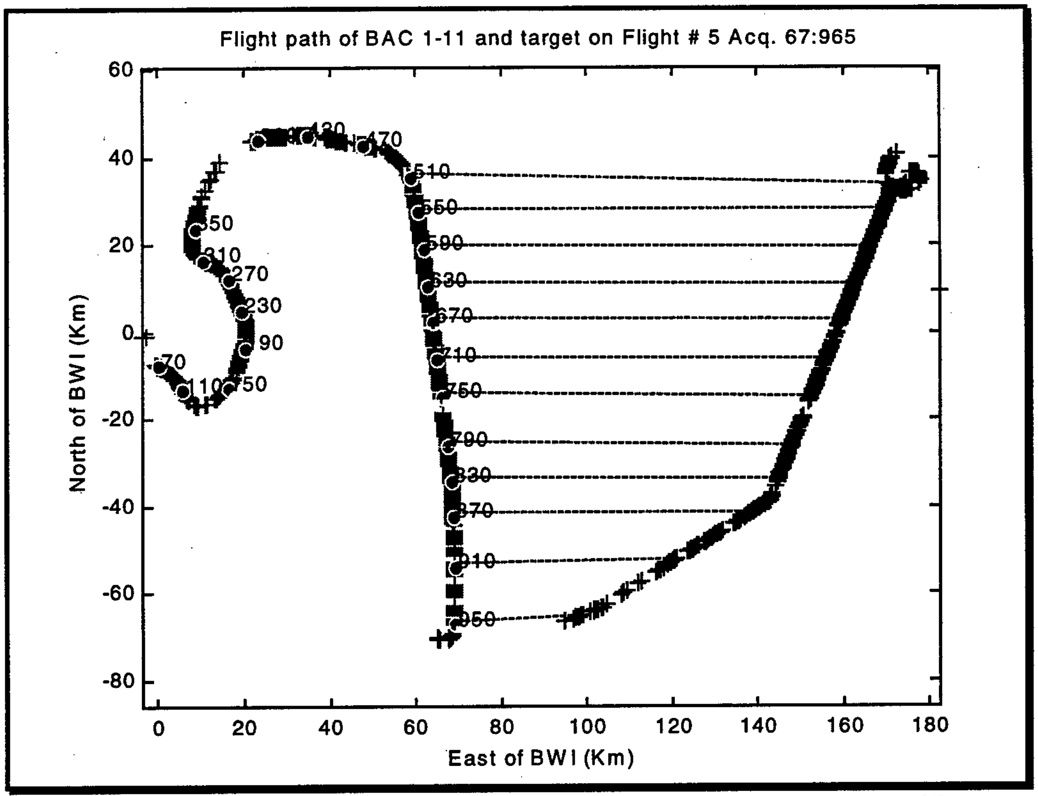

Figure 4. Flight path of BAC 1-11 and target on Flight 5, Acq. 67:965.

Similar curves could be obtained using the BSC log files for different flight numbers and different cycle numbers.

Another example could be obtained from ulog5b.dat. This corresponds to cycle "e". Figure 5 shows the flight path of the BAC 1-11 for this particular cycle.

Most of the data processed and made available through the Internet came from these two cycles.

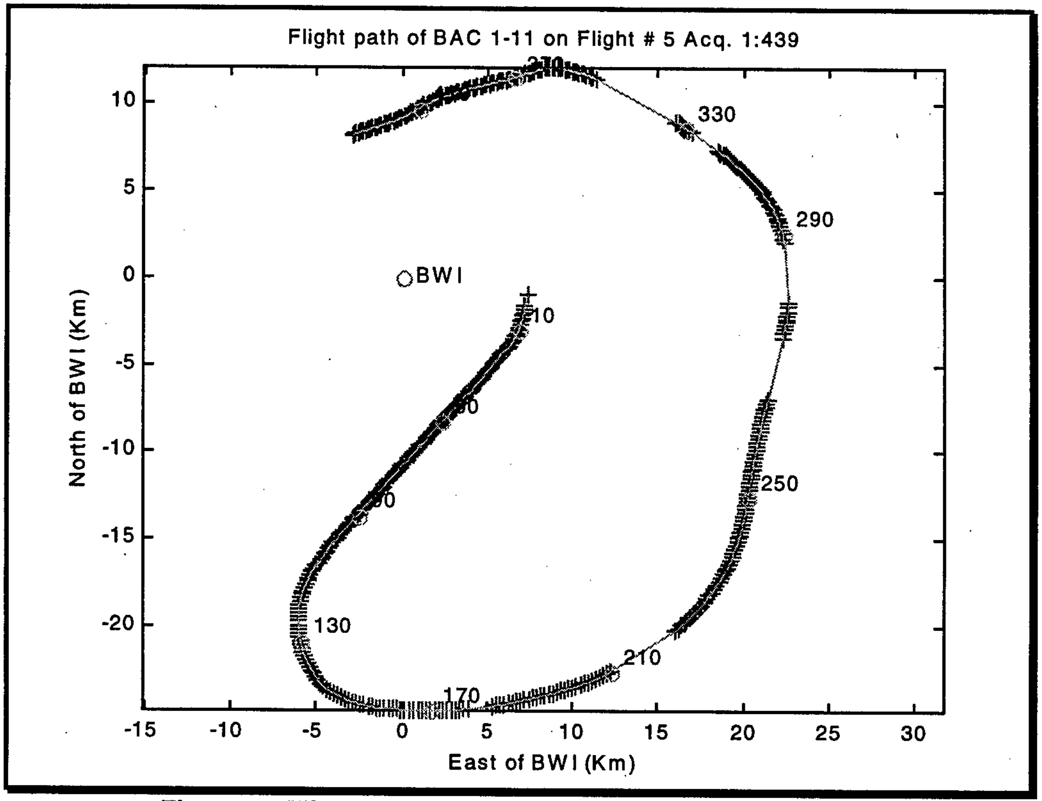

Figure 5. Flight path of BAC 1-11 on Flight 5, Acq. 1:439.

The original 'raw' data files, before the pre-processing step, are approximately 15 megabytes in size. The data files double in size after the pre-processing step is performed.

The pre-processed multidimensional measured radar data is stored in the variable named "CPI1," referred to as the datacube. The format for this datacube is shown in Fig. 6.

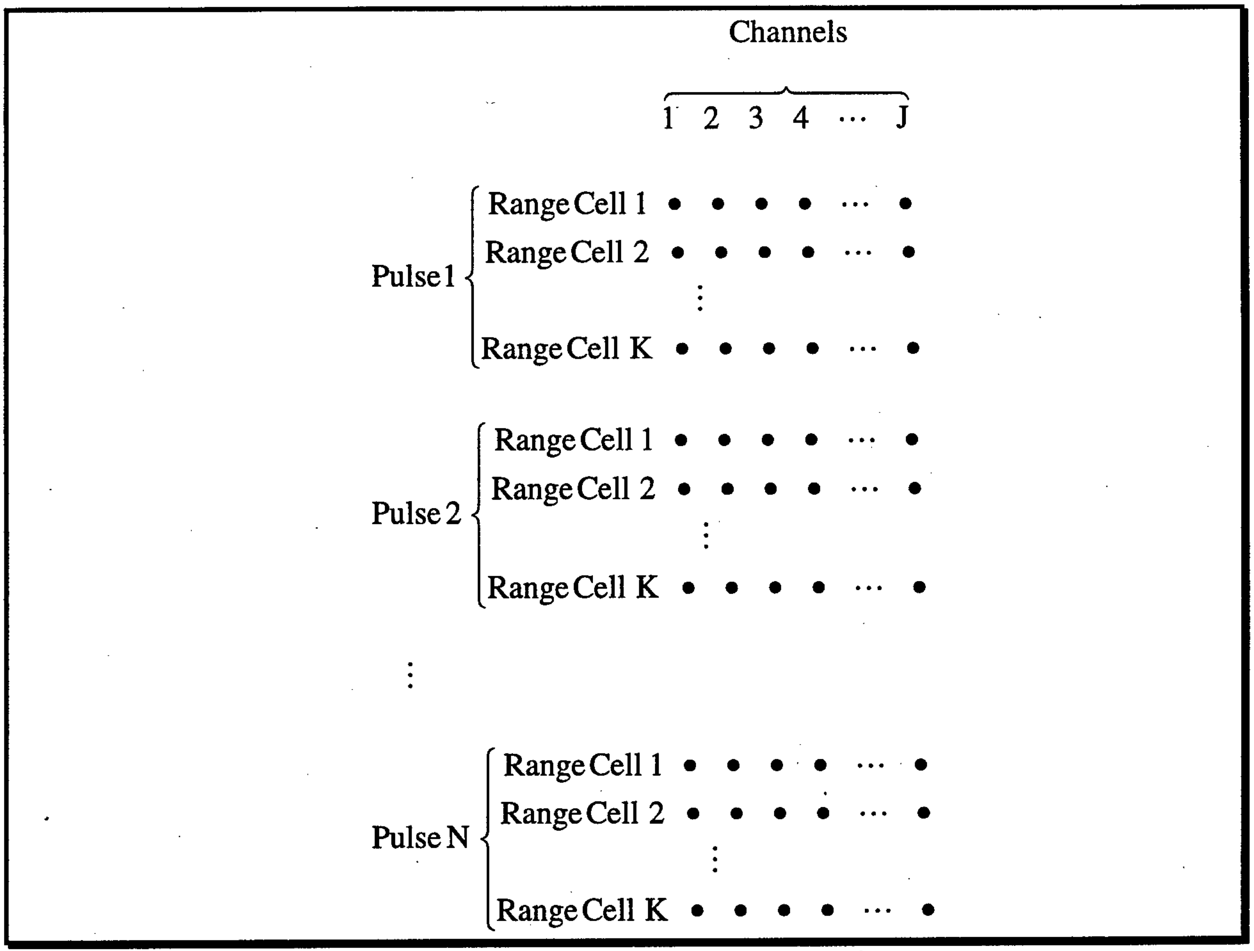

Figure 6. MATLAB format of the datacube.

This format is convenient for certain STAP processing. Other formats designed to facilitate the processing of the data for certain STAP algorithms can easily be obtained.

2.2 Technical Discussion

To assess the quality of the recorded MCARM data, the RADC team used three different techniques. These include the estimated Power Spectral Density (PSD), Eigenspectra, and Signal to Clutter plus Noise Ratio (SCNR). But first, calibration methods used prior to algorithm evaluation had to be developed.

MCARM measured data is provided in a convenient MATLAB format. This software is a good analysis environment for the researcher. In most cases, a datacube is provided. This datacube is available for each CPI being evaluated. A typical datacube consists of IF sampled data where the IF center frequency as well as the sampling frequency are varied, depending on the scenario used. Digital decimation, baseband conversion, pulse compression, and channel balancing are all performed so that a complex-valued datacube of baseband digital in-phase and quadrature phase multi-channel data is available for data processing.

When dealing with measured data, there are several error sources. Some errors come from calibration residual. It has been shown that an error of approximately 1 or 2% leads to a significant spread of the clutter power in the Angle-Doppler spectrum, which in turn can severely affect the performance of the STAP technique being used. Therefore, more research is needed to come up with robust algorithms that will be insensitive to errors of this type. It is even more desirable to have STAP techniques that not only will be robust, but will self calibrate the data. This will lead us to more pragmatic architectures as well as algorithm development.

The Factored Time-Space (FTS) [2] algorithm has successfully been applied to MCARM data previously collected [1]. It is our intent to use several STAP techniques in our evaluation of the performance of the techniques against the already collected data as well as the data to be collected later on. It is also our intent to show that these algorithms are good techniques for clutter mitigation. The analyses have been performed in MATLAB. Some analyses have been carried out using the Rome Laboratory Space-Time Adaptive Processing (RLSTAP) tool. RLSTAP has its own capabilities; however, MATLAB was used as often as needed. We generated plots showing the enhanced target detection, the clutter and jamming cancellation and CFAR capabilities.

Recall that the objective of this effort is the assessment of the quality of airborne data collected from the MCARM flight tests and evaluation of STAP algorithms. Algorithm evaluation is necessary to determine the quality of the data.

2.2.1 Data Validation

In the first phase of this effort, we obtained MCARM data, carefully studied it, and examined the data that has already been processed. This gave us a feel of what to expect for the data that will be recorded later on. Examining the data involved calibrating it, and then making a decision as to whether the data that has been calibrated earlier was properly accomplished. Our analyses showed the steps involved in the data that has already been processed prior to this effort were correct and properly coded. Only slight changes were made, but the overall procedure was correct.

2.2.2 Data Analysis

Data analysis involves doing preliminary analysis on the data such as correlating it with relevant parameters so proper headers are aligned with each data collection file. This step is considered a part of the data organization. Our objective is to organize the data in such a way that it is easily accessible to other users as well as giving enough information about the data so proper processing can be performed. Part of this organization involves describing the data depending on the scenario used in the experiment (see Part I of this report).

3.0 Data Pre-processing

RADC pre-processed the IF-sampled data, delivered under contract by Northrop-Grumman (formerly Westinghouse Electric Corporation), as depicted in Fig. 7.

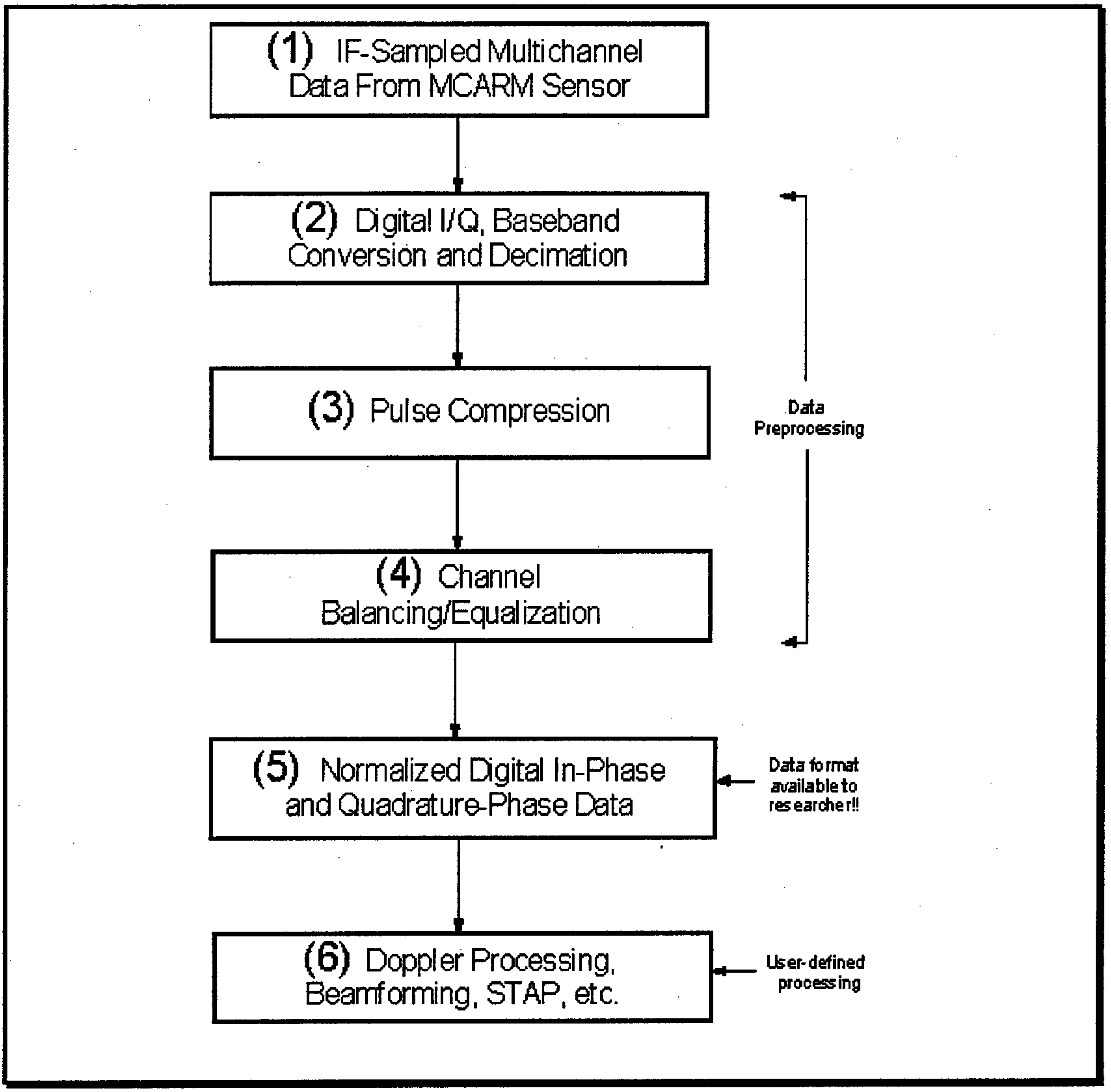

Figure 7. Pre-processing of the MCARM data.

The pre-processed data, shown in block (5), is available to researchers in the format described in the preceding section. However, to help in understanding the steps involved in this pre-processing, an example is taken from flight 5, cycle e, acquisition 152. Acquisition 152 employs a moving target simulator (MTS) in the radar field of view to simulate a moving point source. This file has data of medium PRF (PRF = 1984 Hz). The datacube consists of N = 128 pulses, J = 22 channels and L = 630 range cells. The MTS, located roughly 36 nautical miles from the airborne platform, is matched to the MCARM waveform and generates five tones of varying amplitude. Tones are spaced 200 Hz (13 Doppler bins based on a 15.6 Hz/bin Doppler resolution (PRF/128)) to simulate target velocity. The MTS and transmit azimuth are not coincident.

Figure 8 shows the output of the range profile, and its corresponding spectrum, taken at channel 1 and for pulse 1.

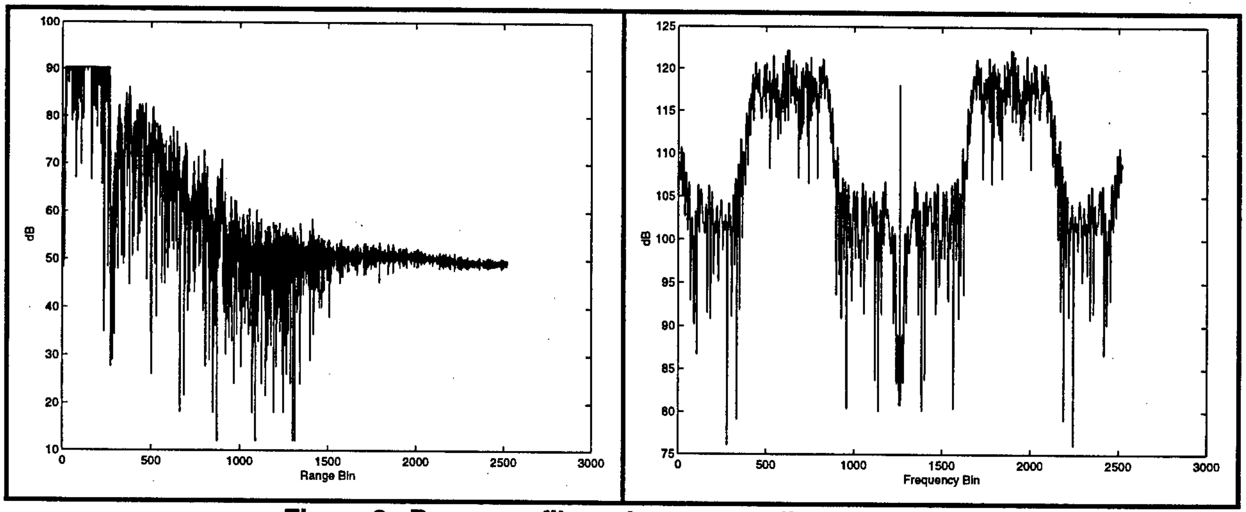

Figure 8. Range profile and corresponding spectrum.

The cutoff of the data in the time domain is due to the saturation of the A/D converters.

After decimation and baseband conversion, we obtain the following diagrams.

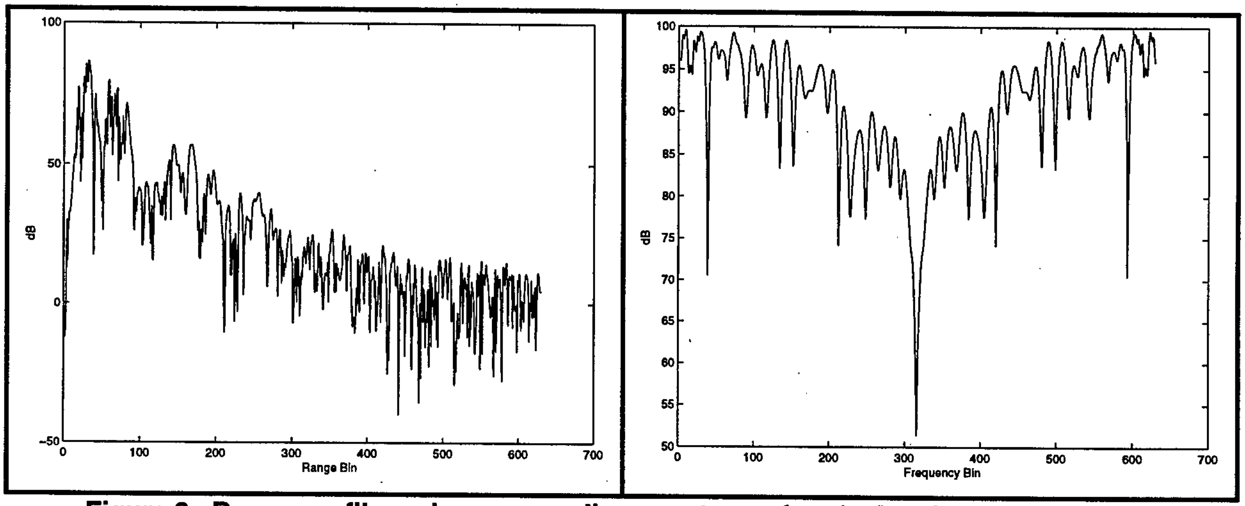

Figure 9. Range profile and corresponding spectrum after decimation and baseband conversion.

The next step in the process is pulse compression. A plot of the time response as well as the frequency domain response is shown in Fig. 10.

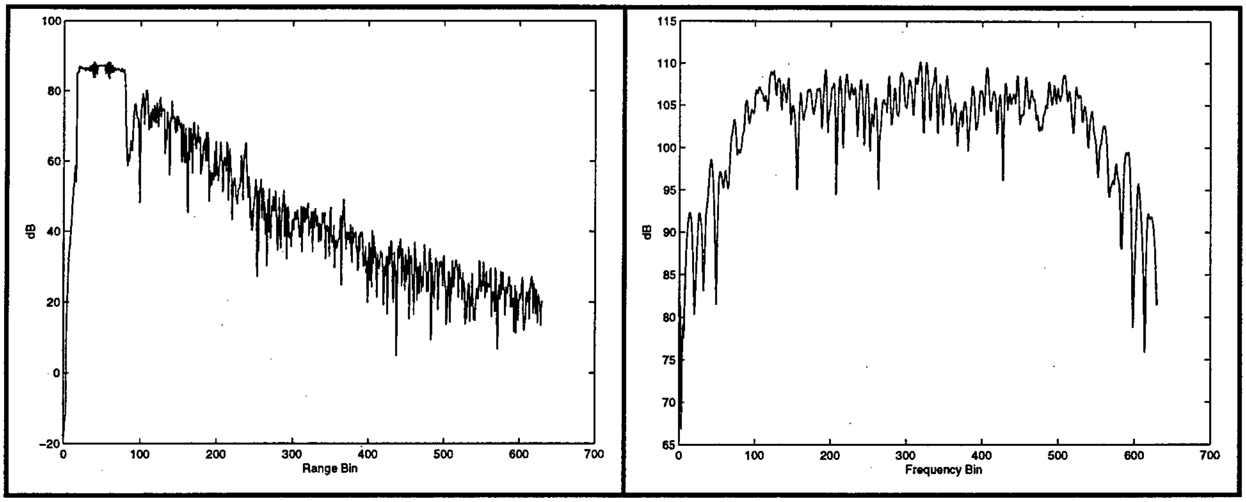

Figure 10. Range profile and corresponding spectrum after pulse compression.

Figures 8-10 show how the different pre-processing steps affect the radar signals. Figure 10 shows a typical radar signal after decimation, baseband conversion, and pulse compression. RADC performed all of the pre-processing steps.

In the following, we describe each step in more detail.

3.1 Digital IQ

Since the receiver bandwidth is 800 kHz centered at a 1.25 MHz intermediate frequency and the sampling rate is 5 MHz, the MCARM data is over-sampled. The sampled data frequency spectrum has bands with a bandwidth of 0.32π centered at frequencies $\pi/2+2\pi n$ and $-\pi/2+2\pi n$, where n is an integer. The data can be converted to complex samples, shifted to baseband and decimated 4 to 1 by the following procedure. (There could be some interest in decimating 2 to 1 and processing the data at a 2.5 MHz rate. If so, the digital I/Q filter should be matched to that bandwidth.). First the sampled data is mixed with (multiplied by) a complex signal with a frequency of one-fourth the sampling rate, i.e. $\pi/2$. This shifts the center frequency of the first positive frequency band to zero. Because this also shifts the band originally centered at $3\pi/2$ to π, it is necessary to low pass filter frequencies below $\pi/2$.

3.2 Pulse Compression

Pulse compression can be performed before or after digital IQ by applying the matched filter for a 1 MHz linear FM sweep of the transmit pulse. Let W be the pulse width and δ denote the digital decimation factor. Then, the filter length L should be

$$L = 4W/\delta.$$

This represents number of decimated samples in a transmit pulse.

3.3 Channel Normalization and Adaptive Beam Pointing

Because the channels could drift with time, the system has a built in test manifold. This test manifold was built as a corporate feed to simulate a signal injected at boresight. The channels are balanced with respect to a test signal injected into the test manifold. After the channels are balanced, STAP analysis can be performed on the data. A set of steering vectors is supplied to steer the adapted beam to any desired location. These steering vectors were collected during the range calibration and will compensate for any deviation in the test manifold from an ideal boresight signal.

3.3.1 Channel Normalization

The channel balancing test signals are injected throughout the flight at about 3-minute intervals so that even short-term drift in the channels can be eliminated. Preliminary data analysis indicates that the channels stay reasonably balanced over the entire flight. To get the maximum cancellation possible, one should pick a balancing signal as close as possible in time to the actual clutter. These channel balance signals are produced in mode "RxTest" or "RxTest50." The acquisitions that contain the channel balance signals can be found by searching the BSC log files for times when the receiver mode (Rcvrmode) is equal to 0 and the beam steering type (BeamSteeringType) is equal to 4. Alternately, one can identify the channel balance files on 8-mm tape by their small size.

The channel balance process can be a simple one point average balance of the receivers. This task would be accomplished by calculating the peak response of the "RxTest" data across channels after pulse compression. Since the pulse compression process sums up the energy across the entire pulse compression bandwidth, this can be seen as an average balance across the band. This simple one point balance should be good for a 40-dB channel to channel balance across the entire band. Given the fact that clutter is a narrow band phenomenon, this simple one point balance should be more than effective for clutter cancellation.

Alternately, the channel balance process can use a multi-point subbanding process whereby the entire bandwidth is broken up (filtered) into discrete subbands before pulse compression and then balanced at each subband interval. This process with 19 subband intervals should attain 55-dB channel to channel balance. This process would be necessary for canceling wide band jamming.

The one point channel balance can be described by the following equation:

$$CN = CM / NR$$

where

- CN is the normalized clutter data vector across the channels of interest for one range gate.
- CM is the unnormalized clutter data vector across the channels of interest for one range gate.

- NR is the normalization vector across the channels of interest for one (there are 4 injected signals to chose from) of the peak signal range gates in the "RxTest" data.

This division represents a point by point (channel by channel) division. This division must be repeated across all range cells of the clutter data using the same normalization vector NR.

The channels of interest for STAP analysis are channels 2 through 8 and 10 through 24. Recall that channel 1 is the analog sum channel and channel 9 is the analog delta channel.

The normalization vector, NR, should itself be normalized to either, one of its elements (channels), or to the noise power in the vector.

3.3.2 Channel Normalization for Flights 4 through 11

For flights 4 through 11, RxTest samples were taken at each receive beam position at which clutter was collected. It is important to choose a RxTest sample at the same receive beam position that the clutter was collected when balancing the channels. This is because the module's phase and amplitude varies with beam position and the injected signal goes through the modules. Given that the RxTest sample vector, NR, comes from the same beam position as the clutter, CM, the channel can be balanced using the same expression as before

$$CN = CM / NR.$$

If a RxTest sample is chosen from a different beam position than the clutter, the procedure in the following section must be followed.

3.3.3 Channel Normalization for Flight 1

For flights 1 through 3, the RxTest samples were not always taken at the same receive position as the clutter data. The RxTest samples for these flights were always sampled at beam position 64. A MATLAB mat file, f1240mod.mat, exists which lists the phase and amplitude settings as a function of beam position. The variable "mod" refers to the elevation type. It could be e-5, e0, or e +5. Using the data in this file, one can adjust the channel balance for the change in module amplitude and phase as follows:

$$CN = (CM / NR) * (MR / MC)$$

where

- MR is a vector of the module settings for the beam position when the RxTest mode was taken.
- MC is a vector of the module settings for the beam position when the clutter was collected.

Again, these divisions and multiplications represent a point by point (channel by channel) division. All multiplications or divisions of CM should be normalized to preserve the original signal level of the clutter. MR and MC can be found using the variable

"McWeightArray" in the mat file f1240mod.mat. This is a 129 row by 24 column matrix where each row corresponds to the module settings (MR or MC) for the 24 channels for a given beam position. Use channels (columns) 2 through 8 and 10 through 24 for STAP analysis.

The variable "RcvAzPointIndex" + 1 is used to index the row in the variable "McWeightArray." For example, the RxTest signal was taken at beam position index number 64. Therefore, row number 65 in "McWeightArray" is to be used.

3.3.4 Channel Normalization for Flights 2 and 3

For flights 2 and 3, in addition to having the RxTest samples at different module settings than the clutter data, the receiver attenuators were set differently. While one could use the header variable that describes the attenuation settings (AttenChannels) to correct the normalization process, this would not account for the slight change in phase for different attenuation settings. Therefore, a RxTest signal was sampled at both attenuation settings and the difference in phase and amplitude is contained in a mat file called "attc_rx.mat." Using the data in this file, one can adjust the channel balance for the change in receiver attenuation settings as follows:

$$CN = (CM / NR) * (MR / MC) / (attc\_rx)$$

where

- attc_rx is a 24 point vector of the difference in attenuation settings between clutter data collection and RxTest data collection and can be found in the mat file "attc_rx.mat."

Again, these divisions and multiplications represent a point by point (channel by channel) division. All multiplications or divisions of CM should be normalized to preserve the original signal level of the clutter. For STAP analysis, channels (columns) 2-8 and 10-24 are to be used.

3.4 Steering Vectors

As mentioned previously, because the test manifold is not exactly equivalent to a boresight injected test signal, the steering vectors should be normalized by this difference. The steering vector is not critical to the clutter cancellation process: the information in the clutter covariance matrix determines cancellation levels. The steering vector simply sets the direction of the collimated beam and as such could conceivably have fairly large errors before the peak gain suffers much. For example, if the ± 20° delta between the test manifold and a pure boresight signal were not compensated for, the loss in gain would only be 0.17 dB. Therefore, one could calculate steering vectors using the theoretical antenna parameters without significant loss.

For convenience, a set of nominal steering vectors are provided which contain the difference between the test manifold and a far field source for a number of beam directions. If anyone desires a different steering direction than those provided, they could use the difference between the test manifold and the boresight signal (file f1240e0.mat, index 65) as a normalization vector and then steer the array by calculating the steering vector given the array spacing and frequency.

3.4.1 Steering Vector Files

Three steering vector files in MATLAB format are listed in the table below

Table 1. List of steering vectors.

Filename	Elevation Angle
f1240e-5	-5.4°
f1240e0	0.0°
f1240e+5	5.4°

There is a set of 129 azimuth steering vectors for each of the 3 elevation angles listed. These are steering vectors in the sense that they are the amplitude and phase taper you would measure in the receivers if a target were received from the direction given. If these vectors are used as beam pointing weights, then the conjugate of these variables should be used. If they are used as weights, they should produce a $\sin(x)/x$ response; however, due to range effects, mutual coupling, and channel mismatch, the resultant pattern will have some odd sidelobes. Each steering vector array has 129 rows, which represent azimuth angle. These angles are evenly spaced in sin space from -1 to +1 (-90° to +90°).

3.4.2 Steering Vectors for Flights 1 through 4

There is a 24 point steering vector, "McSteerArray," which maps to the MCARM combined channels for flights 1 through 4. The values in columns 1 and 9, corresponding to the sum and difference channels, have no real meaning in a steering vector sense. To use these values for the MCARM channels, use columns 2 to 8 and 10 to 24, and then multiply by the corresponding columns of clutter data.

3.4.3 Steering Vectors for Flights 5 through 11

There is a 32 point steering vector, "SteerArray," for flights 5 through 11. Use columns 7 to 28 [7:28] of SteerArray for the 22 channels of clutter data in columns [2:8 10:24] of CPI1.

3.5 Processed Data Nomenclature

Processed data files are named according to the following form: rcxxyyyy.mat,

Where

- r stands for Rome research site

- c is the cycle number
- xx relates to the flight number

and

- yyyy relates to the acquisition number within a given flight.

As example the file

re050152.mat

contains data collected from flight **5**, cycle **e** and acquisition **152**.

To date, we have processed all data files made available to us and our database consists of 569 files. The data files can be grouped into the following categories:

Table 2. List of available MCARM data files.

Filename	Flight Number	Cycle	No. of files
rd05xxxx	5	d	222
re05xxxx	5	e	236
ri04xxxx	4	i	111
		Total	569

The database has a variety of files, consisting of narrow beams as well as blob beams, where the main beam was pointed at indices 55, 65 and 75, corresponding to –8.989°, 0° and 8.989°, respectively.

We have also collected and processed over 100 MTS data files and made them available through the Internet. This gives the user a good blend of data sets to choose from, covering different regions (urban, suburban, forest, water, etc.) with different parameters.

4.0 Data Reduction

This section describes the processes required to reduce the MCARM data to a form suitable for adaptive processing. Auxiliary data files and their use is also described along with some preliminary STAP analysis.

Each sample has 16 bits of information. The A/D offset baseband sampled data has been shifted to the upper 14 bits (the A/D was 14 bits, 2's complement). The lower 2 bits are used for a saturation and R0 indication. Bit '0' indicates an A/D saturation. Bit '1' indicates the start of each IPP, R0. This 16 bits of data can be used 'as is' since the lower two bits will not appreciably affect the results.

The offset baseband data can first be converted to digital I & Q and decimated if desired. The IF bandwidth is 0.8 MHz, so for most applications, the data can be decimated (4 to 1) down to a 1.25 MHz sample rate.

After the data is in complex form, it can then be normalized or channel balanced. Since the pulse contains 1 MHz of Linear Frequency Modulation (LFM), the data should also be pulse compressed. Pulse compression and channel balancing are order independent linear processes which are both applied in the time domain; and therefore, they could be combined in the same process. They will be described here as separate processes.

After digital I/Q, pulse compression and channel balance processing the data is ready to be tested in any STAP algorithm desired. Steering vectors are provided which will steer the adaptive beam in a desired direction.

The steering vectors come from the range calibration data (see section 2). All of the range calibration data was processed using the same steps, i.e., digital I/Q, decimation, and pulse compression. Somewhere in these steps, probably in the digital I/Q, a sign reversal occurs. Therefore, MACRM data shows negative Doppler for beams pointed towards the nose. This could be explained by assuming that the digital I/Q process switched the logical meaning of I & Q -- i.e. Q is I & I is Q. Since the range calibration data is used to steer the beam after normalizing channels, one may want to process the clutter data with the same I/Q flip for consistency of data normalization.

5.0 STAP Processing

5.1 Introduction

Space-Time Adaptive Processing (STAP) is a multi-dimensional adaptive filtering process. It has been shown to hold promise in improving airborne radar performance in severe clutter and jamming environments. STAP involves linearly combining signals from the elements of an array of sensors and multiple pulses of the radar waveform to suppress interference and improve target detection [2,3].

Numerous system design studies and computer simulations have demonstrated the potential benefits of STAP techniques. However, the absence of measured airborne radar data has limited progress towards fielding STAP-based systems. The MCARM program provides a database of measured data, essential for evaluation of STAP methods.

In this report, monostatic MCARM data is used to evaluate the performance of several STAP techniques. The data is taken from 3 different flight paths to show the effectiveness of the proposed techniques. STAP techniques used in the comparative analysis include the fully adaptive technique, reduced-dimension/rank techniques, and eigen-based and parametric techniques. Specifically, we consider the performance of the fully adaptive technique referred to as the Adaptive Matched Filter (CFAR-AMF) [3,4]. We also consider the reduced-dimension techniques such as the Factored Time-Space (FTS) [2], the Extended FTS (EFA) [5], the Adaptive Displaced Phase-Center Aperture (ADPCA) processing [6] and the Joint Domain Localized (JDL) processing [7]. In addition, we include the performance of eigen-based techniques [8-10] such as the principal component technique, also referred to as Eigencanceler (EC) [10] and the Cross-Spectral Metric [11] selection procedure. Finally, a newly proposed STAP technique referred to, as the Parametric Adaptive Matched Filter (PAMF) is also included [12]. A full description of the different techniques is given in section 5.

Computational complexity and the non-homogeneous nature of the airborne radar environment represent two reasons for developing reduced-dimension and reduced-rank techniques. These methods involve linear transformations, such as from the time-domain to the frequency-domain or from the space-time domain to an orthonormal eigen-based coordinate system, and then a selection procedure to reduce the adaptive processor's degrees of freedom. It has been shown in [2,5-12] that these alternative STAP methods often effectively cancel the interference of the transformed signal at significant computational savings. In addition, less stringent secondary data requirements make the reduced-dimension/reduced-rank techniques more pragmatic in commonly encountered operational scenarios.

Typically, one uses a maximum likelihood estimate (MLE) of the interference covariance matrix to implement the adaptive processor [13]. The MLE approach avoids convergence problems associated with recursive methods when a large eigenvalue spread characterizes the interference, as is the case in airborne radar applications. Denote the k-th realization of an M-dimensional, complex baseband signal vector as $\underline{X}_k$. An estimate of the interference covariance matrix of $\underline{X}_k$ is then given by

$$\hat{R}_k = \frac{1}{K} \sum_{i=k-K/2}^{k+K/2} \underline{X}_i \underline{X}_i^H \quad ; \quad i \neq k, \tag{1}$$

where the vectors $\underline{X}_i$ are assumed to be independent and identically distributed (iid) secondary data vectors.

In the adaptive radar problem, the k-th realization, $\underline{X}_k$, corresponds to a measurement at the k-th range cell. It often is referred to as the primary data. When the secondary data is Gaussian iid matched to the interference statistic of the primary data, selecting approximately 2M secondary data yields an acceptable covariance matrix estimate [8]. The output of the adaptive filter for the k-th range cell is then given by

$$y_k = \underline{w}_k^H \underline{X}_k, \tag{2}$$

where $\underline{w}_k$ in Equation (2) is the Mx1 adaptive weight vector, given by

$$\underline{w}_k = \hat{R}_k^{-1} \underline{v}, \tag{3}$$

and $\underline{v}$ is the target steering vector.

The space-time steering vector contains the linear phase progression defining a unity amplitude signal impinging on the array of sensors from a specific direction of arrival with a given Doppler frequency shift [3,4]. When the M by M covariance matrix is known, the weight vector maximizes the signal-to-interference plus noise ratio (SINR) for the chosen target steering vector.

5.2 Reduced-dimension/rank STAP

Consider a multi-channel airborne radar with J channels, collecting N pulses over L unambiguous range cells. We define the space-time signal vector of complex, baseband signals for the k-th range cell as,

$$\underline{X}_k = [x_k(1,1)\ldots x_k(1,J)x_k(2,1)\ldots x_k(2,J)\ldots x_k(N,1)\ldots x_k(N,J)]^T, \tag{4}$$

Using Equation (4) in Equations (1) and (2), we compute the fully adaptive filter output, where $\underline{v}$ is the space-time steering vector defined in [3]. Note that the covariance matrix has dimension JN by JN.

In contrast to the fully adaptive case, reduced-dimension and reduced-rank methods linearly transform Equation (4) and then select groupings of distinct signal subspaces over which to adapt. This has the advantage of lessening computational burden and reducing secondary data requirements for the MLE.

5.2.1 Factored Time-Space (FTS)

The factored time-space (FTS) method [2], also known as post Doppler adaptive beamforming, provides good performance when sufficient separation exists between the interference and desired signal subspaces.

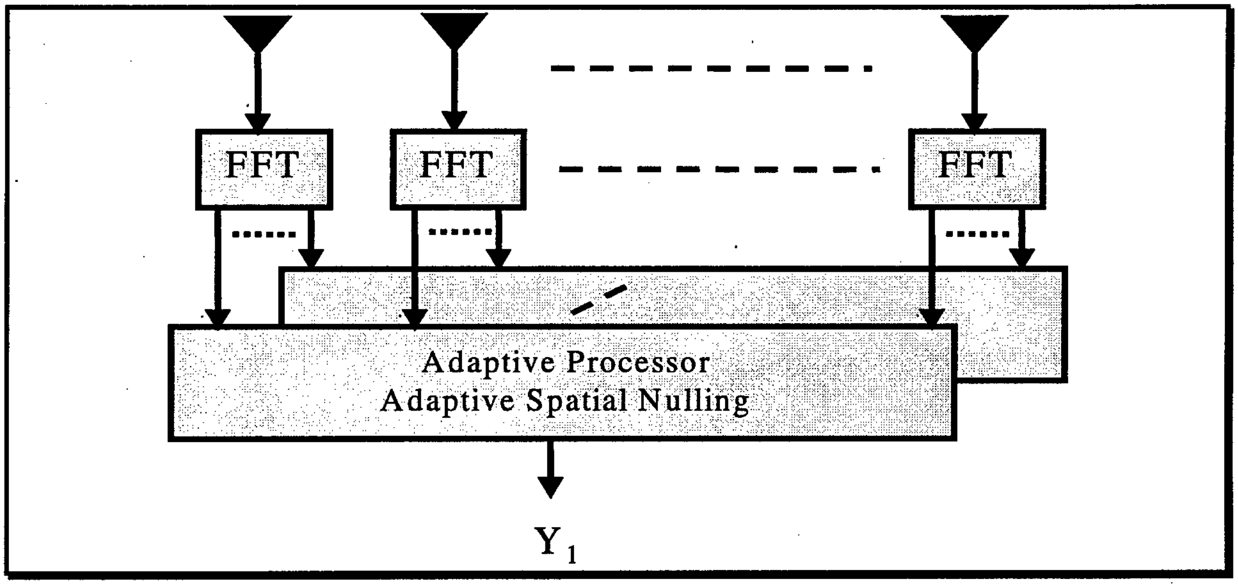

Figure 11. Factored Time-Space Approach (FTS).

To best describe this approach, we first reconfigure the data vector of Equation (4) as the (NxJ) “snapshot” signal matrix,

$$X_k = \begin{bmatrix} x_k(1,1) & x_k(1,2) & \cdots & x_k(1,J) \\ x_k(2,1) & x_k(2,2) & \cdots & x_k(2,J) \\ \vdots & \vdots & \vdots & \vdots \\ x_k(N,1) & x_k(N,2) & \cdots & x_k(N,J) \end{bmatrix} = [\underline{X}_1^{[k]} \quad \underline{X}_2^{[k]} \quad \cdots \quad \underline{X}_J^{[k]}] \tag{5}$$

where $\underline{X}_m^{[k]} = [x_k(1,m),\ x_k(2,m),\ \ldots, x_k(N,m)]^T$ is the (Nx1) vector containing all pulses for channel m. Doppler processing is then performed on each column of Equation (5), yielding the transformed signal matrix given by,

$$\tilde{X}_k = \begin{bmatrix} \tilde{x}_k(1,1) & \tilde{x}_k(1,2) & \cdots & \tilde{x}_k(1,J) \\ \tilde{x}_k(2,1) & \tilde{x}_k(2,2) & \cdots & \tilde{x}_k(2,J) \\ \vdots & \vdots & \vdots & \vdots \\ \tilde{x}_k(N,1) & \tilde{x}_k(N,2) & \cdots & \tilde{x}_k(N,J) \end{bmatrix} = [\tilde{\underline{X}}_1^{[k]} \quad \tilde{\underline{X}}_2^{[k]} \quad \cdots \quad \tilde{\underline{X}}_J^{[k]}] \tag{6}$$

where the vector $\tilde{\underline{X}}_m^{[k]}$ is the Discrete Fourier Transform (DFT) of the vector $\underline{X}_m^{[k]}$. We then select a desired Doppler filter, corresponding to a given row of Equation (6), and adaptively combine the J spatial samples. Hence, the output of the adaptively filtered n-th Doppler filter for the k-th range cell is given by

$$\tilde{y}_k(n) = \underline{v}_s^H [R_k(n)]^{-1} \tilde{\underline{\underline{X}}}_k(n), \tag{7}$$

where $\underline{v}_s$ is the (Jx1) spatial steering vector,

$\tilde{R}_k(n) = \frac{1}{K} \sum_{i=k-K/2}^{k+K/2} \tilde{\underline{\underline{X}}}_i(n) \tilde{\underline{\underline{X}}}_i^H(n)$; $i \neq k$, is the JxJ spatial covariance matrix,

and

$\tilde{\underline{\underline{X}}}_k(n)$ is the transpose of the n-th row of $\tilde{X}_k$.

Note that in estimating $\tilde{R}_k(n)$, 2J iid samples are generally required.

5.2.2 Extended Factored Time-Space (EFA)

The extended factored time-space approach (EFA) [5] is, as the name implies, an extension of FTS. In this case, however, we apply adaptive processing to a fixed number of adjacent Doppler bins in an attempt at potentially enhancing interference rejection in those severe instances requiring more degrees of freedom than FTS. As an example, the order 2 method adapts over three adjacent Doppler bins.

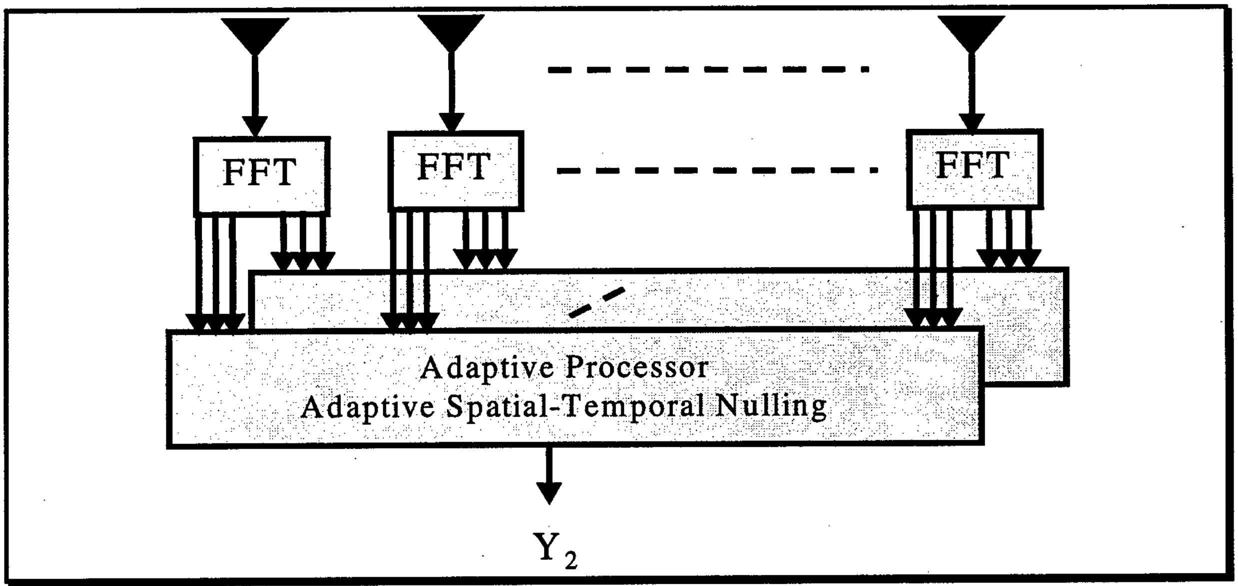

Figure 12. Extended Factored Approach (EFA).

The filtered output of the n-th Doppler filter is given by

$$\breve{y}_k(n) = \underline{v}_F^H \left[\breve{R}_k(n)\right]^{-1} \underline{\underline{\breve{X}}}_k(n), \tag{8}$$

where

$$\underline{v}_F = [0\ 1\ 0]^T \otimes \underline{v}_s, \tag{9}$$

with ⊗ being the kronecker tensor product,

$\underline{v}_s$ is the (Jx1) spatial steering vector,

$\underline{\breve{X}}_k(n) = [\underline{\underline{\tilde{X}}}_k^T(n-1)\ \underline{\underline{\tilde{X}}}_k^T(n)\ \underline{\underline{\tilde{X}}}_k^T(n+1)]^T$,

and

$\breve{R}_k(n) = \frac{1}{K} \sum_{i=k-K/2}^{k+K/2} \underline{\breve{X}}_i(n) \underline{\breve{X}}_i^H(n)$; $i \neq k$, is the (3Jx3J) spatial covariance matrix requiring approximately 6J secondary data samples.

5.2.3 Joint-Domain Localized (JDL) Processing

Joint-Domain Localized (JDL) processing [6] involves adapting over a local processing region (LPR) consisting of adjacent angle and Doppler bins. Reference [6] recommends a 3x3 LPR to cancel ground clutter returns predominantly confined to a diagonal ridge in the angle-Doppler space.

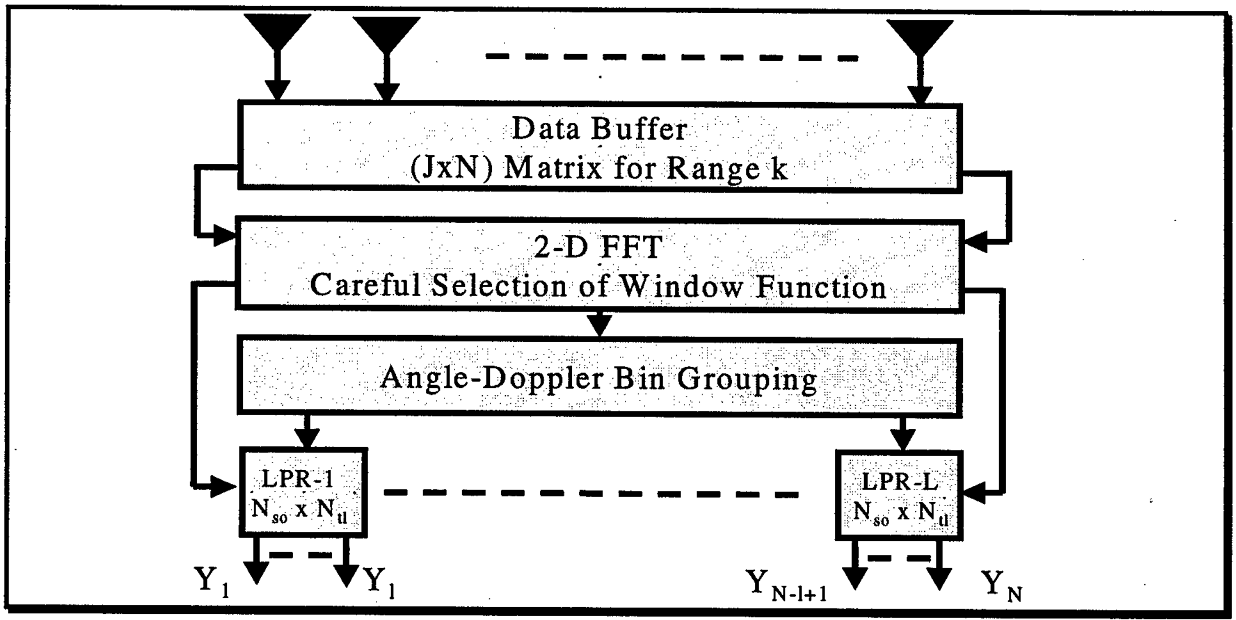

Figure 13. Joint Domain Localized (JDL).

Let the two-dimensional DFT of Equation (5) be given by $\hat{X}_k = F\{X_k\}$, where F{.} is the 2-D Fourier transform operator. The mn-th element of $\hat{X}_k$, expressed as $\hat{x}_k(m,n)$, is referred to as the mn-th angle-Doppler bin. Then, the output of the p-th (3x3) LPR and range k is given by

$$\hat{y}_k(p)=\underline{\tilde{v}}^H\left[\hat{R}_k(p)\right]^{-1}\underline{\hat{X}}_k(p), \tag{10}$$

where

$\underline{\tilde{v}}=[000010000]^T$ is the frequency domain steering vector pointing to the second angle and Doppler bin,

$\hat{R}_k(p)=\frac{1}{K}\sum_{i=k-K/2}^{k+K/2}\underline{\hat{X}}_i(p)\left[\underline{\hat{X}}_i(p)\right]^H$; $i \neq k$ is the 9x9 covariance matrix,

and

$$\begin{aligned}\underline{\hat{X}}_k(p)=[&\hat{x}_k(m-1,p-1)\hat{x}_k(m-1,p)\hat{x}_k(m-1,p+1)\hat{x}_k(m,p-1)\hat{x}_k(m,p)\\&\hat{x}_k(m,p+1)\hat{x}_k(m+1,p-1)\hat{x}_k(m+1,p)\hat{x}_k(m+1,p+1)]^T\end{aligned}$$

The 3x3 JDL processor requires roughly eighteen (18) iid secondary cells for accurate covariance matrix estimation.

5.2.4 Adaptive Displaced Phase-Center Aperture (ADPCA) Processing

Adaptive Displaced Phase-Center Aperture (ADPCA) processing is a special case of the fully adaptive technique when N = 2. We consider N=3, whereby a sub-optimal steering vector $\underline{s}$ replaces the target steering vector $\underline{v}$ [7].

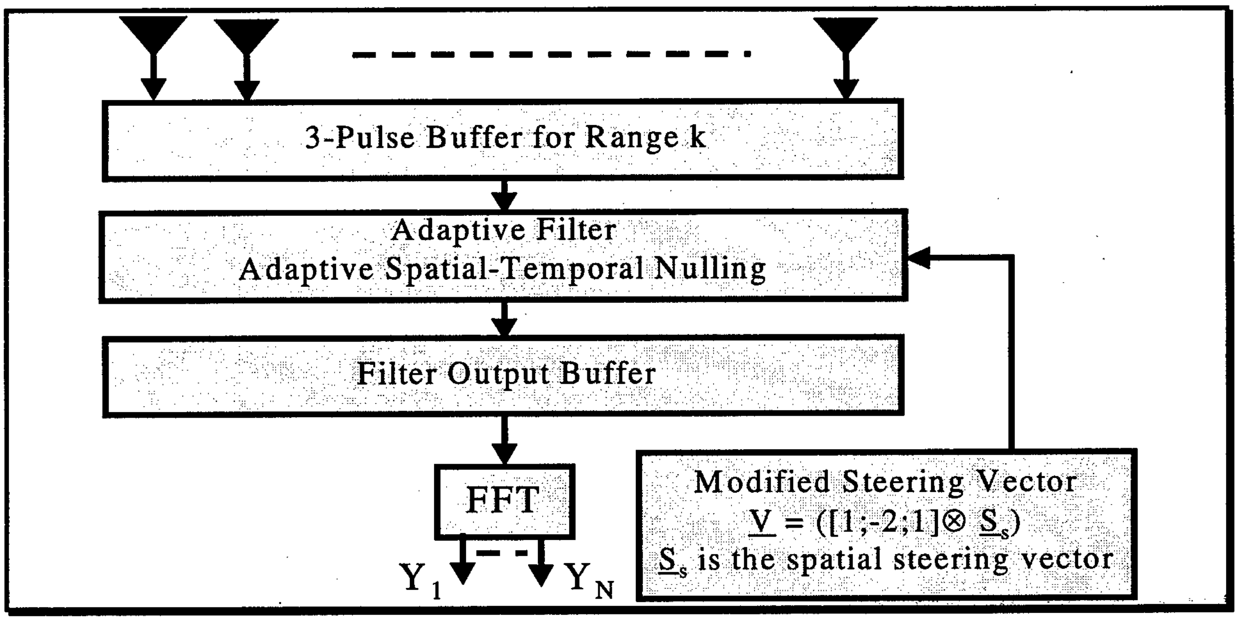

Figure 14. Adaptive Displaced Phase Center Aperture (ADPCA).

The output of the 3-pulse ADPCA processor is given by

$$z_k = \underline{s}^H \hat{R}_k^{-1} \underline{X}_k . \tag{11}$$

In Equation (11), $\underline{s}$ is the target steering vector defined as

$$\underline{s} = [1 \; -2 \; 1] \otimes \underline{v}_s .$$

$\underline{v}_s$ is the (Jx1) spatial steering vector,

where $\underline{X}_k$ and $\hat{R}_k$ are defined by Equations (4) and (1), respectively. The number of secondary data needed to estimate the covariance matrix is the same as that of the EFA technique (6J). In practice, Doppler processing follows the output of ADPCA for integration gain.

5.3 Eigen-Based Techniques

5.3.1 Principal Components (PC) Technique

The minimum norm eigencanceler derived in [10] is a principal components (PC) technique. This method relies solely on the interference information characterized by the largest eigenvalues, or principal components, of the covariance matrix. Specifically, the eigencanceler is a linearly constrained beamformer with a minimum norm weight vector appearing orthogonal to the interference subspace.

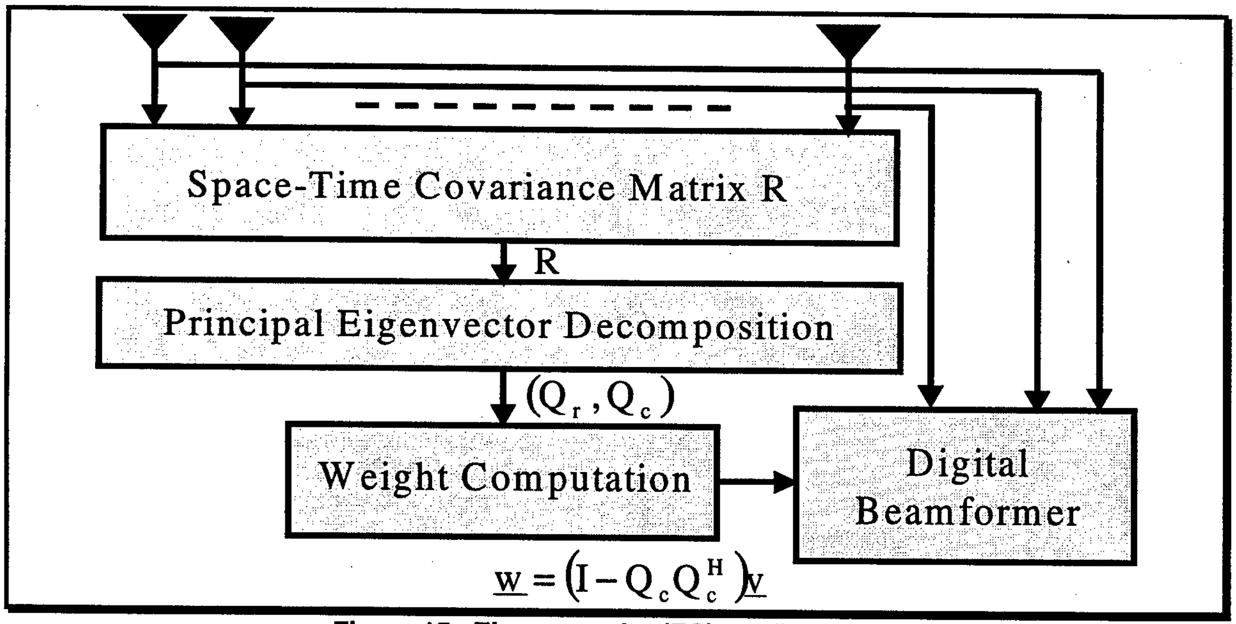

Figure 15. Eigencanceler (EC) configuration.

The eigencanceler weight vector is

$$\underline{w}=(I-Q_cQ_c^H)V[V^H(I-Q_cQ_c^H)V]^{-1}\underline{f}, \tag{12}$$

where

- $V^H\underline{w}=\underline{f}$ is a set of linear constraints,
- V is a constraint matrix,
- $\underline{f}$ is the desired response vector

and

- Q_c is the matrix whose columns are the eigenvectors corresponding to the strongest interference sources.

It is shown in [10] that for an idealized space-time system consisting of J elements and N pulses, the rank (r) of the clutter covariance matrix is bounded by (J+N-1), i.e.

$$r \leq J+\beta(N-1), \tag{13}$$

where β is the clutter ridge slope in the angle-Doppler domains.

This result appears to be much smaller than the dimension of the fully adaptive array (JN). For the single linear constraint, $f=\underline{v}^H\underline{w}$, where f is a scalar and $\underline{v}$ is the steering vector, the weight vector becomes

$$\underline{w}=(I-Q_cQ_c^H)\underline{v}[\underline{v}^H(I-Q_cQ_c^H)\underline{v}]^{-1}f=\alpha(I-Q_cQ_c^H)\underline{v} \tag{14}$$

where α is a constant and is given by

$$\alpha = \frac{f}{\underline{v}^H \left(I - Q_c Q_c^H\right) \underline{v}} \quad . \qquad (15)$$

The output of the eigencanceler for the k-th range cell is

$$y_k = \underline{w}_k^H \underline{X}_k = \underline{v}^H \left[I - Q_c^{[k]} \left(Q_c^{[k]}\right)^H \right]^{-1} \underline{X}_k , \qquad (16)$$

where the constant α has purposely been omitted. Note that the weight vector obtained with this technique is still of dimension (JN x 1).

5.3.2 Cross-Spectral Metric (CSM) Approach

More recently, Goldstein et al. introduced the Cross-Spectral Metric (CSM) technique in [11]. This technique reduces the dimension of the weight vector through an optimal selection procedure shown to maximize the output SINR in the known covariance matrix case. As in [11], consider the adaptive processor in the form of the Generalized Sidelobe Canceler (GSC).

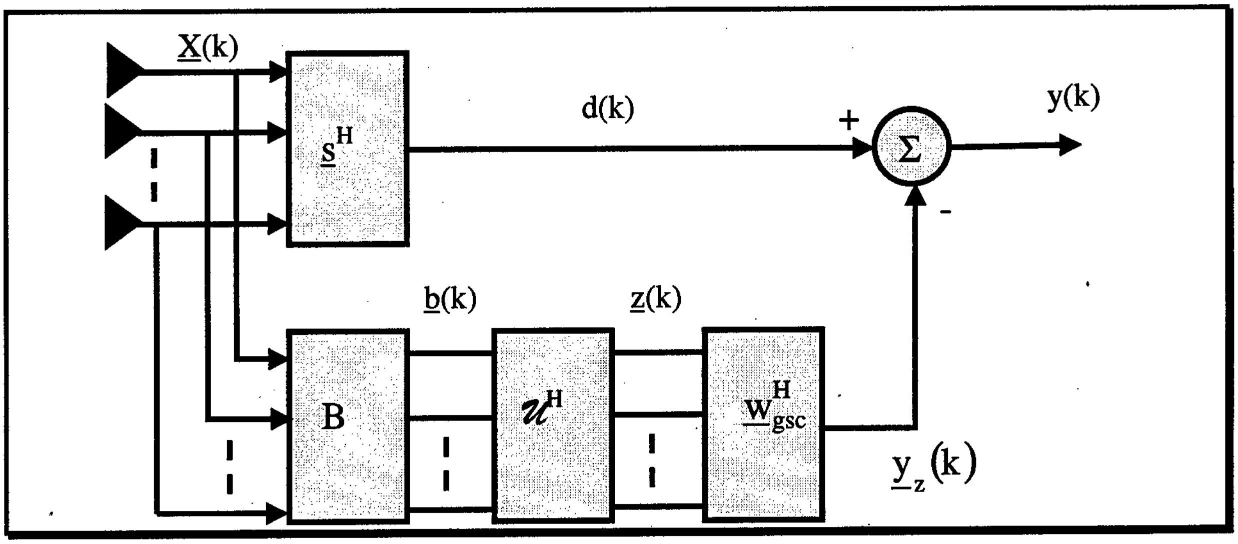

Figure 16. Cross-Spectral Metric-GSC structure.

In the above figure, the vector $\underline{s}$ is the normalized conventional space-time steering vector and the matrix B is referred to as the blocking matrix. The blocking matrix maps the test cell $\underline{X}(k)$ into the null space of $\underline{s}$, i.e.

$$B\underline{s} = \underline{0} . \qquad (17)$$

The matrix $\mathcal{U}$ reduces the dimension of the algorithm. It has dimension (JN-1) x r and its are composed of the r eigenvectors chosen from the eigen-decomposition of the noise subspace covariance matrix R_b given by $R_b = BRB^H$.

These eigenvectors are chosen according to the Cross-Spectral Metric (CSM) given by

$$\left|\frac{\underline{\mathcal{U}}_i \underline{r}_{bd}}{\sqrt{\lambda_i}}\right|^2,$$

where λ_i and $\underline{\mathcal{U}}_i$ are the eigenvalues and the corresponding eigenvectors of the matrix R_b, respectively, and $\underline{r}_{bd}$ is the cross-correlation vector between the noise subspace data vector and the steering vector and is given by

$$\underline{r}_{bd} = BR\underline{s}. \tag{18}$$

where R is the full (JN x JN) space-time covariance matrix of the data vector $\underline{X}(k)$. The r eigenvectors with the largest CSM are included in $\mathcal{U}$, resulting in the optimum SINR. The output of the CSM processor is given by

$$y(k)=\underline{w}_k^H \underline{X}(k). \tag{19}$$

The weight vector in (19) is expressed as

$$\underline{w}_k = \underline{s} - B^H \mathcal{U} \underline{w}_{gsc}, \tag{20}$$

where

$$\underline{w}_{gsc} = R_z^{-1} \underline{r}_{zd}. \tag{21}$$

The matrix R_z is expressed as

$$R_z = \mathcal{U}^H R_b \mathcal{U} = \Lambda_r, \tag{22}$$

where the matrix R_b is given by $R_b = BRB^H$ and the matrix Λ_r is the diagonal matrix composed of r eigenvalues corresponding to the eigenvectors which form the matrix $\mathcal{U}$.

The cross-correlation vector $\underline{r}_{zd}$ is given by

$$\underline{r}_{zd} = E\{\underline{z}d^*\} = \mathcal{U}^H \underline{r}_{bd}, \tag{23}$$

where $\underline{r}_{bd}$ is given by Equation (18).

5.4 Parametric Adaptive Matched Filter (PAMF)

The architecture for the parametric adaptive matched filter (PAMF) [12] is shown in Fig. 17.

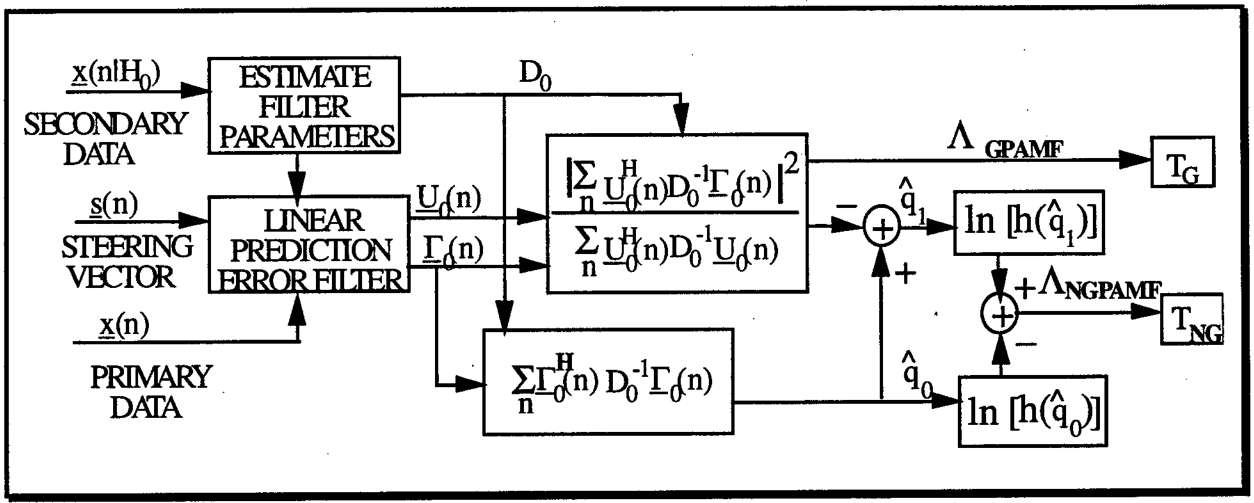

Figure 17. Parametric adaptive matched filter (PAMF).

In this method, multichannel coefficients of linear prediction are estimated directly from the secondary data using parameter estimation methods. These are then used to set the JxJ matrix weights in a multichannel prediction error filter (PEF) with a pre-specified order P. A typical delay line based PEF filter is shown in Fig. 18. However, lattice or state-space filters could be used.

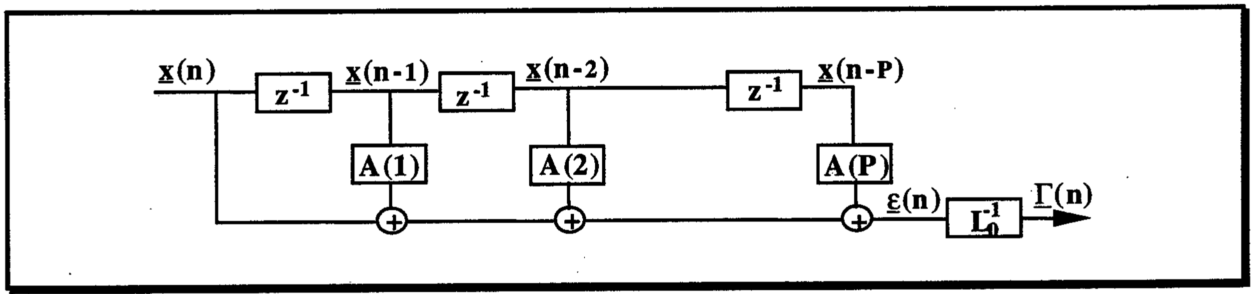

Figure 18. PEF with a tapped delay line structure.

Thus, the parameter estimation methods are used in lieu of the sample matrix estimator. When the error output of such a filter is minimized in a minimum mean-squared error (MMSE) sense, the resulting output is an innovation process [14]. In this report, the Nuttall-Strand algorithm [15,16] is used in a time series approach. It provides an estimate of the JxJ matrix autoregressive (AR) coefficients and white noise driving covariance matrix S.

For Gaussian processes, the PAMF test statistic is shown to be [12]

$$\Lambda_{\mathrm{GPAMF}} = \frac{\left|\sum_{n=1}^{N} \underline{U}_0^H(n) D_0^{-1} \underline{\Gamma}_0(n)\right|^2}{\sum_{n=1}^{N} \underline{U}_0^H(n) D_0^{-1} \underline{U}_0(n)} \tag{24}$$

where $\underline{\Gamma}_0(n)$ is the whitened observation data Jx1 vector output from the PEF, $\underline{U}_0(n)$ is the rotated steering vector obtained by passing the time sequence form of the steering vector, $\underline{s}(n)$, through the filter. The JxJ matrix D_0 is obtained from an LDL decomposition of the estimated S and is a measure of the variance of $\underline{\Gamma}_0(n)$. Even though it has not been considered in this report, a test statistic was derived for the non-Gaussian clutter case. The processes are modeled as a spherically invariant random process (SIRP). The test statistic is given by [12]

$$\Lambda_{NGPAMF} = \ln\left[h_{2JN}(\hat{q}_1)\right] - \ln\left[h_{2JN}(\hat{q}_0)\right] \tag{25}$$

where

$$\hat{q}_0 = \sum_{n=1}^{N} \underline{\Gamma}_0^H(n) D_0^{-1} \underline{\Gamma}_0(n) \tag{26}$$

and

$$\hat{q}_1 = \hat{q}_0 - \frac{\left|\sum_{n=1}^{N} \underline{U}_0^H(n) D_0^{-1} \underline{\Gamma}_0(n)\right|^2}{\sum_{n=1}^{N} \underline{U}_0^H(n) D_0^{-1} \underline{U}_0(n)} = \hat{q}_0 - |\hat{a}|^2 \sum_{n=1}^{N} \underline{U}_0^H(n) D_0^{-1} \underline{U}_0(n). \tag{27}$$

It can be shown that [12]

$$\hat{a} = \left(\sum_{n=1}^{N} \underline{U}_0^H(n) D_0^{-1} \underline{\Gamma}_0(n) \Big/ \sum_{n=1}^{N} \underline{U}_0^H(n) D_0^{-1} \underline{U}_0(n) \right). \tag{28}$$

5.5 Some Practical Issues

Performance metrics play an important role in comparing algorithm performance. In this report, we consider three metrics in our analysis. First, we show the variations of the modified sample matrix inversion (MSMI) test statistic over range. This allows us to assess the detection performance potential of the various methods when a known target is included in the data at a specific point in range. Under certain constraints, the MSMI test statistic demonstrates an embedded constant false alarm rate (CFAR) characteristic, appropriately scaling the data over range and enabling the selection of a fixed threshold. It is generically defined as

$$\text{Nmsmi} = \left|\underline{w}^H \underline{x}\right|^2 \Big/ \underline{w}^H \hat{R} \underline{w}. \tag{29}$$

A similar measure has been developed in [17]. It is a modified version of the Nmsmi and is referred to as the "normalized likelihood test." It is given by

$$\text{Nnlt} = \frac{\left| \underline{w}^H \overline{\underline{x}} \right|^2}{\left(\underline{w}^H \hat{R} \underline{w} \right) \left(\overline{\underline{x}}^H \hat{R}^{-1} \overline{\underline{x}} \right)}, \tag{30}$$

where $\overline{\underline{x}}$ is defined as the normalized data vector and is given by

$$\overline{\underline{x}} = \frac{\underline{x}}{\left\| \underline{x} \right\|}. \tag{31}$$

It is claimed [17] that this metric would perform extremely well with measured data. This, however, was not the case with MCARM data. For these reasons, this metric is not discussed any further in this report.

Finally, we consider the SINR loss, defined as

$$L_s = \left(\underline{s}^H \hat{R}^{-1} \underline{s} \right) / \left(\underline{s}^H \hat{R}_{\text{noise}}^{-1} \underline{s} \right). \tag{32}$$

This latter metric compares the adaptive filter's loss in SINR over the noise-limited condition. It aids in determining target visibility over the whole space occupied by interference and noise. However, estimating the noise covariance matrix is a very difficult task. For these reasons, we plotted, where needed, only the numerator of Equation (32). In this report, we only use the metric given by Equation (29).

Estimating the interference covariance matrix represents another practical concern when processing measured data. Unlike the case of simulating data with a known covariance matrix, each range cell of measured data represents a random process with unknown statistics. In addition, the data among the L unambiguous range cells can statistically vary within a coherent processing interval, resulting in an environment termed non-homogeneous. Such non-homogeneous signal environments arise for a variety of reasons, including spatially varying clutter, discretes, moving scatters and coherent repeater forms of electronic countermeasures, to name a few. Non-homogeneities exacerbate the adaptive filter implementation by violating the iid assumption intrinsic to the application of the MLE for covariance matrix formulation.

It is sometimes useful to screen the secondary data based on covariance structure and excise outliers in an approach analogous to a trimmed-mean CFAR [18,19]. For example, consider the Generalized Inner Product (GIP) scalar test statistic, given by

$$z_i = \underline{X}_i^H R_T^{-1} \underline{X}_i, \tag{33}$$

where $\underline{X}_i$ is an M dimensional signal vector and R_T is a test covariance matrix whose eigenvectors span the interference scenarios of interest. Equation (33) represents the sum of the squares of the output of a whitening filter defined by $R_T^{-1/2}$. We may use Equation (33) to assess the relative covariance structure of the available secondary data vectors and select those vectors appearing most homogeneous. We then form a MLE using these homogeneous data vectors to devise a covariance matrix estimate best suited to the majority of the range cells. As indicated earlier, one goal of this analysis is to briefly demonstrate the impact of non-homogeneous environments on adaptive filter performance through specific examples using measured MCARM data.

5.6 Data Analysis

In this report, data from several acquisitions have been used to demonstrate the effectiveness of the proposed techniques. Our analysis considers data from MCARM flight 5. More specifically, files rd050575, re050050, re050152 and re050280 have been used. Note that except for file re050050 which has a low PRF (PRF = 500 Hz), files rd050575, re050152 and re050280 have a medium PRF (PRF = 1984 Hz). Recall that in flight 5, the MCARM sensor is an L-band phased array antenna, configured as an 11 over 11 array.

Several papers [20-26] have used the MCARM data in their analyses. It is shown that the MCARM data provides much useful information, not usually available with simulated data. These include but are not limited to crab angle, internal clutter motion, mutual coupling, phase errors, etc.

To test detection performance potential, a target-like signal with known signal-to-noise ratio is injected into the data, except for file re050152 which already contains 5 tones generated by a moving target simulator (MTS) device. We will give the results of each analysis separately, and concluding remarks will be made at the end of this report.

5.6.1 Data file: rd050575

In this particular case, only J = 14 channels, N = 16 pulses and all 630 unambiguous range cells have been used. This allowed us to use all the previously described algorithms, since the sample support size for the covariance matrix estimation was met. A target was artificially injected at Doppler bin 3, range bin 290 and angle bin 65, corresponding to boresight. A plot of the angle-Doppler profile for this particular range is given in Fig. 19.

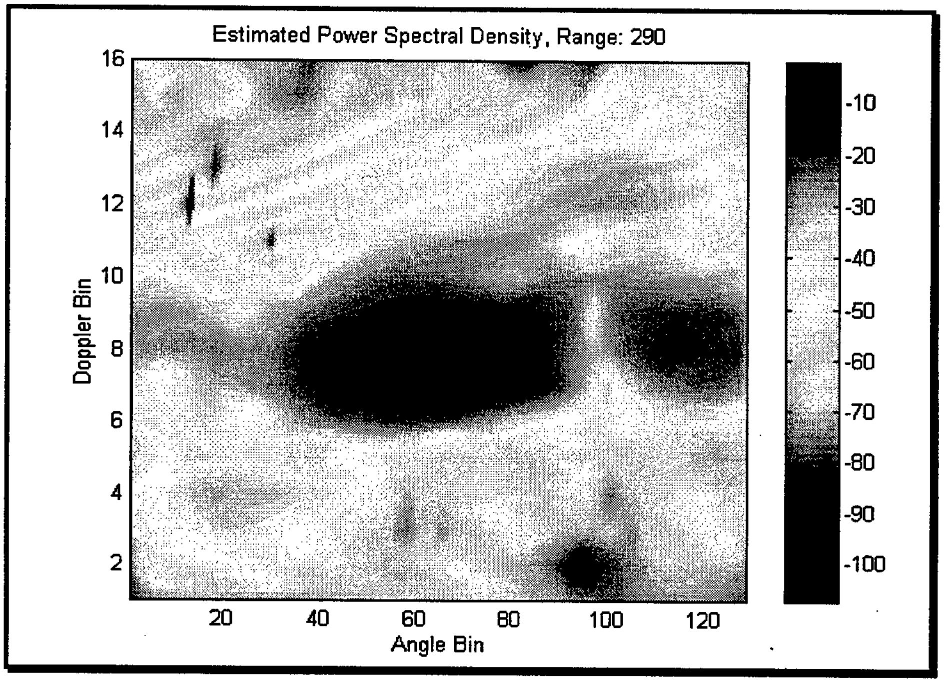

Figure 19. Estimated PSD, Range Cell 290.

Figure 20 shows the output of a digital beamformer weighted with a Hanning window.

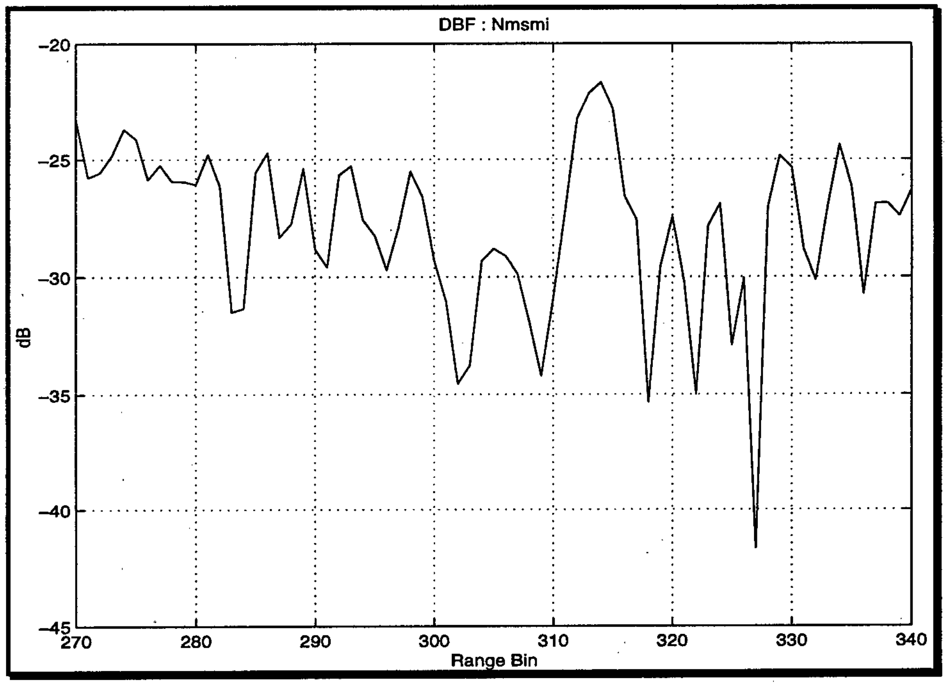

Figure 20. Output of digital beamformer, Doppler bin 3.

Note from Fig. 20 that the digital beamformer was unable to detect the target at range bin 290. It is very difficult for conventional techniques, such as digital beamforming, to detect the target, since it is competing with high sidelobe clutter. Note also that there is a strong false alarm at range bin 315.

In the remainder of this report, we define the target gain as the ratio between the peak at the target's location (in this case range bin 290) and the next highest peak.

Figure 21 shows the MSMI test statistic using a fully adaptive processor, or at times referred to as the CFAR-AMF.

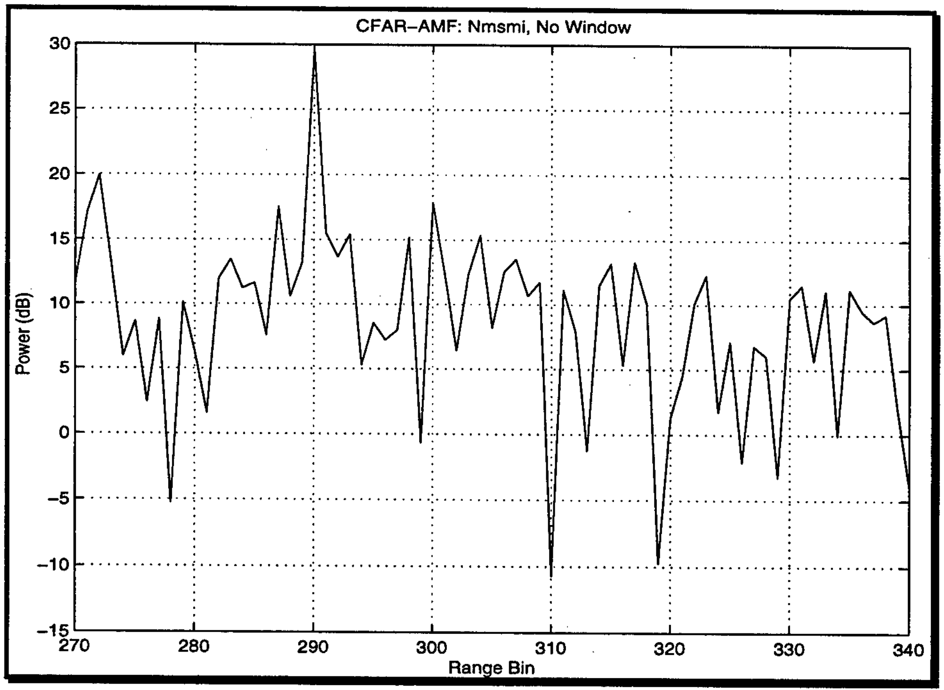

Figure 21. CFAR-AMF output.

In this case 2K samples have been used as secondary data for covariance matrix estimation, where K = 224. Note from Fig. 21 that the target is well recovered at its exact location (range bin 290) with a target gain of 9.69 dB.

Figure 22 shows the eigenspectrum of the interference covariance matrix. This matrix was computed using 448 samples (secondary data) taken symmetrically around range bin 290.

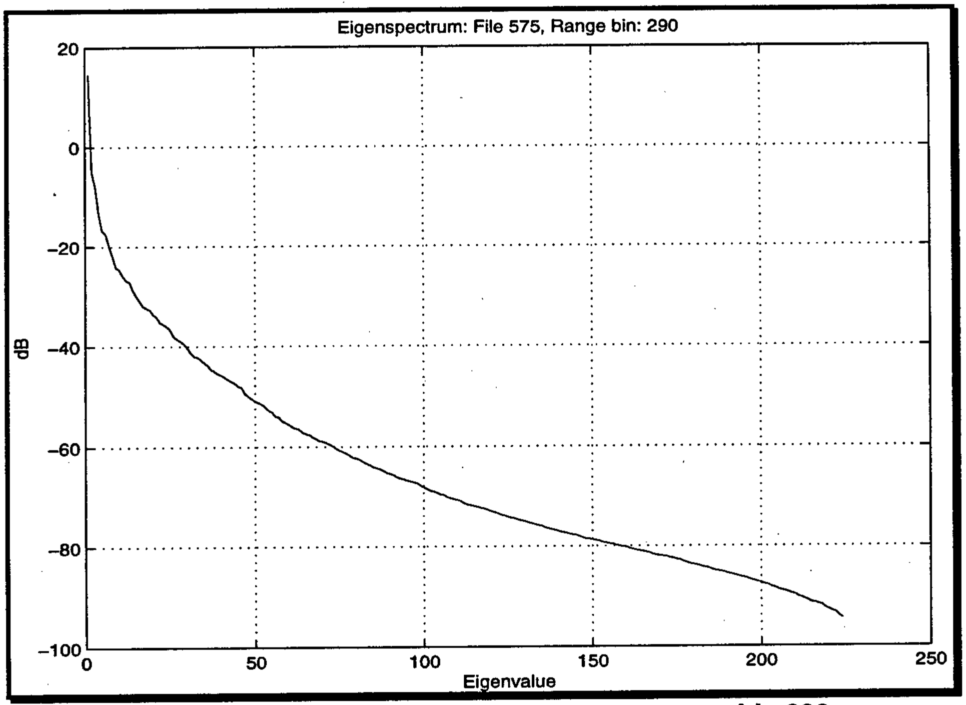

Figure 22. Interference eigenspectrum at range bin 290.

Note from Fig. 22 that the interference spectrum does not show a sharp cutoff at the interference rank value. There is rather a gradual decrease that will cause major problems in the rank selection. This is best illustrated using the eigencanceler and the cross-spectral metric approaches.

Next the principal components (PC) technique, also known as the eigencanceler, was tested against this set of data. In theory, the number of principal components to be used is bounded by the following quantity [10]:

$$r < (J + N - 1),$$

where J and N are the number of channels and the number of pulses, respectively. In our scenario, r should be less then 29. Figure 23 shows the output of the eigencanceler with $r = 29$.

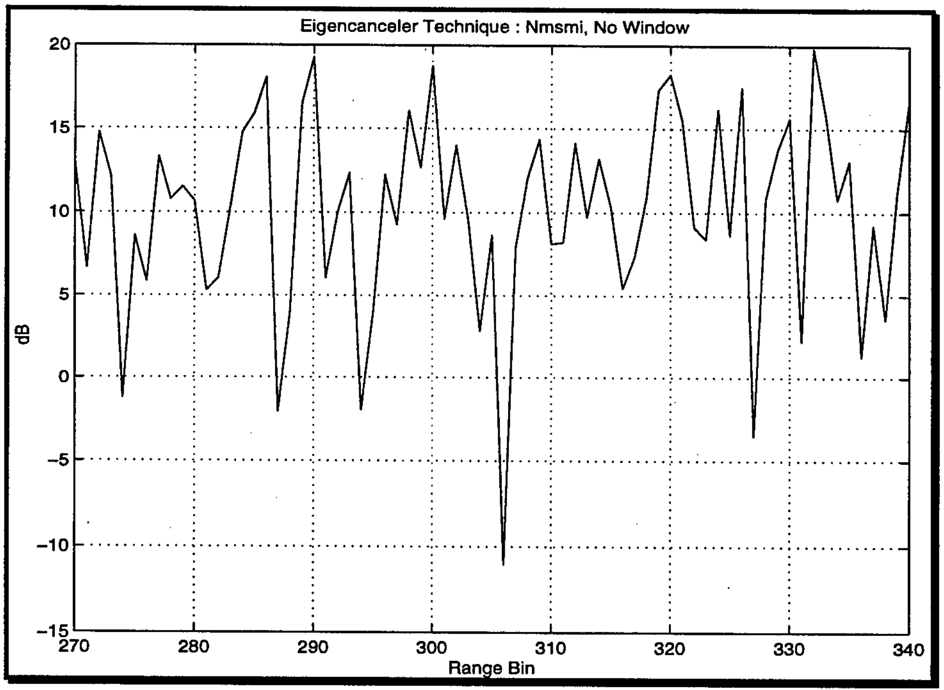

Figure 23. Output of eigencanceler, r = 29.

Note from Fig. 23 that the eigencanceler was unable to detect the target. After several trials, we realized that r should be much higher than 29 for the eigencanceler to perform well.

Figure 24 shows the output of the eigencanceler when r = 120.

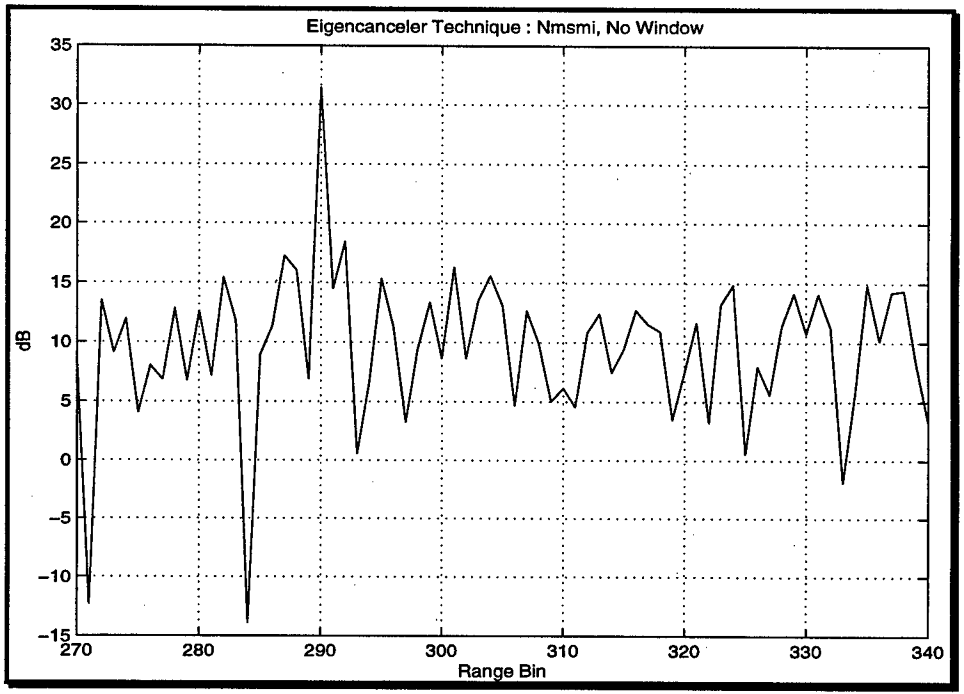

Figure 24. Output of eigencanceler, r = 120.

Note from this figure that the target gain is 13.05 dB. This represents an improvement of 3.36 dB when compared to the CFAR-AMF, which is particularly significant when dealing with low flying / low small cross-section targets. In the above case, the covariance matrix was estimated using only 2r (r =120) samples. Later in this section, we explain the reasons behind these results.

To have a fair comparative analysis with the cross-spectral metric, we show below the performance of the eigencanceler when r = 160. Note again that 2r (r = 160) samples have been used in the covariance matrix computation.

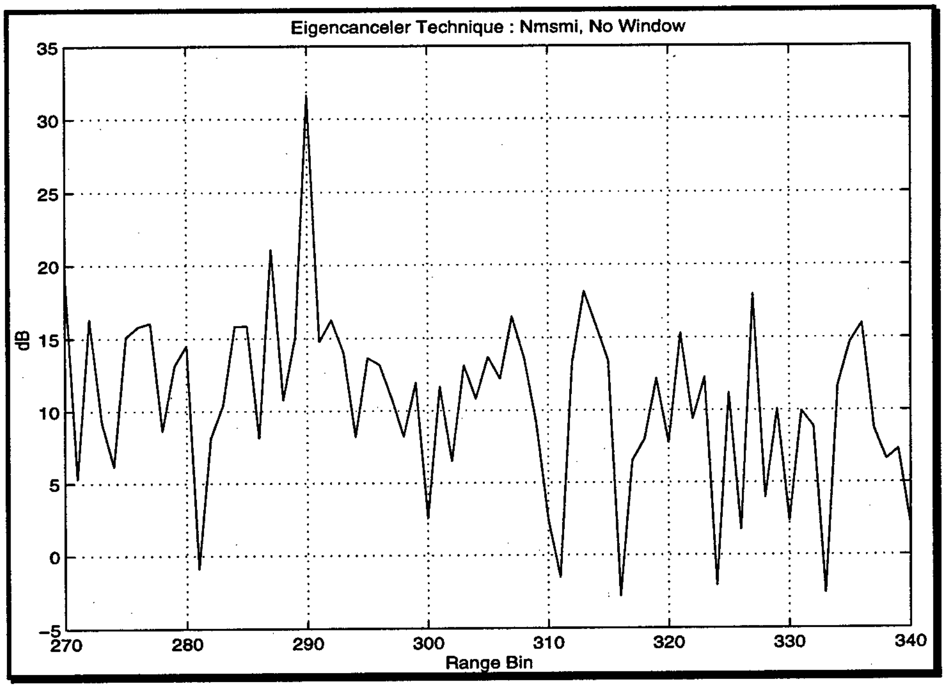

Figure 25. Output of eigencanceler, r = 160.

Note from Fig. 25 that the target gain is now 10.54 dB. This is about 0.85 dB (less than 1 dB) better than the CFAR-AMF. Actually, the best response was obtained when r = 120 (80 samples fewer to use for covariance estimation). Acceptable performance can be achieved with much lower values of r (actually, we obtained good performance with r = 60).

It is shown in [11] that when using the cross-spectral metric (CSM) approach, the number of principal components could be much less than that used for the eigencanceler. However, using the MCARM data, this proved to be untrue. The degradation of the CSM approach with a value of r = 29 is shown in Fig. 26.

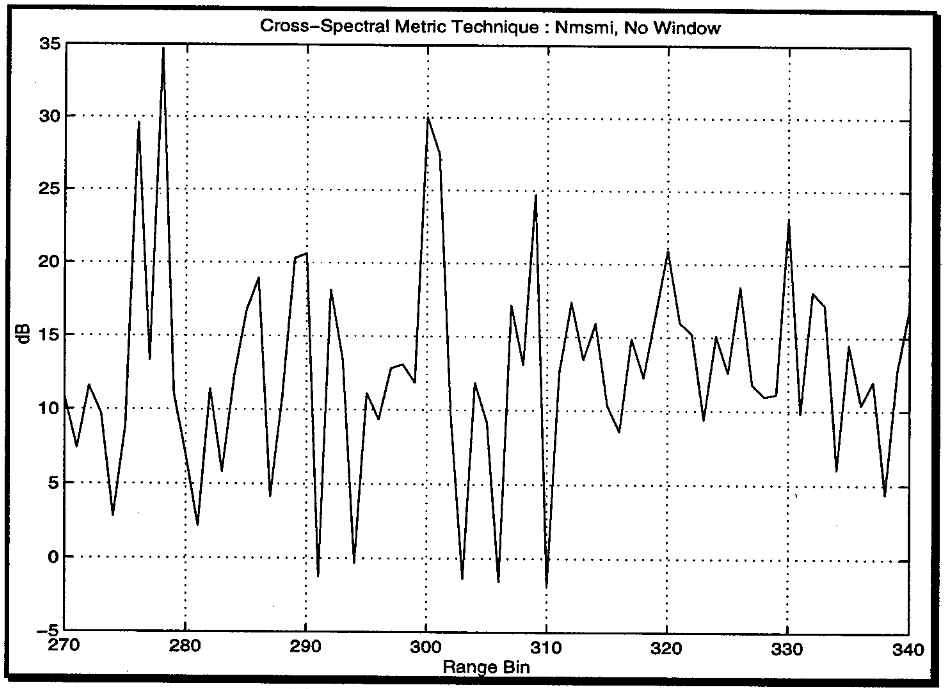

Figure 26. CSM output, r=29.

Note that the target was not detected in this case. After several trials, the target was successfully detected using a higher value for r.

Figure 27 shows the CSM output when r =160.

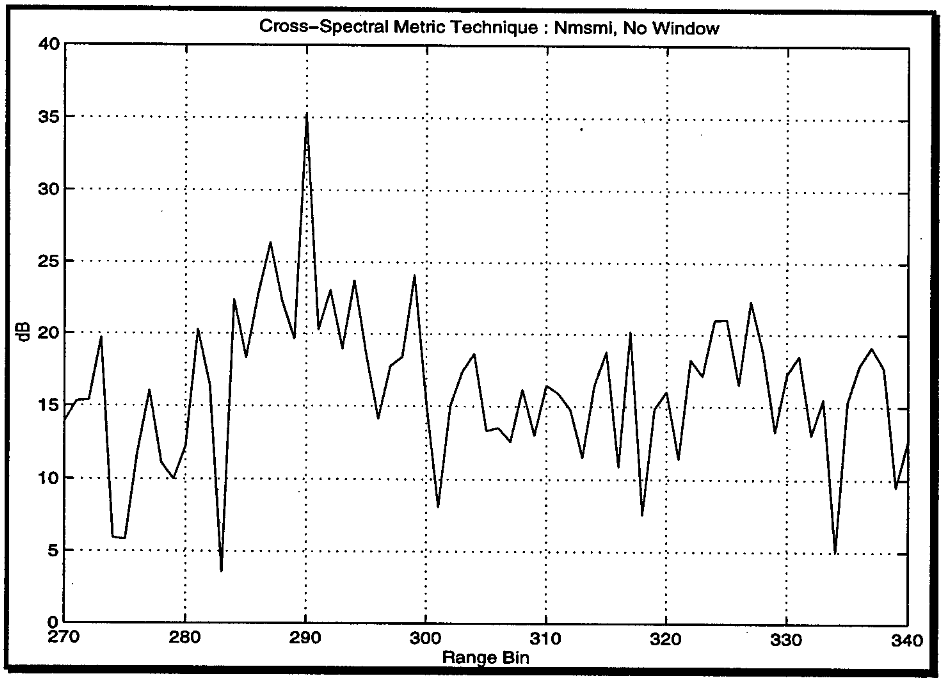

Figure 27. CSM output, r = 160.

In this case, the target shows up very clearly with a target gain of 8.87 dB. There is a loss of about 1.67 dB when compared to the PC method with the same number of components used.

A higher value of m achieves a better performance detection. In the case where r = 200, we obtained the best possible performance of CSM. Figure 28 shows the CSM output when r = 200.

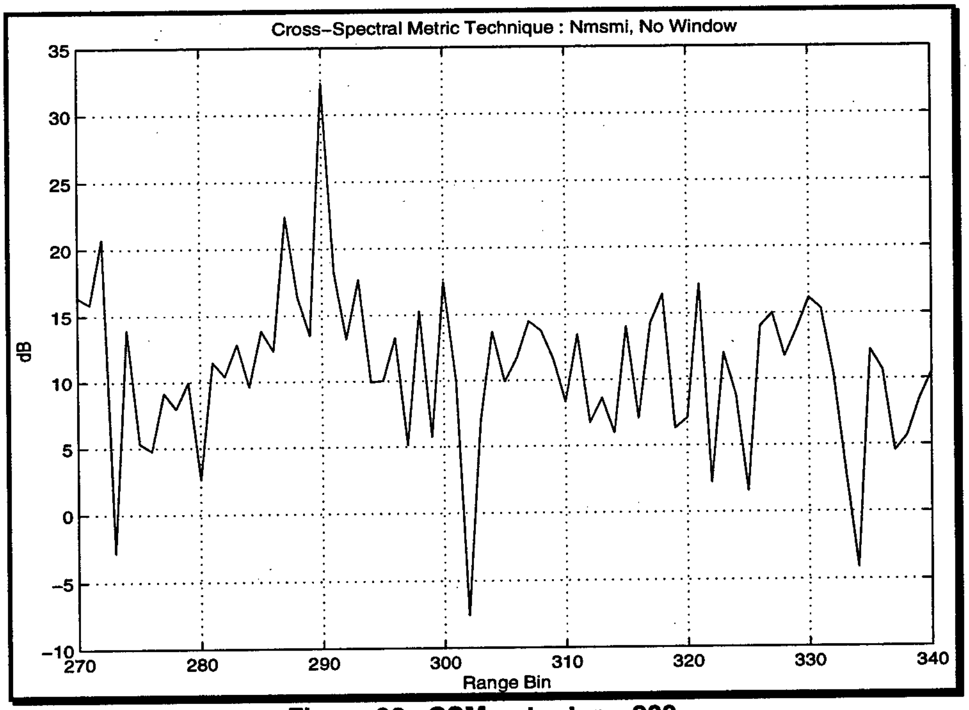

Figure 28. CSM output, r = 200.

In this case, the target shows up very clearly at range bin 290 with a target gain of 9.98 dB. There is an improvement of about 1.11 dB when compared to the case where r = 160. This is not a considerable improvement considering the added computational burden.

For this specific data set, the environment consists of rural terrain with many highways and speckles of urban clutter. This causes a lot of non-homogeneities in the radar returns. The assumption made regarding the iid nature of the secondary data, needed for the covariance computation, is no longer valid. Therefore, the theoretical values for rank reduction are no longer valid. Furthermore, it is not necessary to take all the 2K (K=224) samples for the covariance matrix estimation, since the secondary data further away from the cell under test is non-homogeneous in comparison, thereby degrading the secondary data. For these reasons, by just taking 2(100-160) samples (instead of 2(224)), the covariance matrix is better estimated and this results in better target detection.

Figure 29 shows the output of the factored time-space approach.

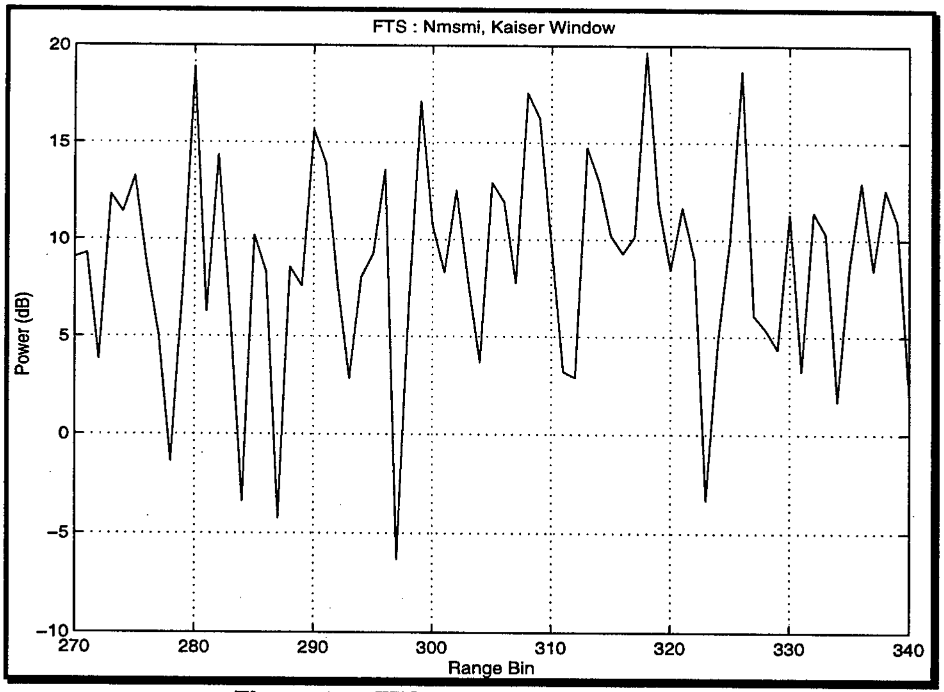

Figure 29. FTS output, Doppler bin 3.

The choice of the window used in Doppler processing is very important. Of all the windows used in this case, the FTS approach failed to identify the target at Doppler bin 3. Note that only 2*J (28) samples have been used in the covariance estimation.

Next the extended factored approach (EFA) has been examined using this set of data. The output of the EFA processor is shown in Fig. 30.

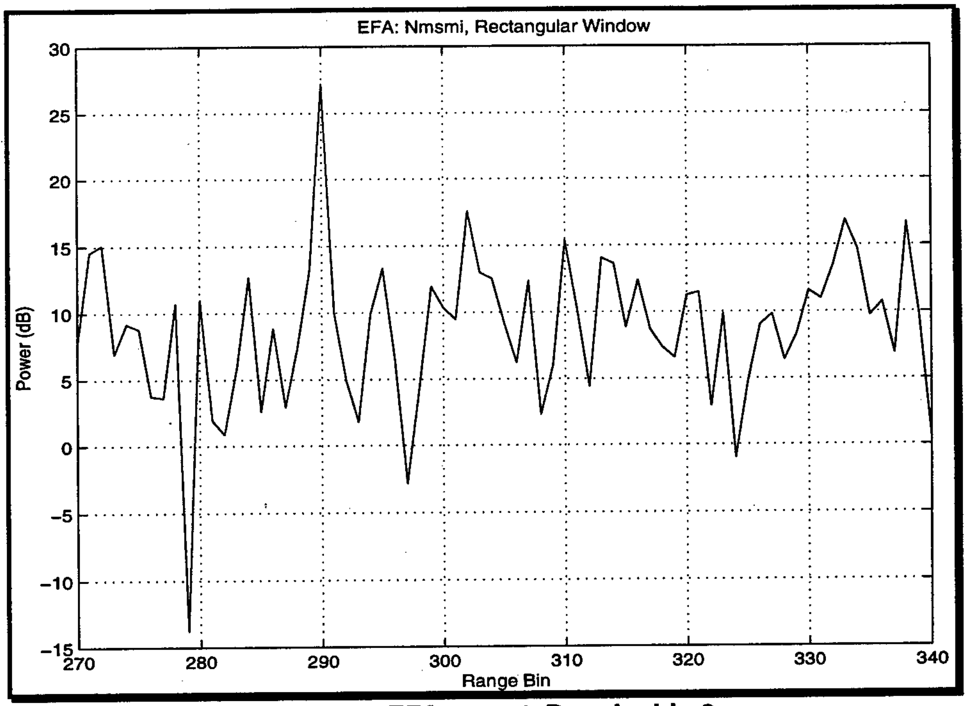

Figure 30. EFA output, Doppler bin 3.

The target shows up very clearly at range 290 with a target gain of 9.66 dB. Note that a rectangular window has been used in this case. 2JN (84) samples have been used in the covariance matrix estimation.

The joint domain localized (JDL) approach with a 3x3 LPR was analyzed next. The output of the JDL processor at Doppler bin 3 is shown in Fig. 31.

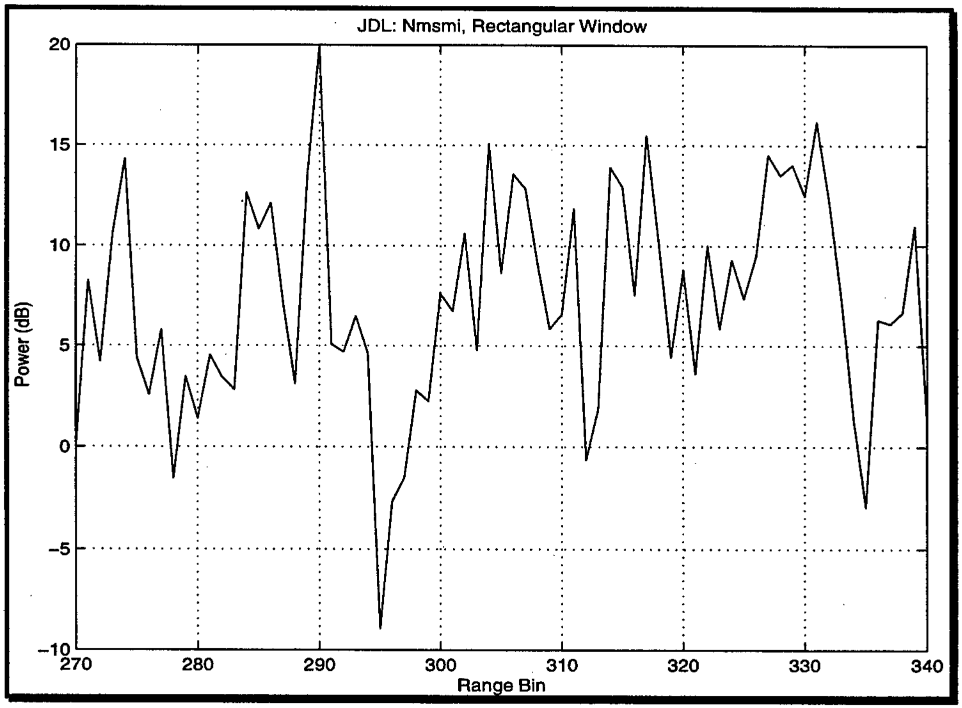

Figure 31. JDL output, Doppler bin 3.

Of all the windows, the rectangular window (uniform window) performed best. The target is well located at range bin 290 with a target gain of 3.80 dB. The number of secondary data needed is K = 2*(9) = 18.

Finally the 3-pulse adaptive displaced phase center aperture (ADPCA) was analyzed. Figure 32 shows the output of the ADPCA processor at Doppler bin 3.

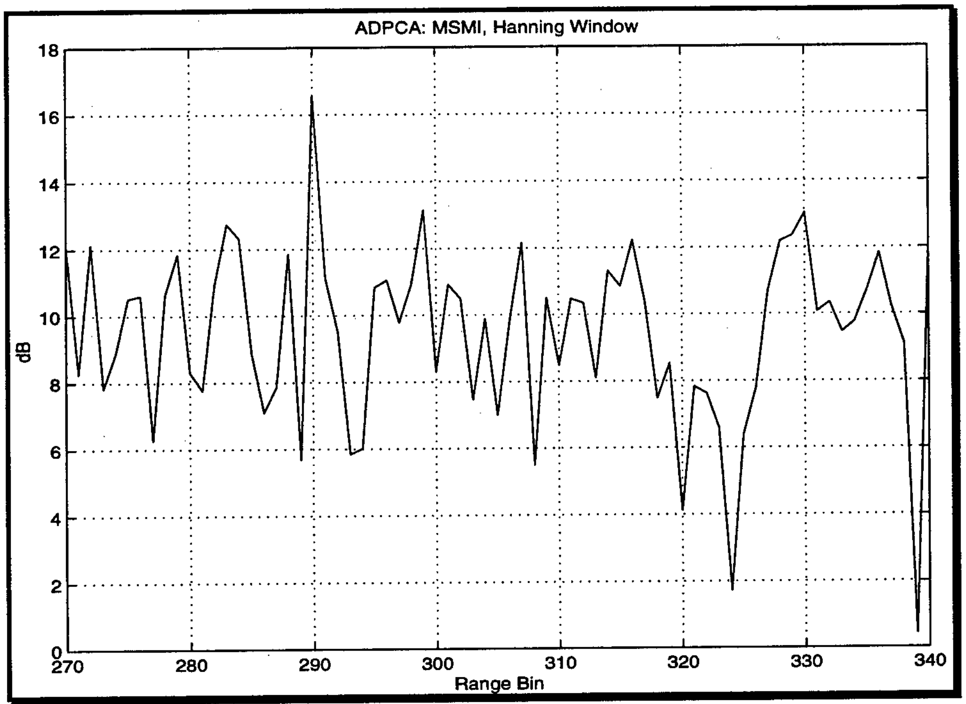

Figure 32. 3-pulse ADPCA output, Doppler bin 3.

Of all the windows tried, the Hanning window performed best. In this case, the target was detected with a 3.42 dB target gain. Note in this case that the number of secondary data used is the same as with EFA (84 samples).

From this brief analysis, it can be said that adaptive processing improves the detection performance of airborne radar significantly over conventional approaches. Recall that the digital beamformer failed to detect the target. Moreover, except for the FTS algorithms, all other STAP processors performed well with adequate sample support size. Considering the computational complexity of the algorithm as one requirement in selecting the right technique, EFA appears to be a good candidate for a fielded approach. However, this test is not enough to determine algorithm selection. More analysis is needed to achieve this goal.

Covariance estimation and window selection are two primary issues to be considered in each case. The non-homogeneous nature of the clutter returns violates the iid assumption made about the secondary data. Non-homogeneity detectors are being developed and appear to be good candidates for fielded STAP-based systems (see [18,19]). NHD is introduced next in this report.

Next we provide a table comparing the number of secondary data needed for covariance estimation, assuming 2K samples are used.

Table 3. STAP approaches and number of secondary data needed.

Approach	K
CFAR-AMF	449
Eigencanceler	300
Cross-Spectral Metric	300
Factored Time Space	28
Extended Factored Time Space	84
Joint Domain Localized	18
ADPCA	84

In the remainder of this report, we only consider the FTS, EFA, and JDL approaches. We introduce here the concept of the non-homogeneity detector. To achieve our comparative analysis, we first run the three mentioned algorithms and then include the non-homogeneity detector. Data is taken from files re050050, re050152 and re050280.

5.6.2 Data file: re050050

Initially, to test detection performance potential, we inject a target-like signal with known signal-to-noise ratio (amplitude = 2.4e-4) into the data at range bin 1000, 0° azimuth (angle bin 65) and Doppler bin 27. The datacube consists of $N = 32$ pulses, $J = 22$ channels and $L = 2500$ range cells, where L is the number of available range cells. With the above parameters, the target competes with very strong mainlobe and sidelobe ground clutter returns as can be seen in Fig. 30. We estimate the interference covariance matrix from a total of 2M secondary data, where M equals the dimension of the transformed data vector, using a symmetric sliding window [7]. We then select 4M secondary data, run the non-homogeneity detector (NHD) and then select the 2M most homogeneous data.

A plot of the angle-Doppler profile for this particular range is given in Fig. 33. It might be worth noting that the clutter ridge slope, β, is greater than unity, which results in a Doppler ambiguous case.

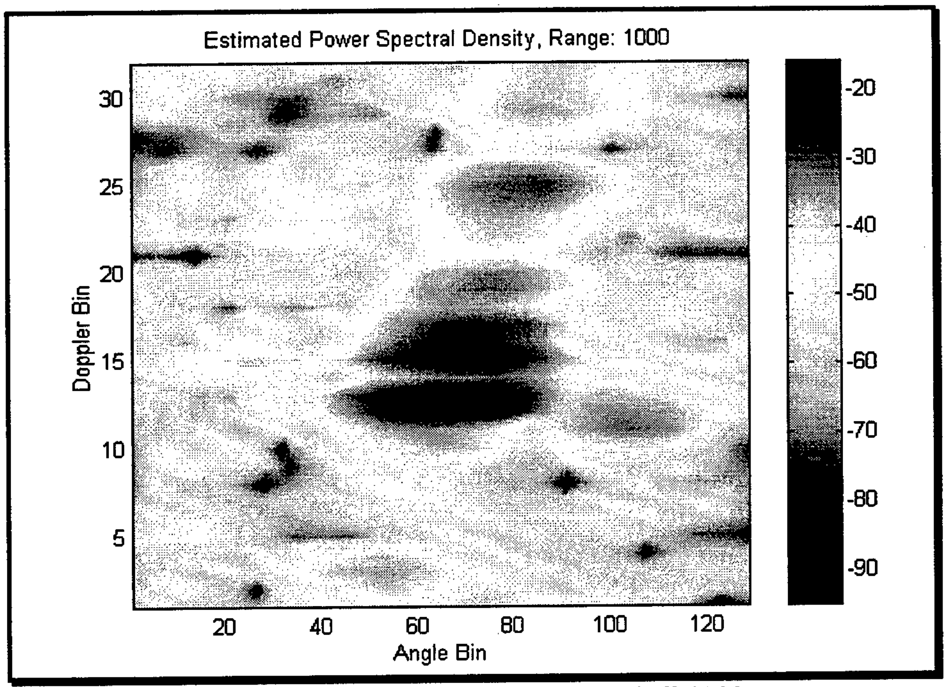

Figure 33. Estimated PSD, Range Cell 1000.

A cut at angle bin 65 (boresight) is shown below in Fig. 34. Note that in both Figs. 33 and 34, the data is taken before adaptation.

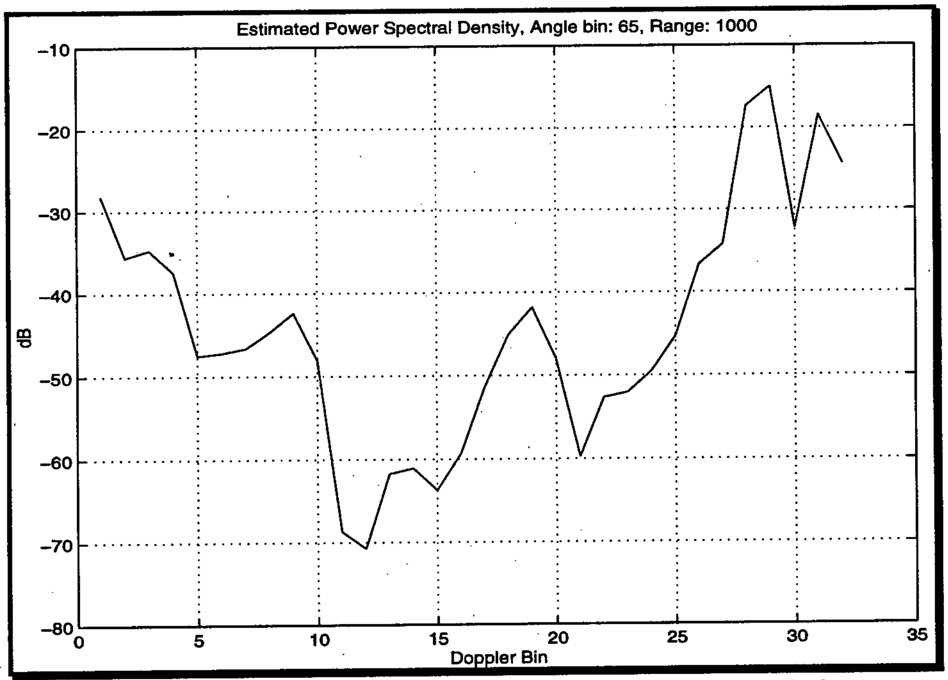

Figure 34. Cuts of the estimated PSD at angle bin 65 (0°).

From Fig. 34, it can be seen that the target competes with very strong sidelobe clutter. In this case the ratio between the mainbeam clutter and this sidelobe clutter is 19.22 dB.

Figure 35 shows the output of a digital beamformer, weighted with a Hanning window, plotted over range, at Doppler bin 27.

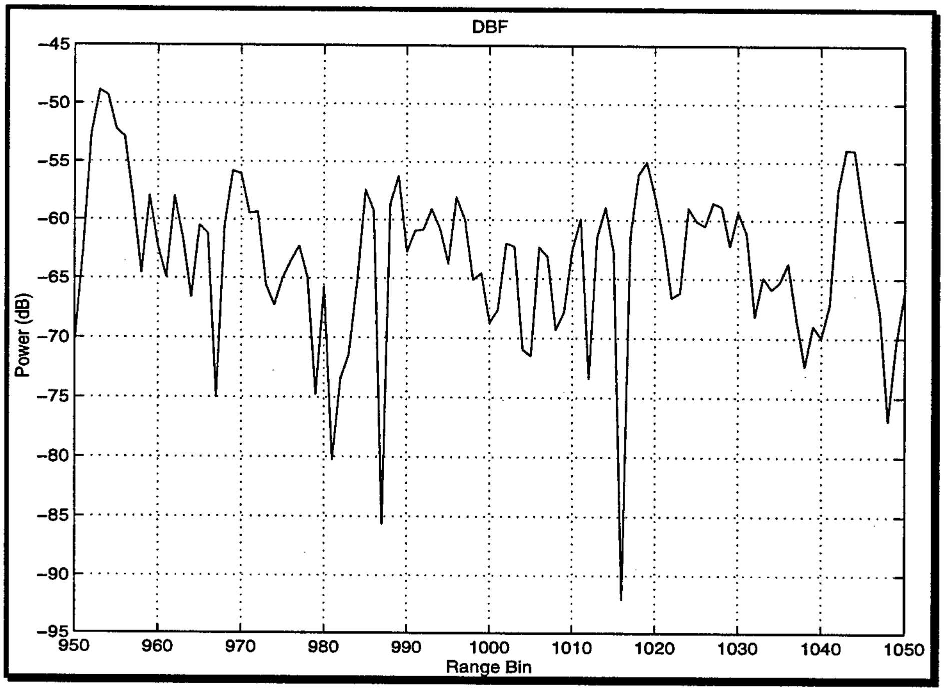

Figure 35. Output of digital beamformer.

This figure shows the digital beamformer was unable to uncover the target, located at range bin 1000, suggesting that more sophisticated STAP techniques such as FTS, EFA and JDL are needed.

Figure 36 shows the output of the factored time-space approach weighted with a Hanning window.

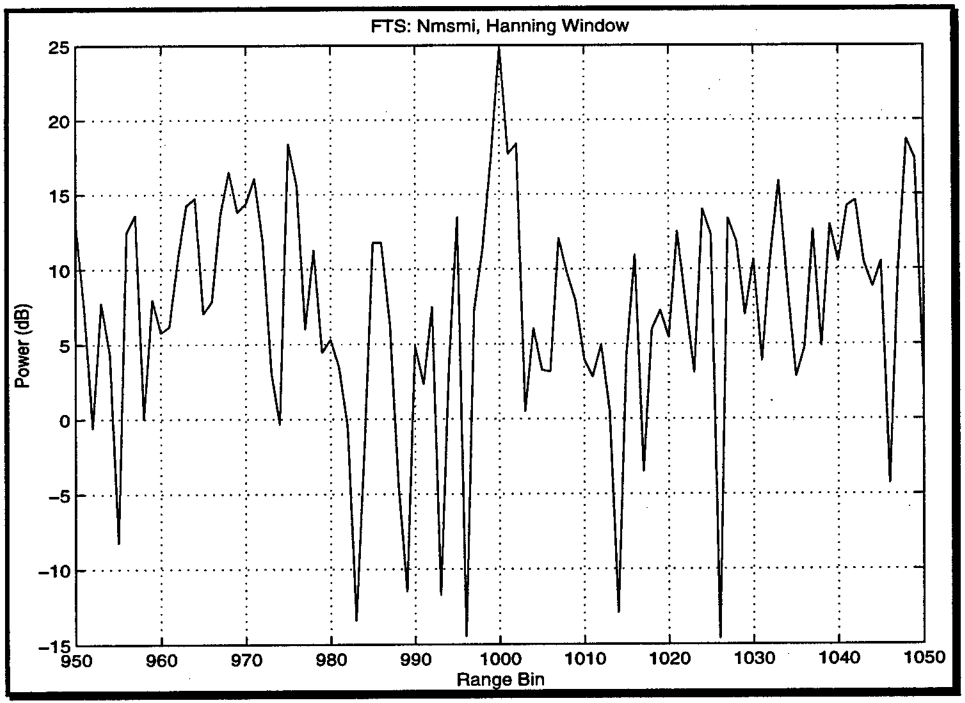

Figure 36. FTS output, Doppler bin 27.

The choice of the window used in Doppler processing is very important. Of all the windows used in this case, the Hanning window performed best for the FTS approach. The target was successfully detected with a target gain of 6.26 dB.

Applying NHD to the same data file, the output of the FTS processor at Doppler bin 27 is shown in Fig. 37. A Hanning window has also been applied to the data so that a fair comparative analysis could be made.

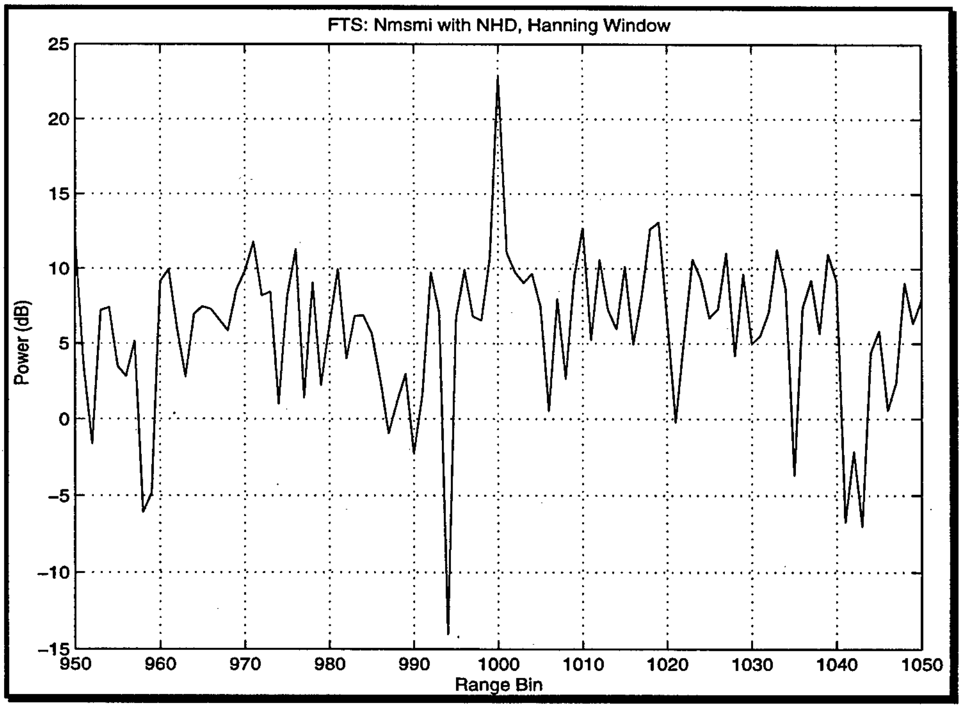

Figure 37. FTS output with NHD, Doppler bin 27.

Note in this case that the target gain is of 9.83 dB. This represents an improvement of 3.57 dB. This is a substantial improvement when detecting weak targets as considered in this analysis. Note that the added computational burden is acceptable, considering the significant improvement obtained in target detection.

Without NHD, 44 samples were used in the covariance matrix computation, whereas 88 samples have been used with the NHD. However, note that only the 44 most homogeneous secondary data have been used in the covariance matrix estimation.

Next the extended factored approach (EFA) has been examined using this set of data. The output of the EFA processor is shown in Fig. 38.

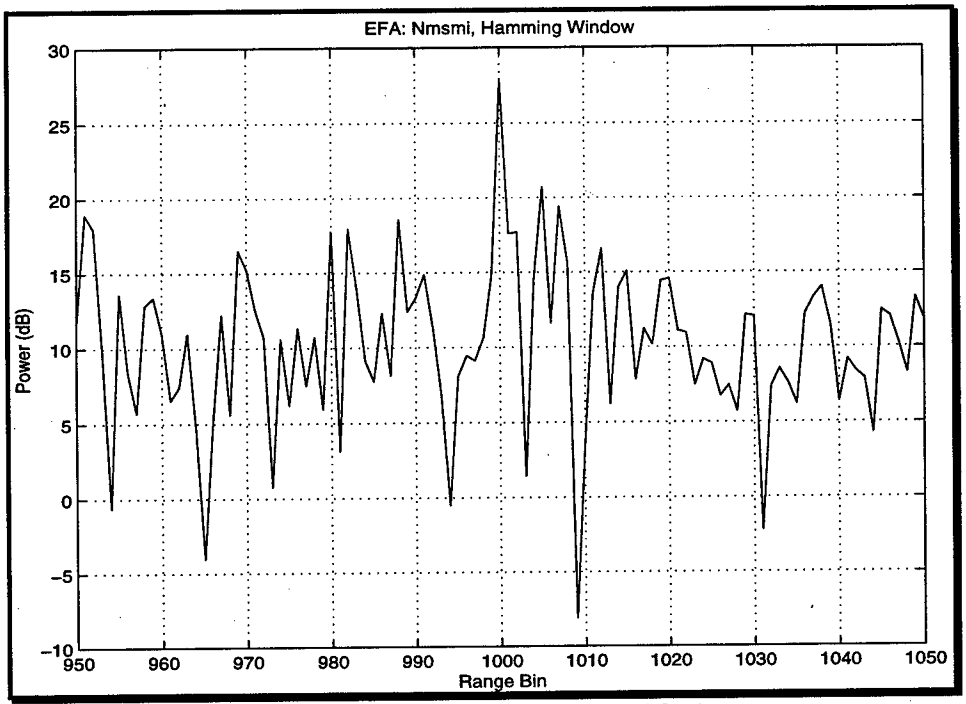

Figure 38. EFA output, Doppler bin 27.

From the above figure, it can be seen that the target shows up very clearly at range bin 1000 with a target gain of 7.24 dB. Note that a Hamming window has been used. In this case 132 (6J) samples (secondary data) have been used in the covariance matrix estimation.

When applying the NHD detector, there was no improvement in target detection as can be seen in Fig. 39.

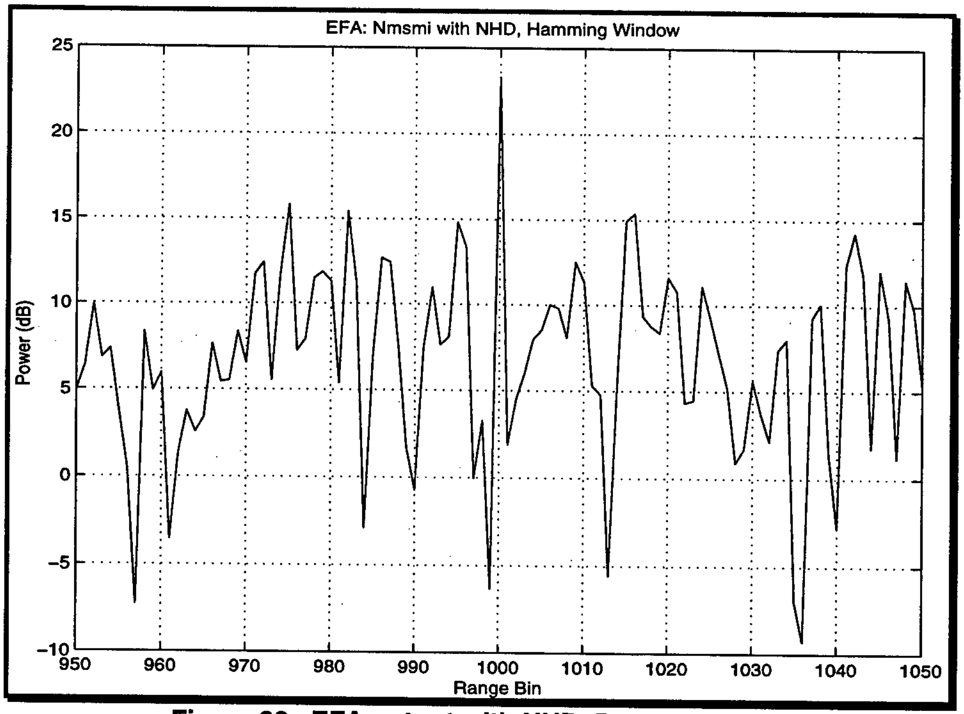

Figure 39. EFA output with NHD, Doppler bin 27.

The target shows up very clearly at range bin 1000 with a target gain of 7.50 dB, showing an improvement of only 0.26 dB. Later, we will give explanations as to why the NHD detector did not perform significantly better than the symmetric window.

The joint domain localized (JDL) approach with a 3x3 LPR was analyzed next. The output of the JDL processor at Doppler bin 27 is shown in Fig. 40.

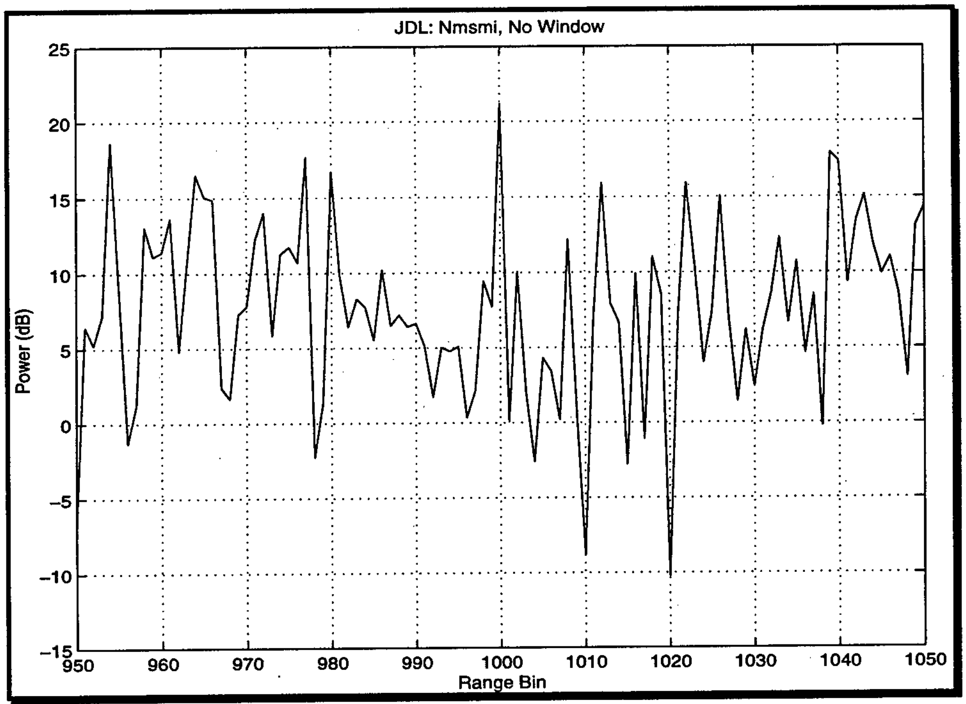

Figure 40. JDL output, Doppler bin 27.

From the above figure, it can be seen that the target shows up at range bin 1000 with a target gain of 3.22 dB. We point out that of all the windows used the rectangular window performed best. Other windows actually have greatly deteriorated the performance of the JDL processor. For these reasons, the choice of a window is very important in STAP processing. Recall that only 18 secondary data have been used in this case.

The purpose of our analysis is to determine the effectiveness of the proposed techniques in detecting the target while canceling the interference. We decided to keep the 2M criteria for our sample support size in the covariance matrix estimation. Another approach would have involved using the largest number of available secondary data. In this report, the largest number was that of the EFA approach (6J = 132). An analysis could have been done using 132 samples for all three proposed techniques.

When applying the NHD detector, there was improvement in target detection as can be seen in Fig. 41.

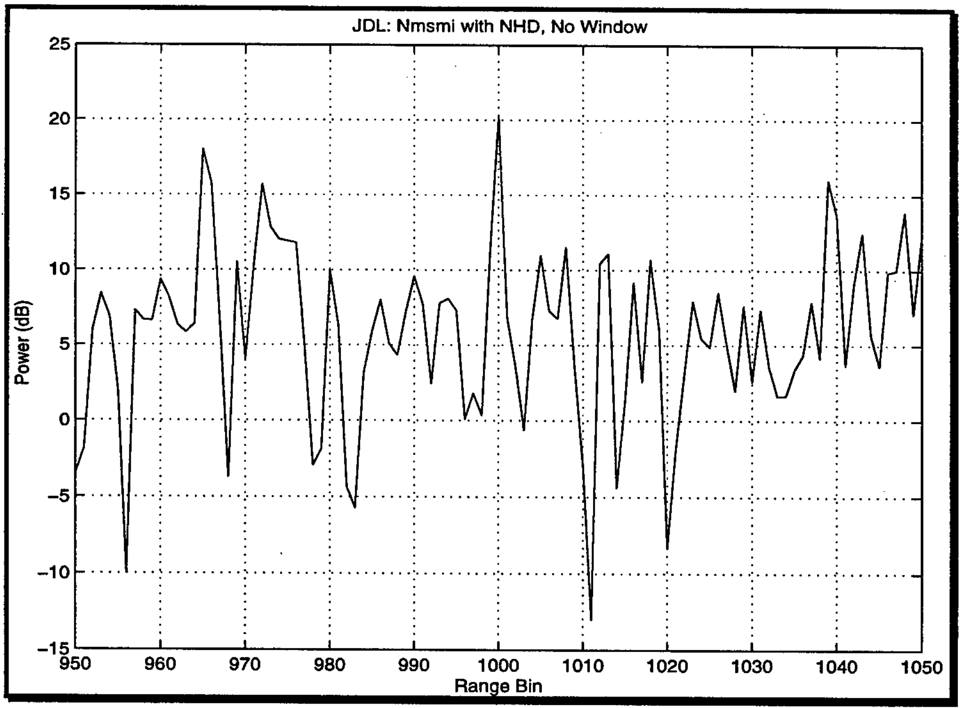

Figure 41. JDL output with NHD, Doppler bin 27.

The target shows up clearly at range bin 1000 with a target gain of 4.31 dB, exhibiting a gain of 1.09 dB in performance.

For this specific data set, the environment consists of rural terrain and sea clutter at far away ranges (see Fig. 42 below).

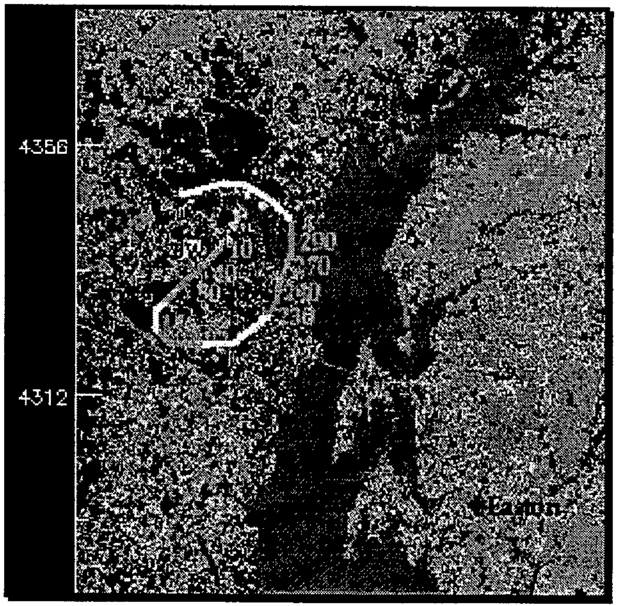

Figure 42. MCARM Flight Path, Flight 5, Cycle e.

Figure 42 by itself is not enough in analyzing the non-homogeneity of the clutter returns. The GIP defined in Equation (31) is a better indicator of the homogeneity of the data. To further analyze the homogeneity of the data, the GIP for all three techniques is plotted in the following figures.

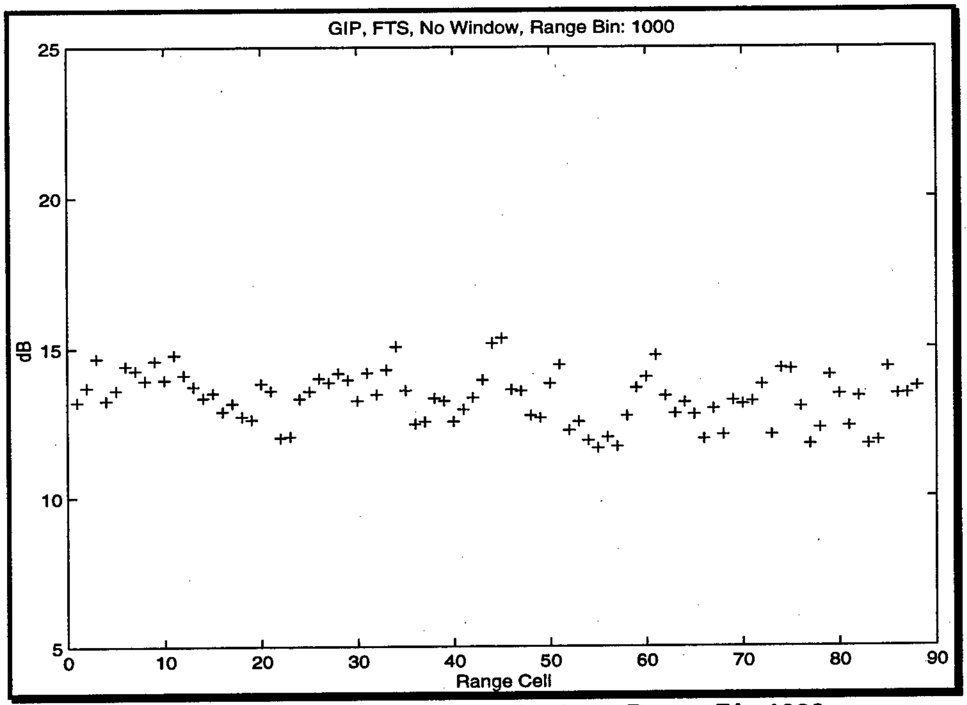

Figure 43. GIP, FTS, Symmetric about Range Bin 1000.

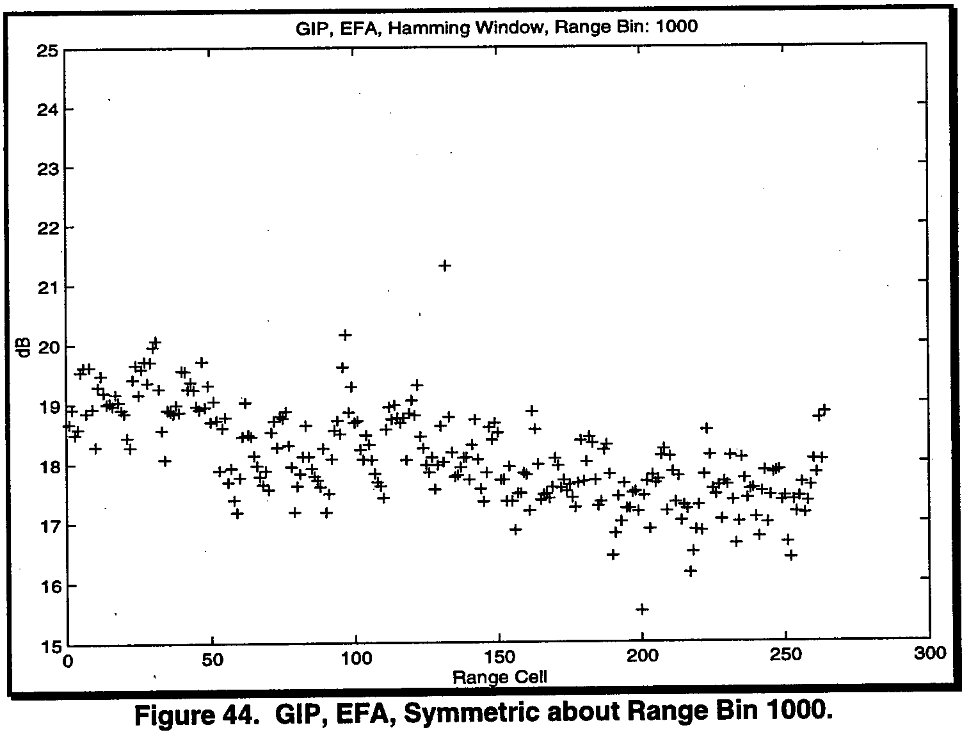

Figure 44. GIP, EFA, Symmetric about Range Bin 1000.

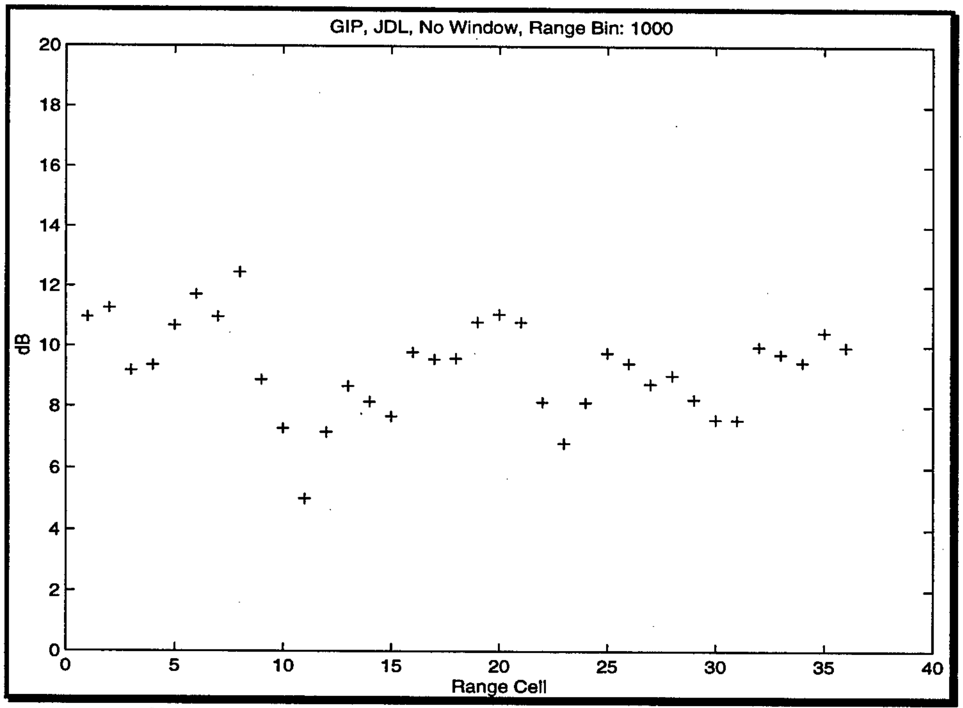

Figure 45. GIP, JDL, Symmetric about Range Bin 1000.

Secondary data was selected symmetrically around range bin 1000. The caption "Range Cell" in the above figures denotes all of the secondary data considered for the NHD selection and then for the covariance matrix computation for the three techniques. Note from the above figures that the data is pretty homogeneous. The deviation of the GIP over range is very minimal. For these reasons, use of the NHD detector does not appear to make a significant contribution to the target detection.

5.6.3 Data file: re050152

In this case, no artificial target was injected. File re050152 employs a moving target simulator (MTS) in the radar field of view to simulate a moving point source. This file has data of medium PRF (prf = 1984 Hz). The datacube consists of N = 128 pulses, J = 22 channels and L = 630 range cells. The MTS, located roughly 36 nautical miles from the airborne platform, is matched to the MCARM waveform and generates five tones of varying amplitude. Tones are spaced 200 Hz (13 Doppler bins based on a 15.5 Hz/bin Doppler resolution (prf/128)) to simulate target velocity. The MTS and transmit azimuth are not coincident.

A plot of the angle-Doppler profile for this particular range is shown in Fig. 46.

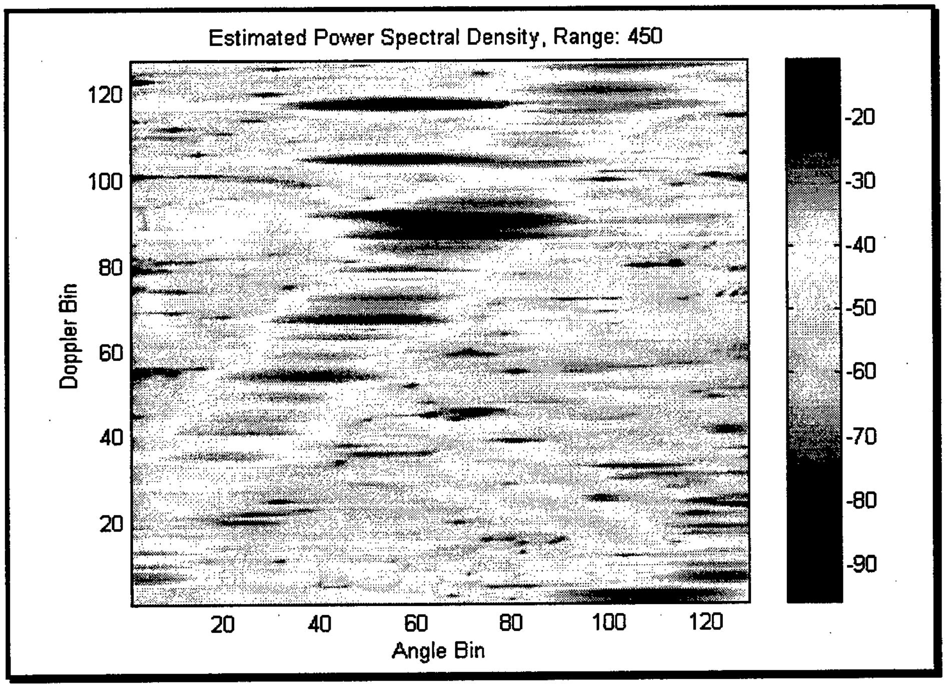

Figure 46. Estimated PSD, Range Cell 450.

The 5 tones are actually visible in range bins 448-452. However, they are best seen at range bin 450.

To again demonstrate the need for STAP processing, we first run this set of data through a digital beamformer. Figure 47 shows the output of a digital beamformer weighted with a Hanning window. The arrows show the locations of the five tones.

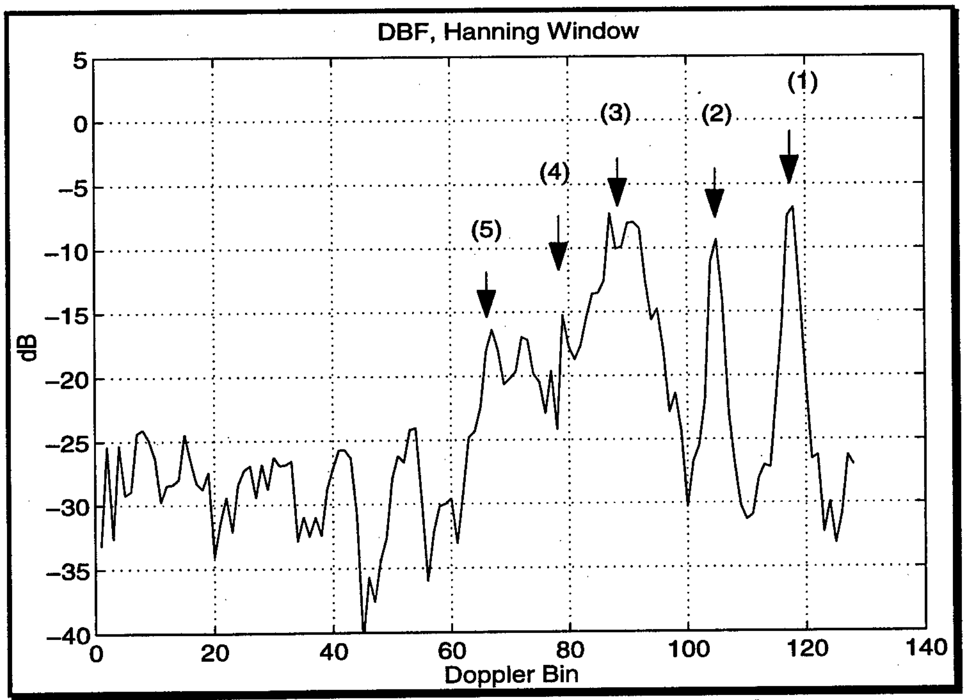

Figure 47. Output of digital beamformer, Range Bin: 450.

Note from this figure that the digital beamformer was not able to detect all the 5 tones. Only 2 tones are barely distinguished from the disturbance. The remaining 3 tones are completely embedded in clutter, thus establishing a STAP processing requirement.

Figure 48 shows the output of the factored time-space approach weighted with a Kaiser window.

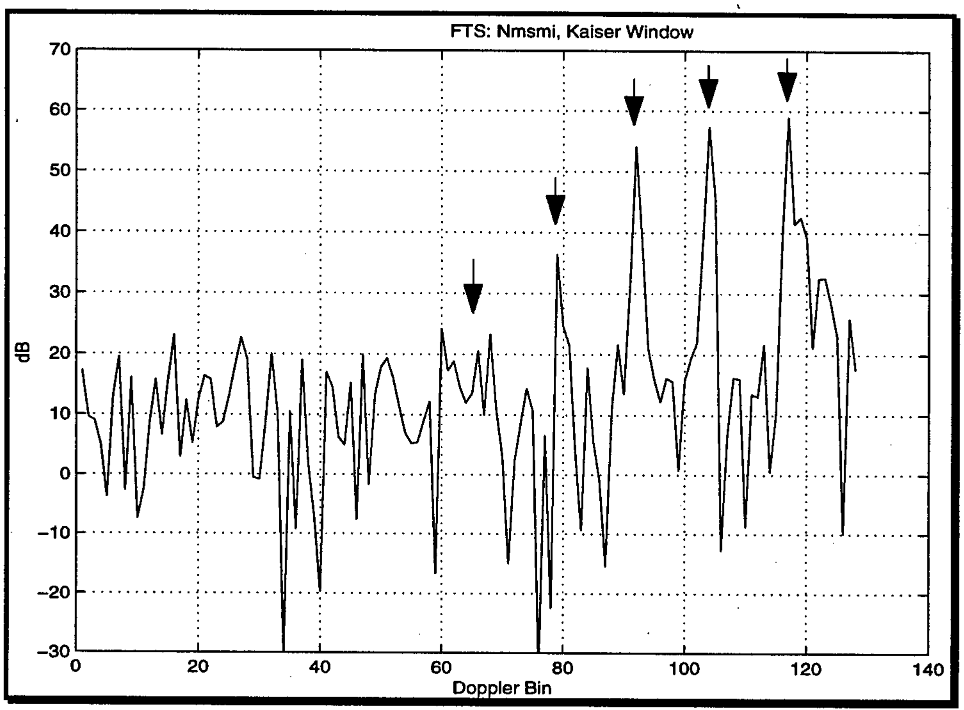

Figure 48. FTS output, Range bin: 450.

A note at this stage should be made about the number of samples needed in the covariance matrix computation. Again, we calculate the interference covariance matrix from a total of 2M secondary data, where M equals the dimension of the transformed data vector, using a symmetric sliding window [6]. However, preliminary results obtained using only 2M secondary data were not as good as had been expected. For these reasons, we chose to use 4M secondary data in the covariance matrix estimation, for all three algorithms. This gave acceptable results and did not cause any problems with the availability of the secondary data (L = 630). We then select 8M secondary data, run the NHD detector and then select the 4M most homogeneous data.

Note from Fig. 48 that the first 4 tones are clearly distinguished from the clutter. The 5-th tone, however, is still buried in the clutter. The result shown in Fig. 48 is the optimum. A Kaiser window with parameter k1= log10(22) and k2 = log10(128) was used.

Applying NHD to the same data file using a Kaiser window with the same parameters, the output of the FTS processor at range bin 450 is shown in Fig. 49.

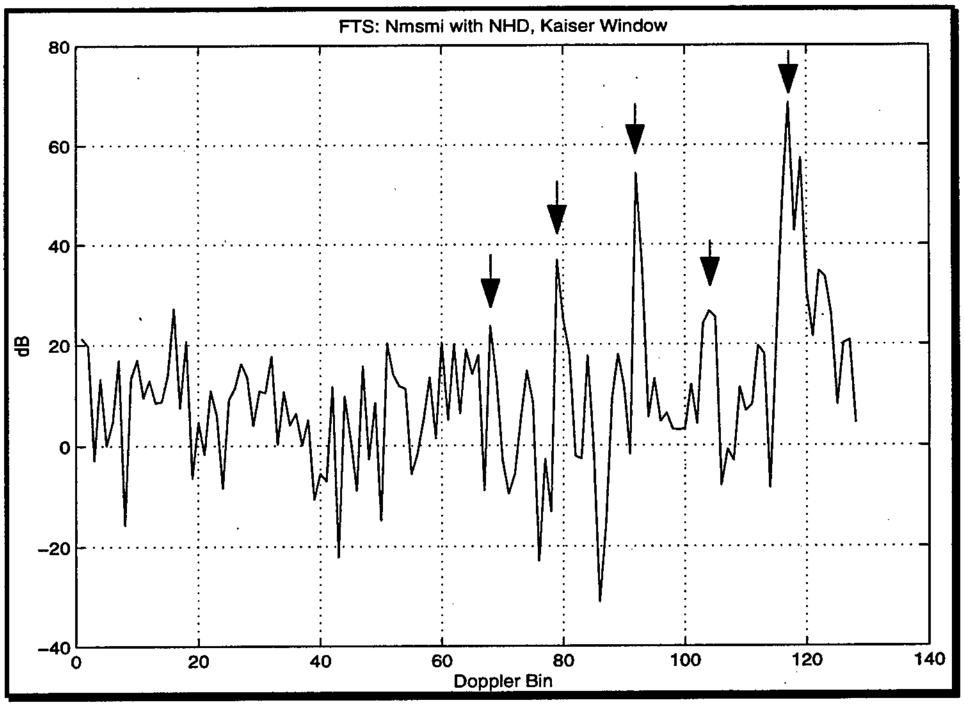

Figure 49. FTS output with NHD, Range bin: 450.

Note in this case that the NHD detector did not improve the detection of all 5 tones. Whereas the 5-th tone is a bit enhanced, the 2-nd tone is almost nulled out. Note also that the 1-st tone has a false alarm competing with it. As we observed earlier, the terrain over which acquisition 152 was taken is fairly homogeneous, thus making the NHD detector unneeded.

Next the extended factored approach (EFA) was examined using this set of data with a sample support size of 6J (132) range cells. The output of the EFA processor is shown in Fig. 50.

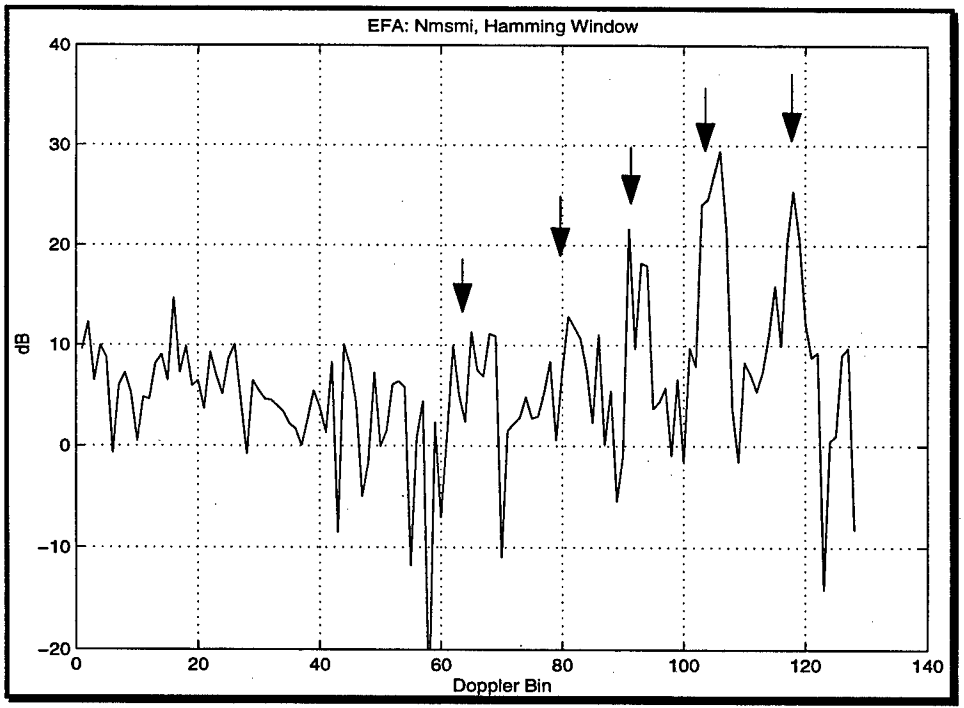

Figure 50. EFA output, Range bin: 450.

From the above figure, it can be seen that the EFA processor did not perform as well as the FTS processor.

When applying the NHD detector, there was very little improvement in target detection as can be seen in Fig. 51.

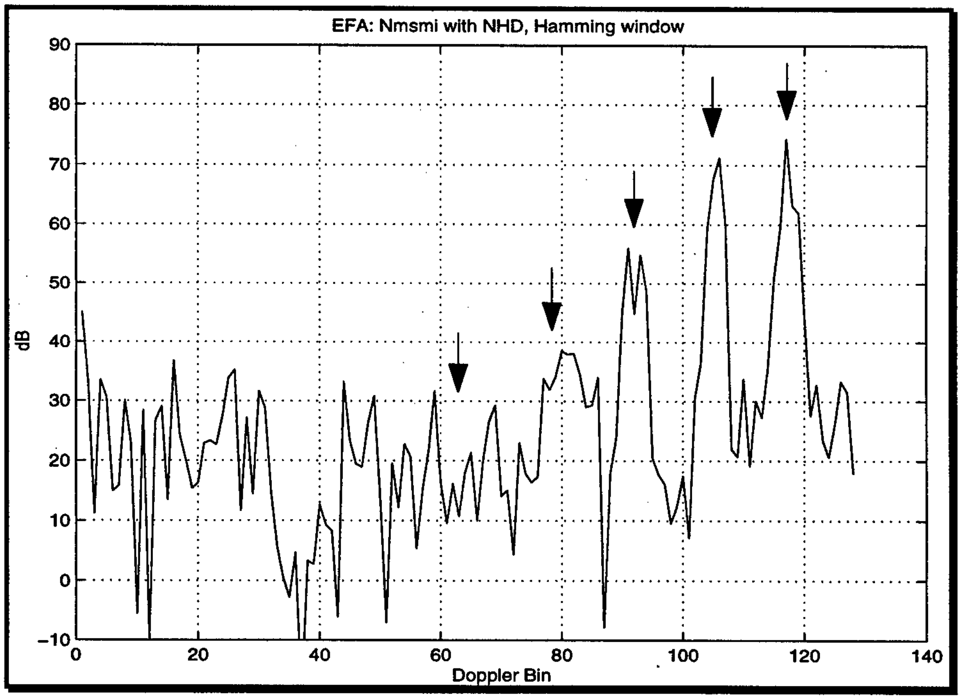

Figure 51. EFA output with NHD, Range bin: 450.

Note that in both cases a Hamming window has been used.

The joint domain localized (JDL) approach with a 3x3 LPR was analyzed next. The output of the JDL processor at range bin 450 is shown in Fig. 52.

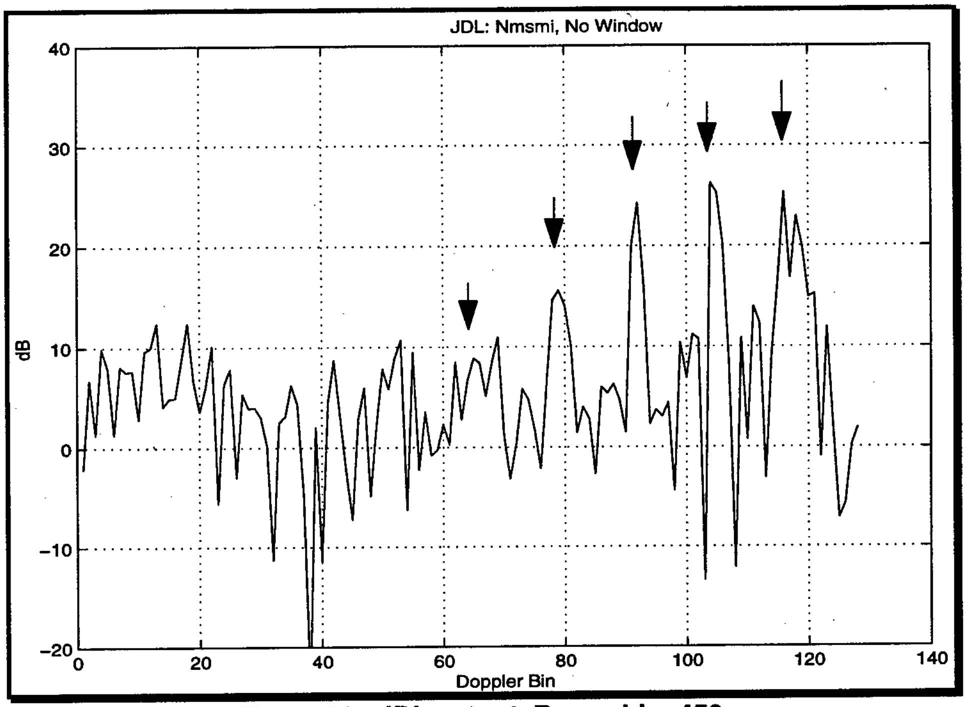

Figure 52. JDL output, Range bin: 450.

From the above figure, it can be seen that the JDL processor clearly uncovered 3 tones. The 5-th tone is still buried in the clutter whereas the 1-st tone is competing very strongly with a false alarm.

Applying the NHD detector, the output of the JDL processor is shown in Fig. 53.

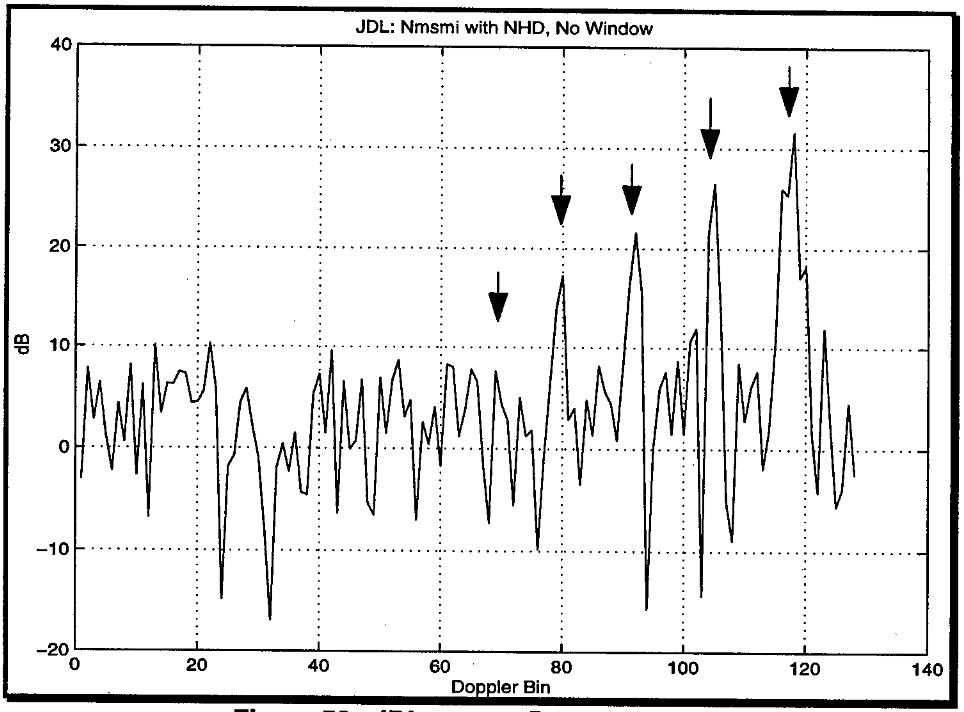

Figure 53. JDL output, Range bin: 450.

In this case, the JDL algorithm with NHD did a slightly better job in detecting 5 tones. Note that the 5-th tone is still buried in the clutter. The 1-st tone is, however, better detected.

Considering the number of secondary data needed and the amount of computations needed, the JDL processor in conjunction with NHD performed the best. Note also that of all the windows that we tried, the rectangular window yielded the best results.

In the next three figures, we plot the GIP for the three different techniques. This is done to determine the impact of the NHD detector on the proposed techniques.

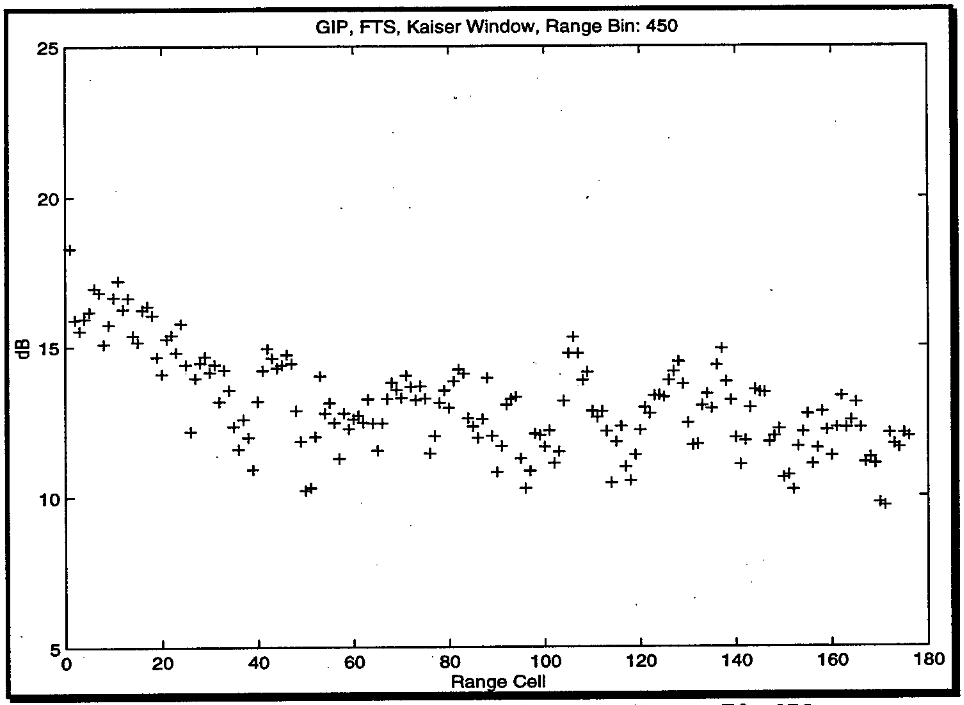

Figure 54. GIP, FTS, Symmetric about Range Bin 450.

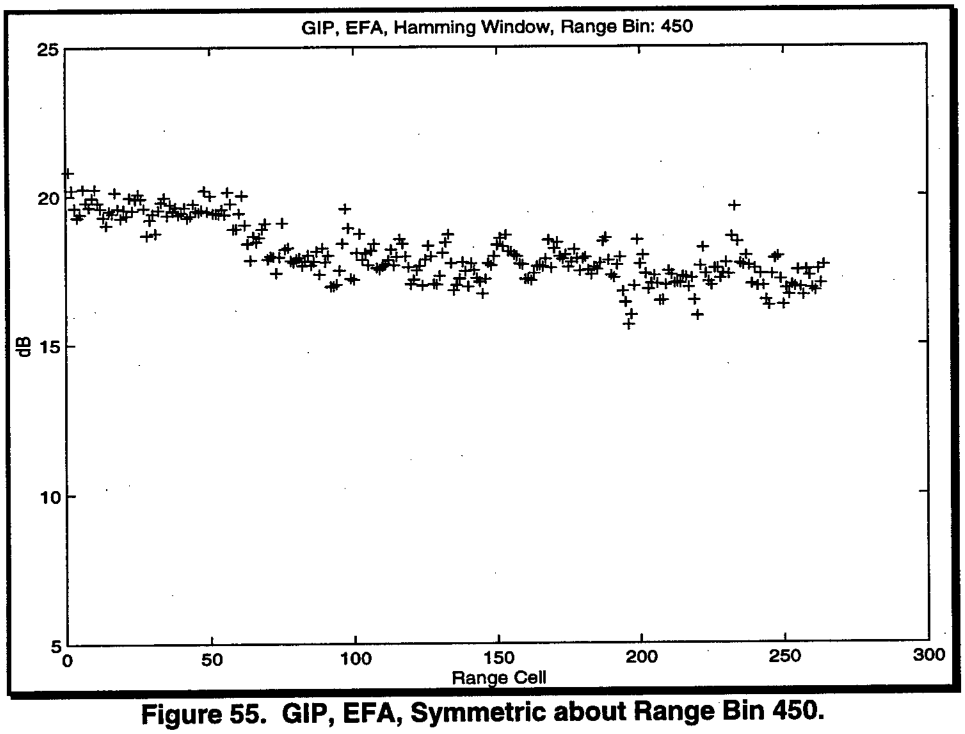

Figure 55. GIP, EFA, Symmetric about Range Bin 450.

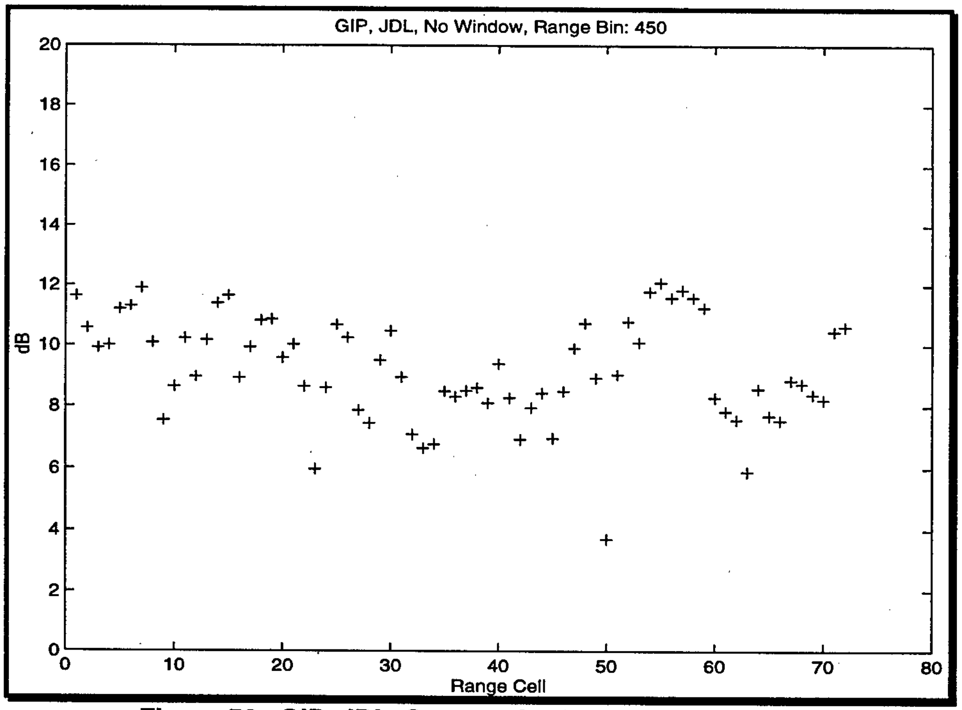

Figure 56. GIP, JDL, Symmetric about Range Bin 450.

As we mentioned earlier, the GIP determines how homogeneous the secondary data is, with respect to the primary data. Just as in the previous case, secondary data was selected symmetrically around range bin 450. The caption "Range Cell" in the above figures denotes all of the secondary data considered for the NHD selection and then for the covariance matrix computation for the three techniques. Note from figures 53 through 56 that the data is pretty homogeneous. The deviation of the GIP over range is very minimal. For these reasons, use of the NHD detector does not appear to make a significant contribution to the target detection.

5.6.4 Data file: re050280

In this case, we inject a target-like signal with known signal-to-noise ratio (amplitude = 5e-3) into the data at range bin 290, 0° azimuth (angle bin 65) and Doppler bin 7. The datacube consists of $N = 128$ pulses, $J=22$ channels and $L = 630$ range cells. With the above parameters, the target competes with very strong mainlobe and sidelobe ground clutter returns. We calculate the interference covariance matrix from a total of 2M secondary data, where M equals the dimension of the transformed data vector, using a symmetric sliding window [6]. We then select 4M secondary data, run the non-homogeneity detector and then select the 2M most homogeneous data.

A plot of the angle-Doppler profile for this particular range is given in Fig. 57.

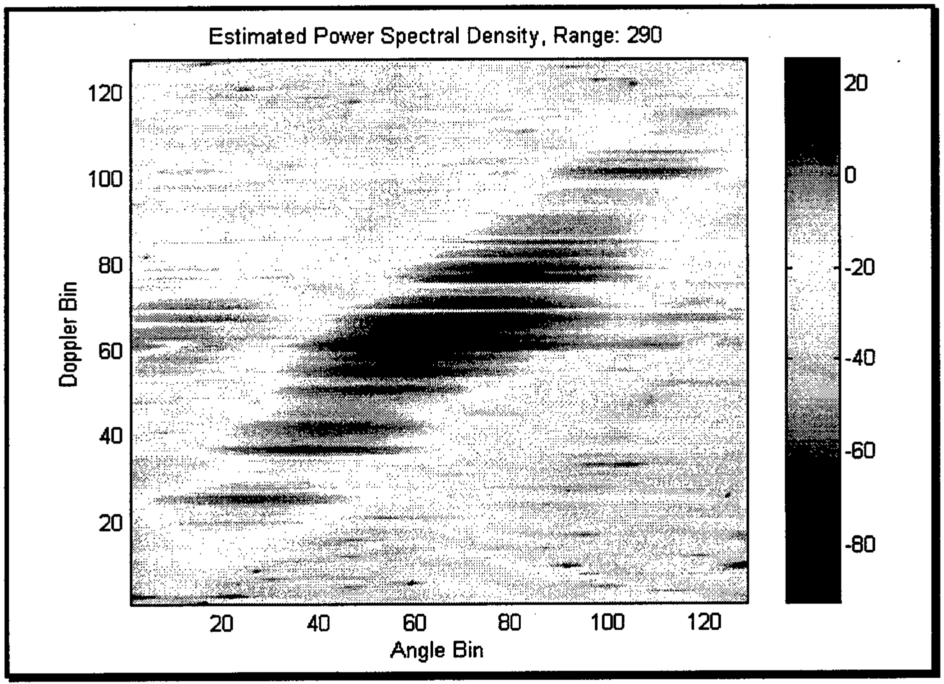

Figure 57. Estimated PSD, Range Cell 290.

Figure 58 shows the output of a digital beamformer weighted with a Hanning window.

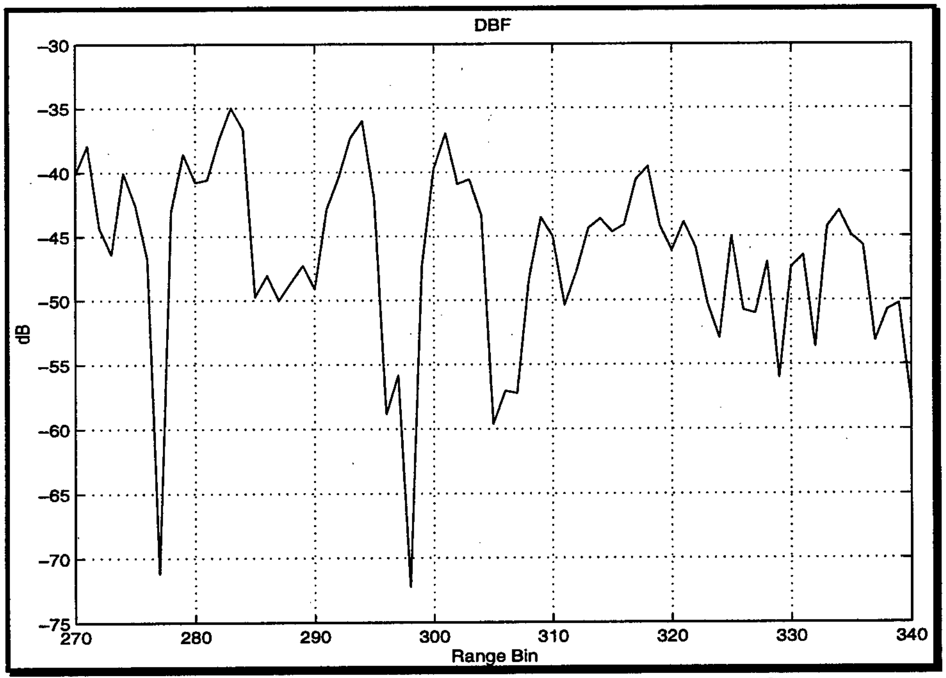

Figure 58. Output of digital beamformer, Doppler Bin: 7.

Note from this figure that the digital beamformer was not able to uncover the target, located at range 290.

Figure 59 shows the output of the factored time-space approach, weighted with a rectangular window (a uniform window was applied).

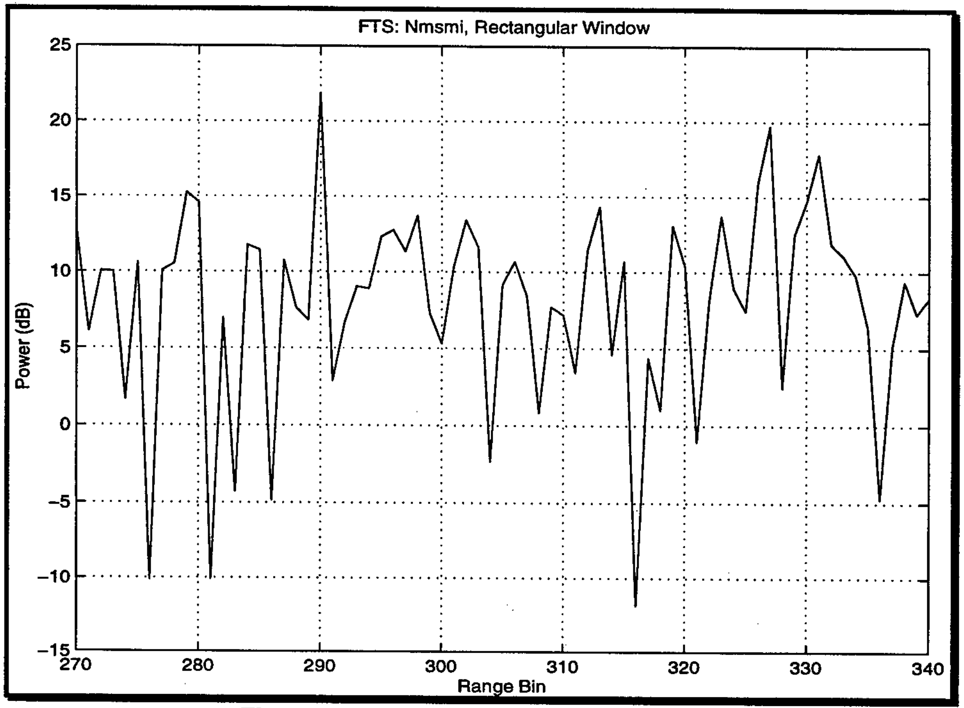

Figure 59. FTS output, Doppler bin: 7.

The choice of the window used in Doppler processing is very important. Of all the windows used in this case, the rectangular window performed best for the FTS approach. The target was successfully detected with a target gain of 2.18 dB.

Applying NHD to the same data file, the output of the FTS processor at Doppler bin 7 is shown in Fig. 60.

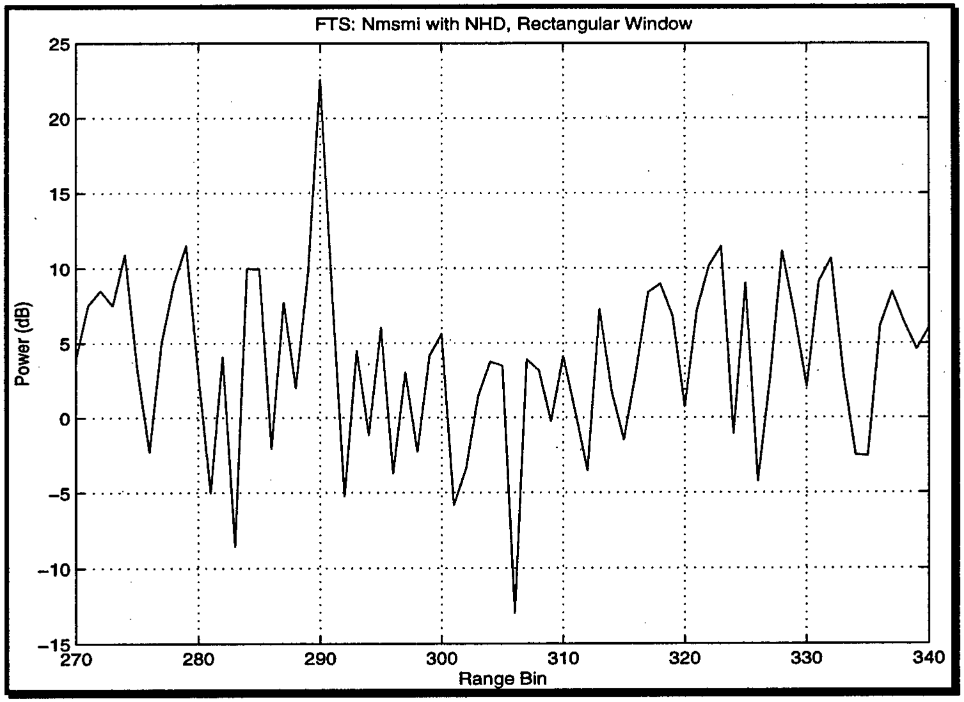

Figure 60. FTS output with NHD, Doppler bin: 7.

Note in this case that the target at range bin 290 is better detected with a target gain of 11.06 dB, which in turn results in an improvement of 8.88 dB. This is significant considering the types of targets being detected (low cross-section targets).

Next the extended factored approach (EFA) has been examined using this set of data. The output of the EFA processor is shown in Fig. 61.

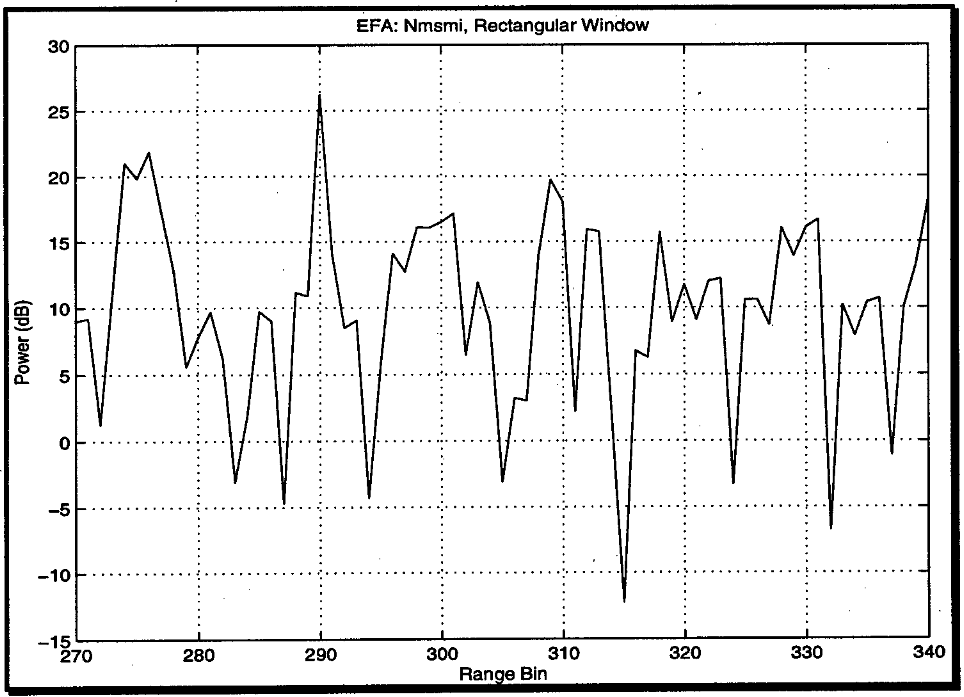

Figure 61. EFA output, Doppler bin 7.

From the above figure, it can be seen that the EFA processor detected the target at range bin 290 with a target gain of 4.37 dB. Again, of all the windows tried, the rectangular window performed the best.

When applying the NHD detector, the output of the EFA processor at Doppler bin 7 is shown in Fig. 62.

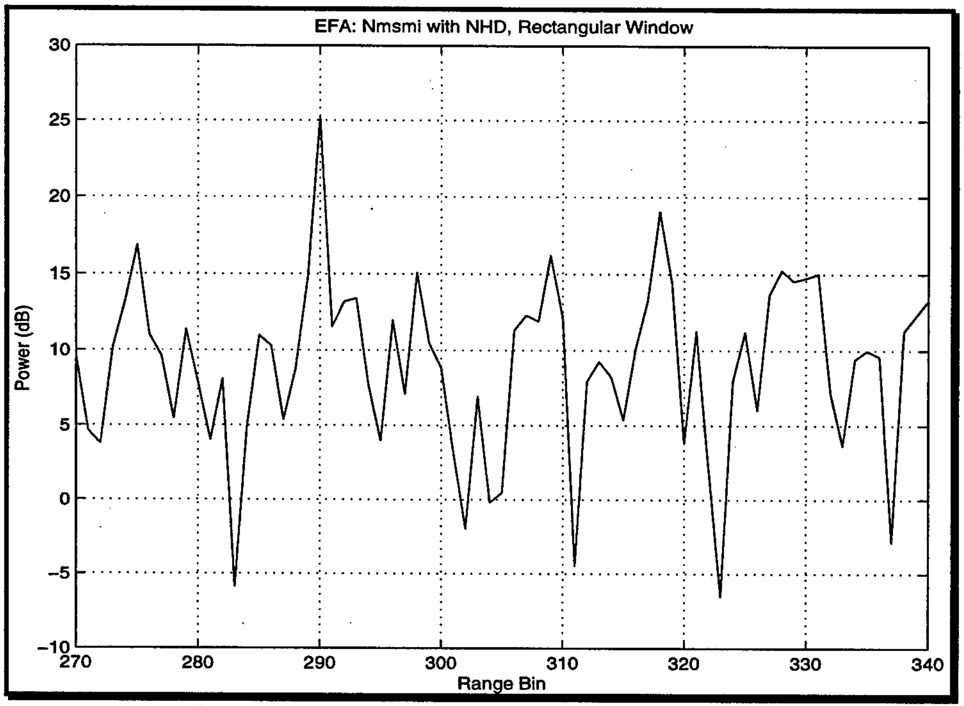

Figure 62. EFA output with NHD, Doppler bin 7.

Note in this case that the target at range bin 290 is better detected with a target gain of 6.13 dB, resulting in a small improvement of 1.76 dB.

The joint domain localized (JDL) approach with a 3x3 LPR was analyzed next. The output of the JDL processor at Doppler bin 7 is shown in Fig. 63.

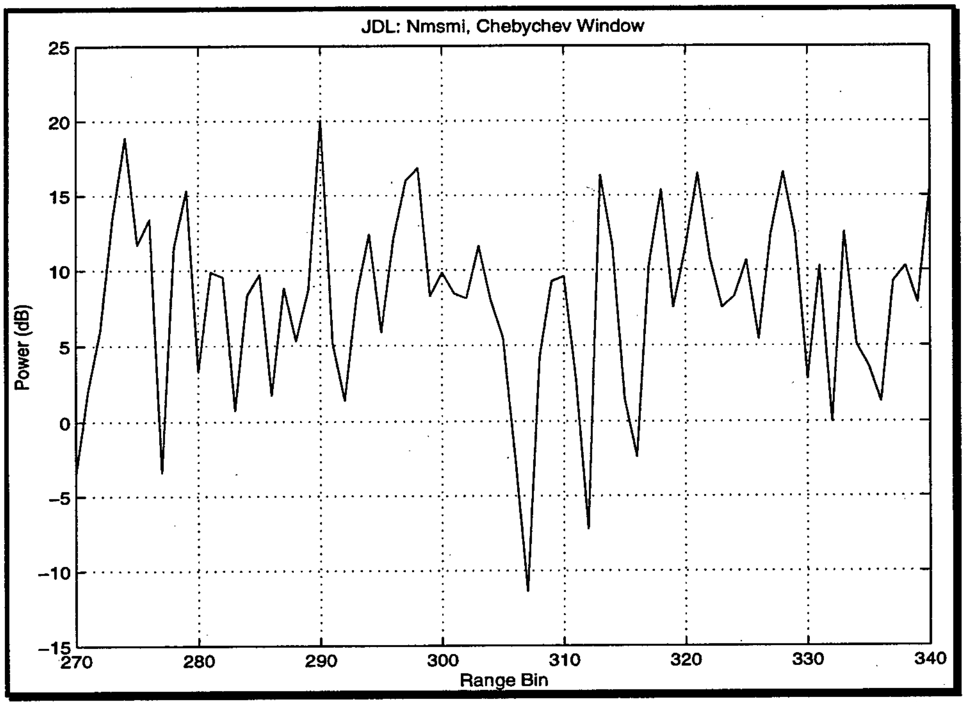

Figure 63. JDL output, Doppler bin 7.

The JDL processor successfully uncovered the target at range bin 290 with a target gain of 1.16 dB.

The output of the JDL processor at Doppler bin 7, when applying the NHD detector, is shown in Fig. 64.

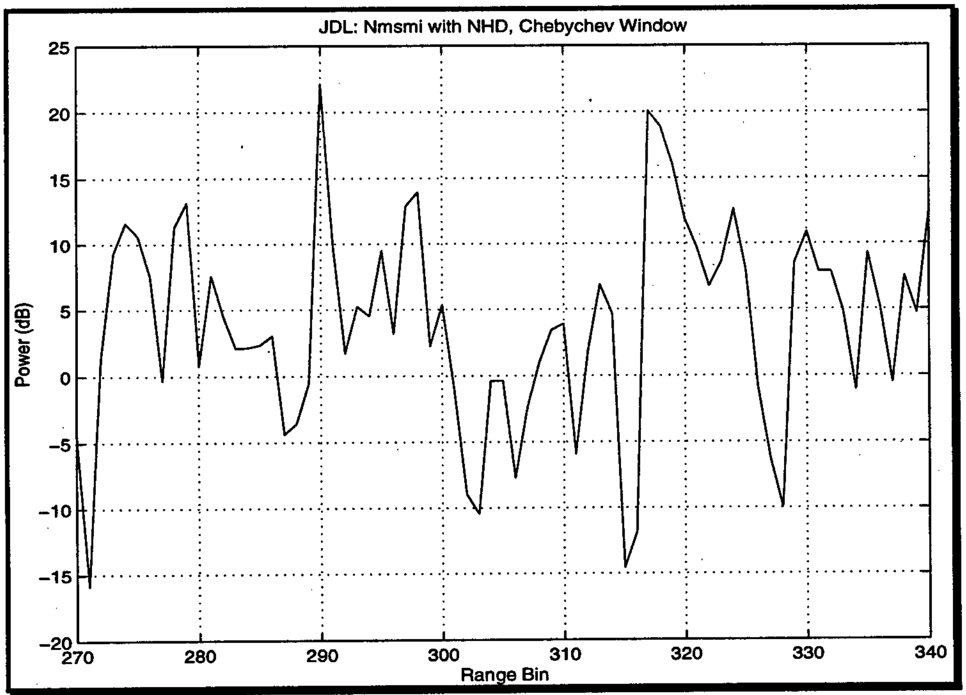

Figure 64. JDL output with NHD, Doppler bin 7.

Note in this case that the target at range bin 290 is better detected. The ratio between the peak at range bin 290 over the next highest peak is now 2.08 dB. That amounts to a small improvement of 0.92 dB.

The reasons why the EFA and JDL processors did not perform very well are due to the nature of the measured data. For this specific data set, the environment consists of rural terrain with many highways and speckles of urban clutter. This causes considerable non-homogeneities in the radar returns. The assumption made regarding the iid nature of the secondary data, needed for the covariance computation, is no longer valid. Therefore, the secondary data further away from the cell under test is non-homogeneous in comparison, thereby degrading the secondary data. This is best illustrated in the following figures.

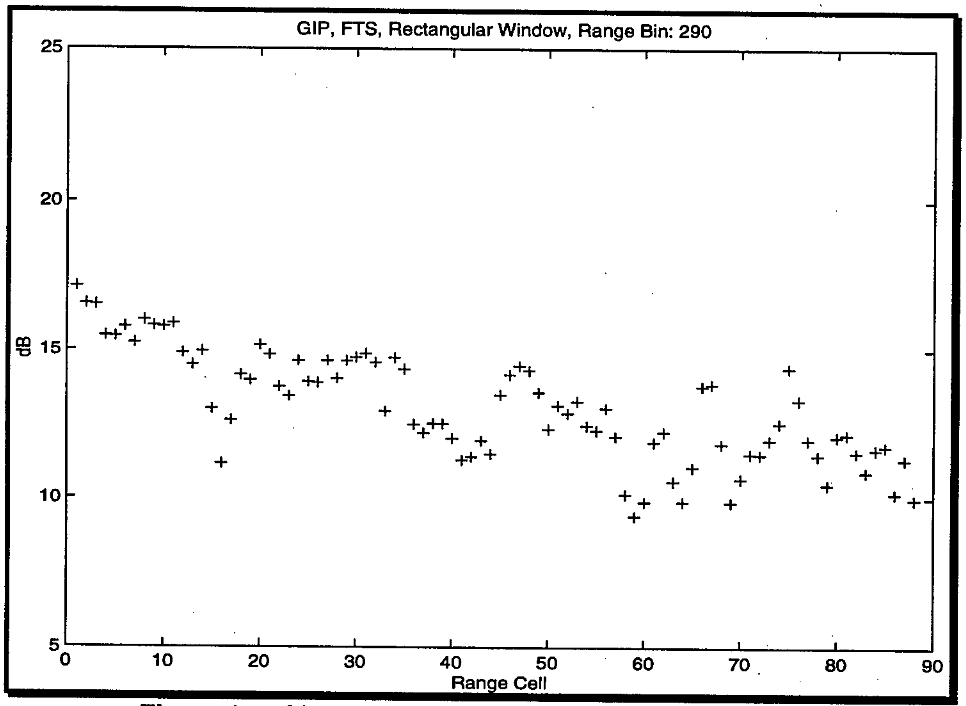

Figure 65. GIP, FTS, Symmetric about Range Bin 290.

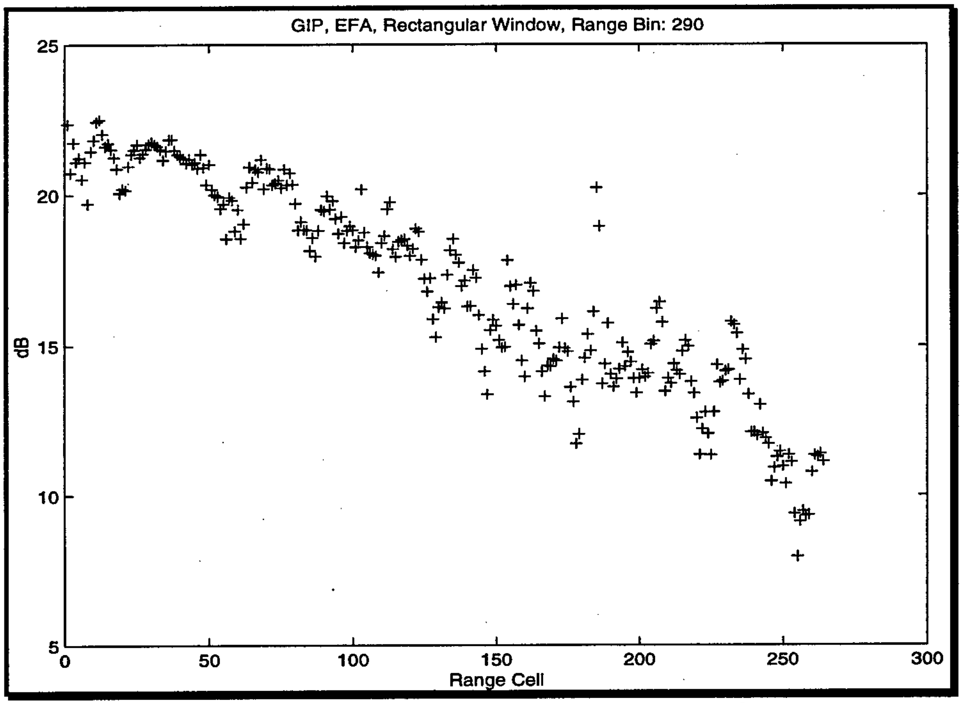

Figure 66. GIP, EFA, Symmetric about Range Bin 290.

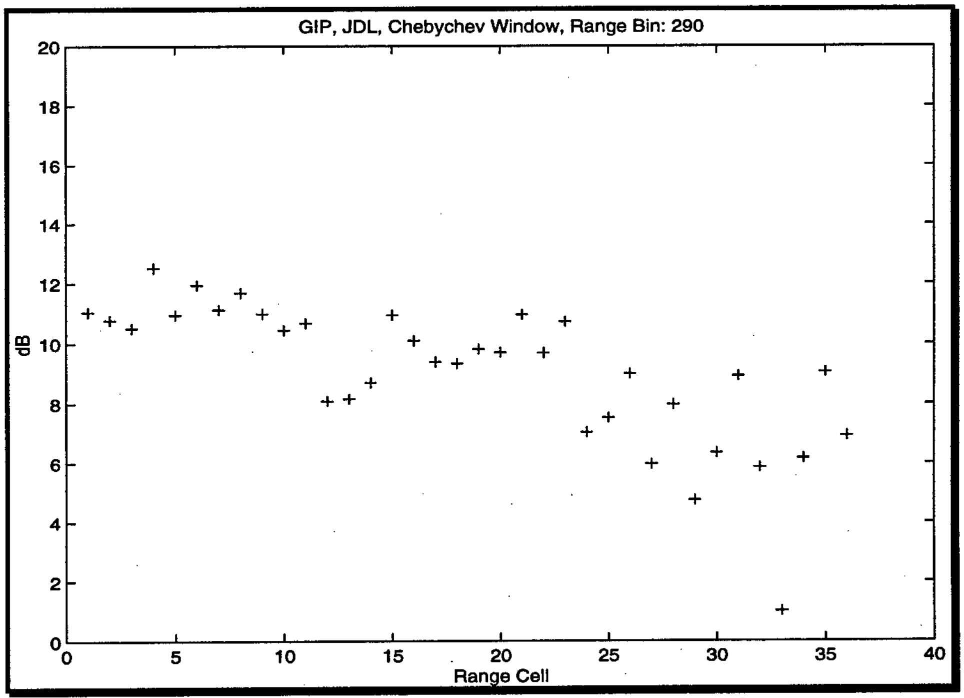

Figure 67. GIP, JDL, Symmetric about Range Bin 290.

Figures 65 through 67 show the GIP for all three techniques. Note from these figures that the GIP fluctuates considerably from its mean. For example in the case of the EFA approach, the mean value of the GIP is 18.2 dB. This makes it extremely difficult for any processor to obtain a covariance estimate representative of the cell under test.

Taking into consideration all of the above mentioned conditions, one can easily assert that the FTS process did very well in detecting the target, but use of the NHD detector not only improved the target detection but allowed for a better clutter cancellation.

5.6.5 Parametric Adaptive Matched Filter (PAMF)

For completeness, we include here some results obtained using the PAMF approach. After extensive analysis, it was found that for the PAMF approach to work well, the number of pulses N should be much higher than the number of channels J. However, we note that the number of secondary data needed K, is much less than the 2JN criteria.

Another important issue is the model order selection. It is shown [15] that the model order P is given by:

$$P = \text{round}\left\{\frac{3\sqrt{N}}{J}\right\}, \qquad (34)$$

where the function "round" rounds to the closest integer value. We give below, a table that shows the cases considered. Note that the analysis was carried out using data from file rd050575. In each case, a target was injected at range bin 290 and 0° azimuth angle. The amplitude and Doppler bin have been varied, depending on the scenario used. We will be showing the results obtained using the value of P as suggested by Equation (34) and the value of P that gave the best results. We will also show the effects of P on the test statistic.

Table 4. Cases studied with the PAMF

Case	J	N	K	P (suggested)	P (used)
1	2	32	6	8	4
2	2	64	6	12	1
3	2	128	6	17	9
4	4	64	12	6	7
5	4	128	12	8	10
6	14	64	42	1	1

Figure 68 shows the variation of the test statistic as a function of range bin, for case 1. In this case, the target's amplitude is assumed to be 0.15 and located in Doppler bin 3. This way, the target competes with very strong sidelobe clutter. The digital beamformer was first run and was unable to detect the target.

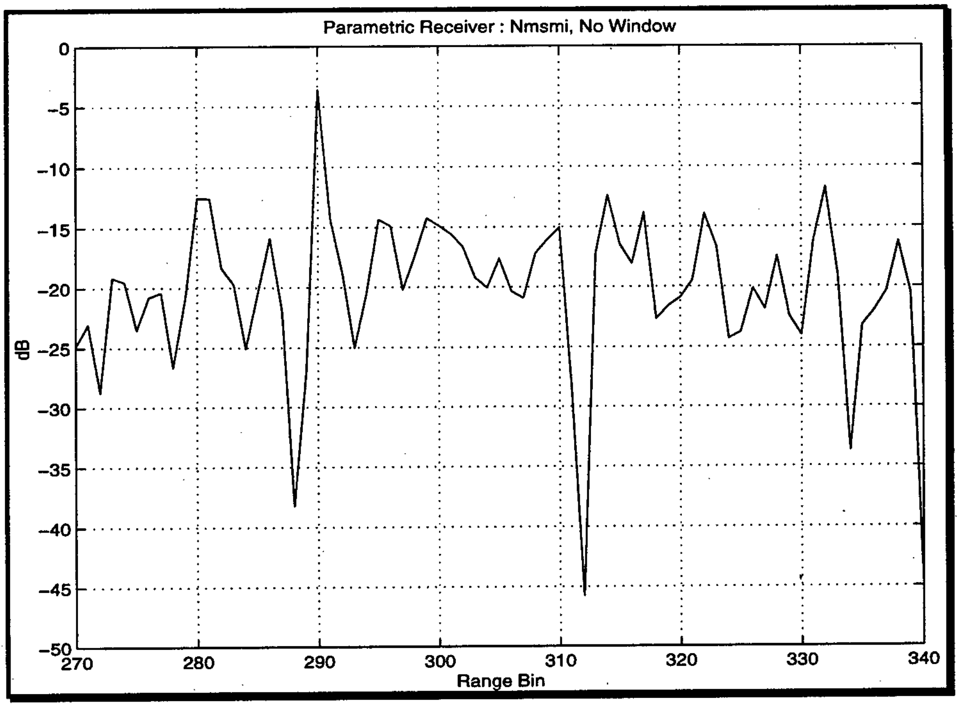

Figure 68. PAMF output, P = 4.

From the above figure, note that the PAMF did very well in detecting the target at range bin 290. Moreover the target gain is 8.14 dB. Note also that K = 6 secondary data have been used in the algorithm. If the suggested value of P (P = 8) had been used instead, the target gain is only 1.84 dB. This shows an improvement of about 6.3 dB, which is significant in our case.

Next, we plot the target gain as a function of the model order P.

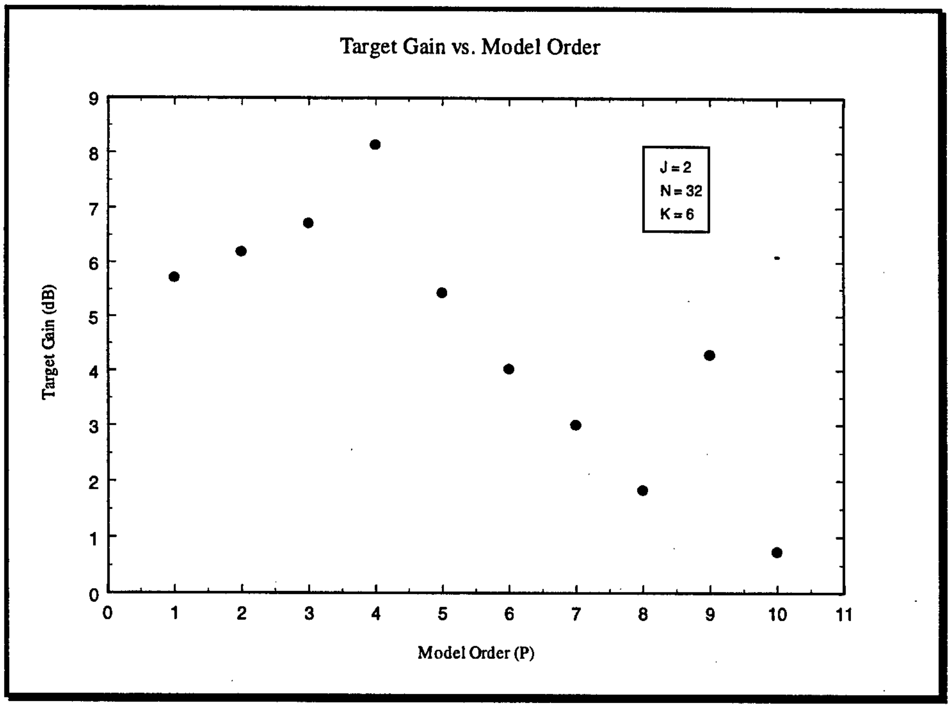

Figure 69. Target Gain vs. Model Order P.

Note from the figure that the target gain peaks at P = 4, and then starts decreasing from there. For some unknown reasons that we are still investigating, it does peak up again at P = 9.

Next we show the variations of the test statistic as a function of range bin for case 2. In this case the target's amplitude was assumed to be 0.15 and located at Doppler bin 3. The digital beamformer was unable to detect the target at range bin 290. The output of the PAMF is shown in Fig. 70.

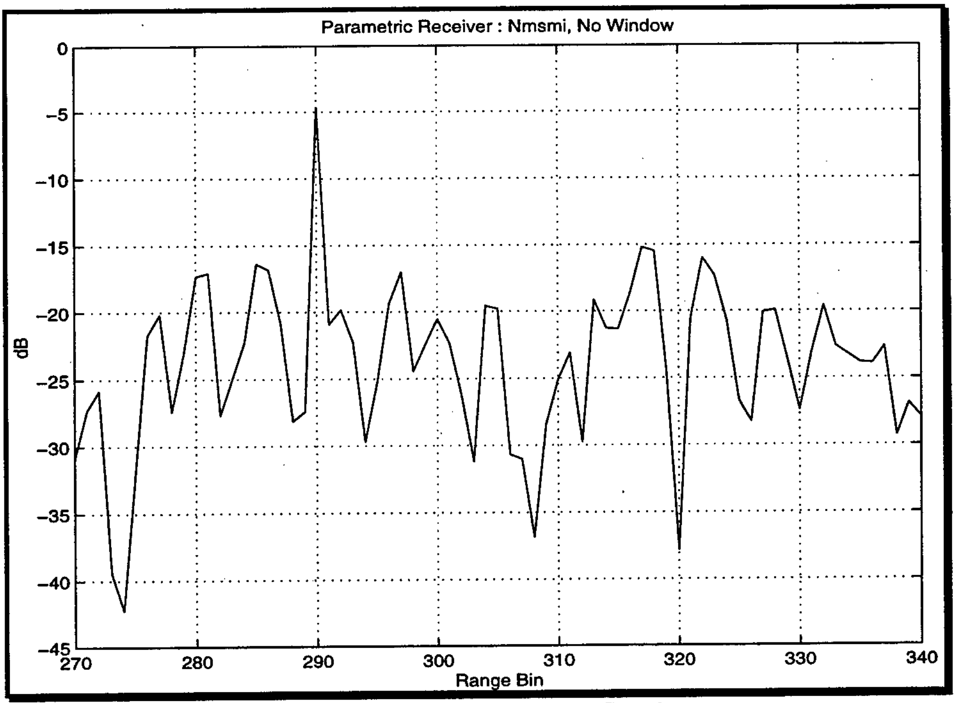

Figure 70. PAMF output, P = 1.

Note from this figure that the PAMF was able to correctly locate the target at range bin 290 with a target gain of about 10.56 dB. Moreover this was done at P = 1. The suggested value of P in this case is P = 12. However, with this value the target gain is 8.05. This amounts to a drop of about 2.51 dB.

Next, we plot the target gain as a function of the model order P.

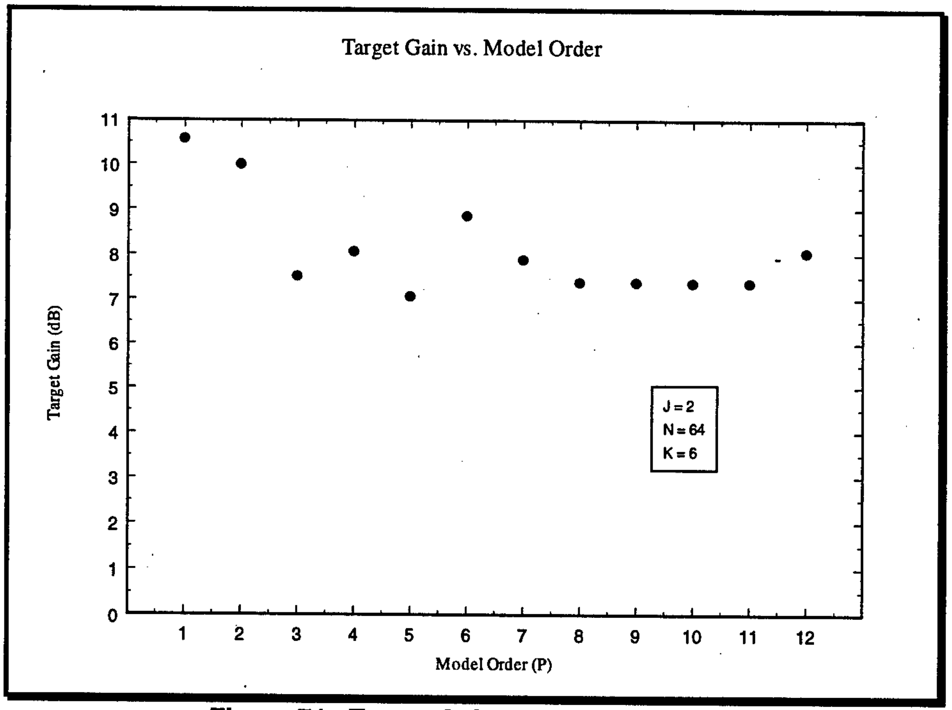

Figure 71. Target Gain vs. Model Order P.

As we mentioned earlier, the best performance was achieved at P = 1, even though the suggested value is P = 12. From Fig. 71, it is clear that the PAMF performs well with P = 1. Using a higher value of P results actually in an inferior performance of the PAMF. Moreover, with higher model order value, there is an increase in computational load. This is an undesirable feature of the PAMF receiver and is a function of the systems parameters.

Next, we show the performance of the PAMF for case 3. In this case J is still assumed to equal to 2 (J = 2), however the number of pulses is increased to 128 (N = 128). The number of secondary data is still kept to 6 (K = 6). A target was injected at Doppler bin 3 with amplitude 0.15. Figure 72 shows the PAMF output with a model order P = 9.

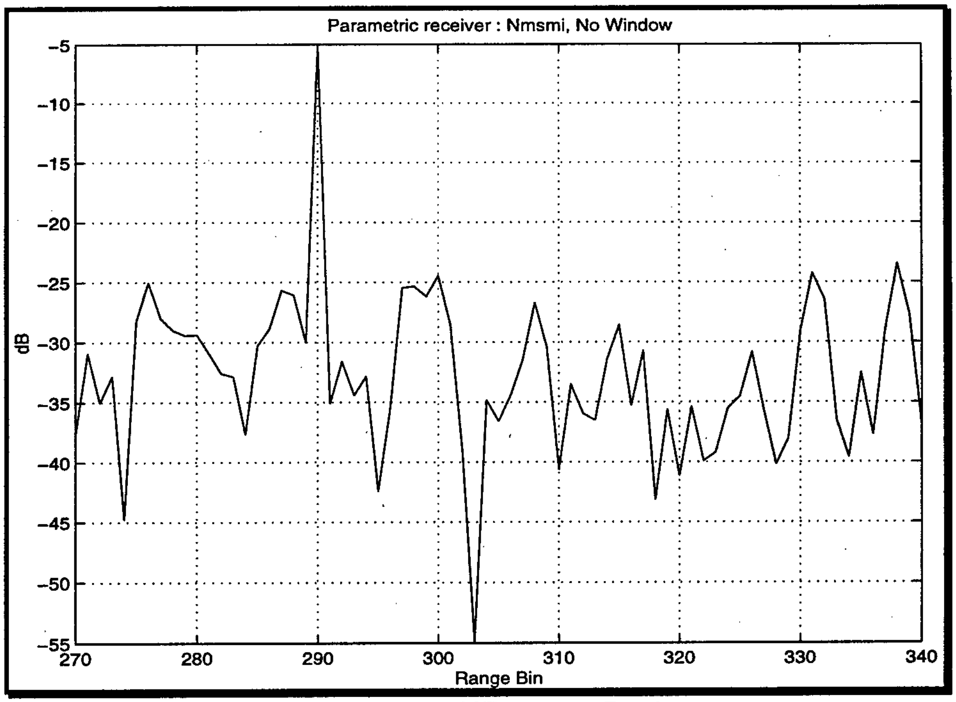

Figure 72. PAMF output, P = 9.

From Fig. 72, it can be seen that the target has been well recovered with a target gain of 18.36 dB. If the suggested value of P (P = 17) has been used instead, the gain drops to 15.65 dB, showing a loss of about 2.71 dB.

Figure 73 shows the performance of the PAMF as a function of the model order P.

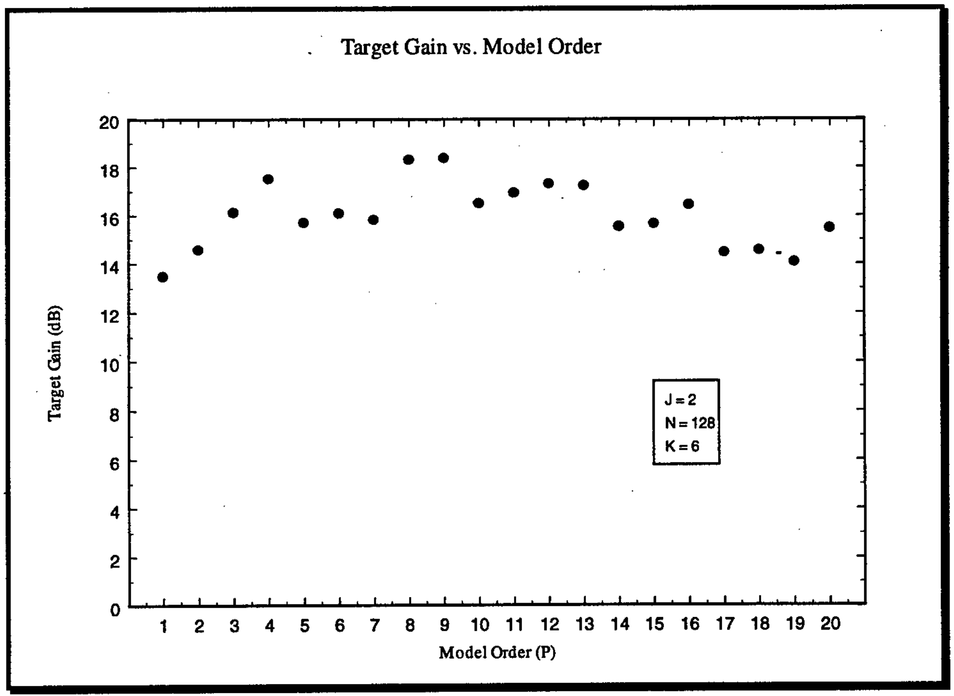

Figure 73. Target Gain vs. Model Order P.

As can be seen from Fig. 73, the PAMF performance drops with value of P greater than 9. Note however that this drop is not as accentuated as in previous cases. This is again another feature of the PAMF that needs to be more closely looked at. Note that these issues are beyond the scope of the present effort.

Figure 74 shows the performance of the PAMF algorithm for case 4. Here, the number of elements J was increased from 2 to 4 (J = 4). N = 64 pulses are considered in this case. A total of K = 12 (3J) secondary data have been used. A target was injected at Doppler bin 3 with amplitude 0.075. The digital beamformer was first run and failed to identify the target at range bin 290.

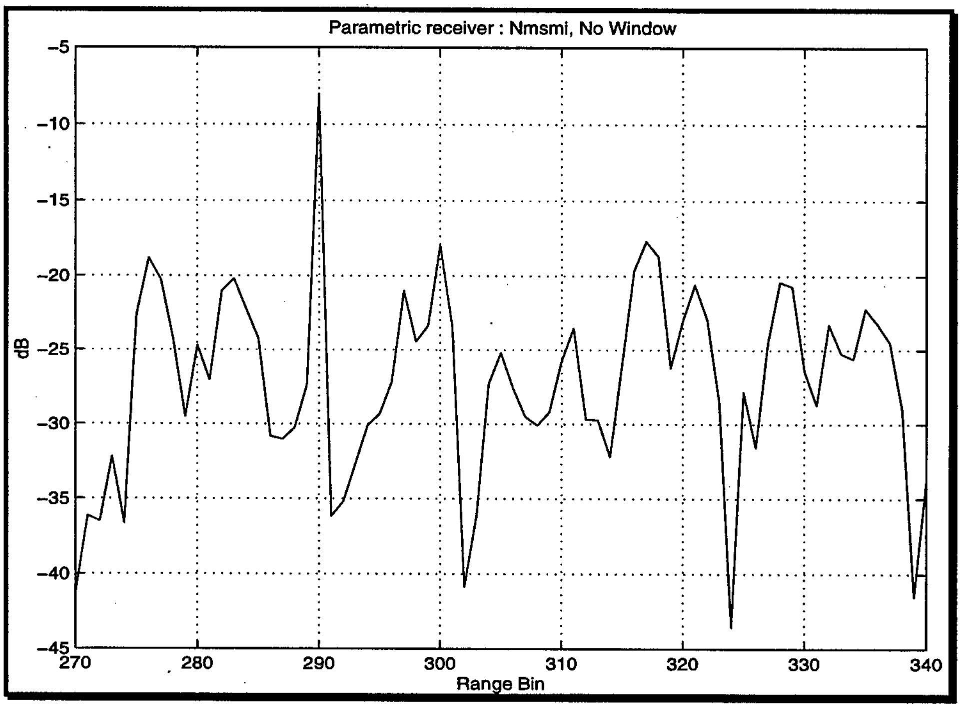

Figure 74. PAMF output, P = 7.

From Fig. 74, it can be seen that the PAMF clearly distinguishes the target at range bin 290 with a target gain of 9.69 dB. If the suggested value of P (P = 6) is used instead, this gain drops to 8.35 dB, showing a loss of 1.34 dB.

Figure 75 shows the variations of the target gain over the model order P.

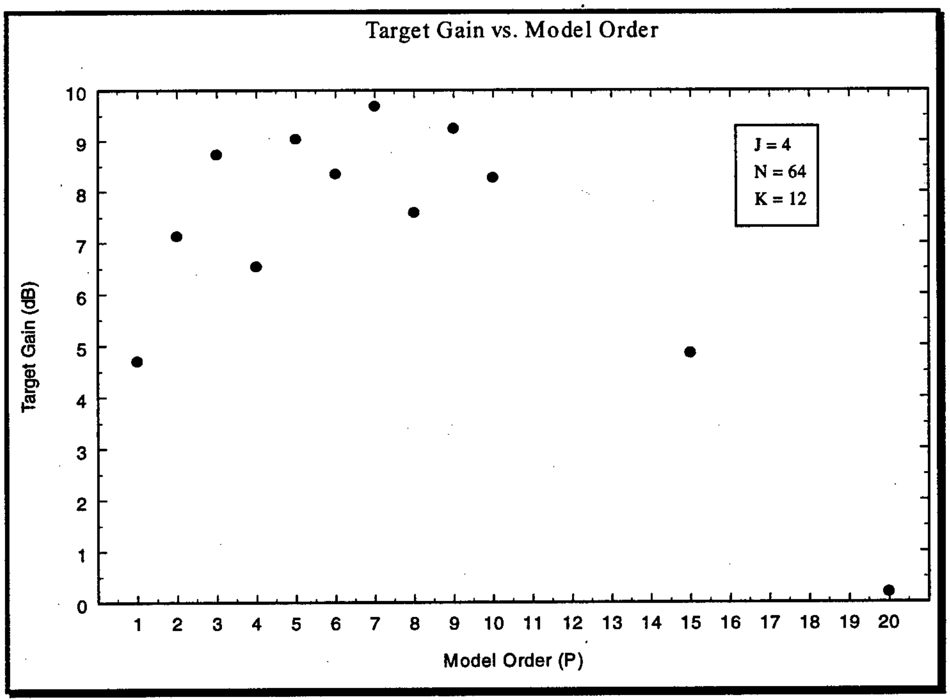

Figure 75. Target Gain vs. Model Order P.

Note from this figure that the target gain drops considerably as expected for higher value of P (beyond 10). This is consistent with the theory, which predicts that the algorithm performs better with low model order.

Next we study scenario 5. In this case J = 4, but the number of pulses was increased to 128. K is still kept to 12. The target is at Doppler 3 with amplitude 0.1.

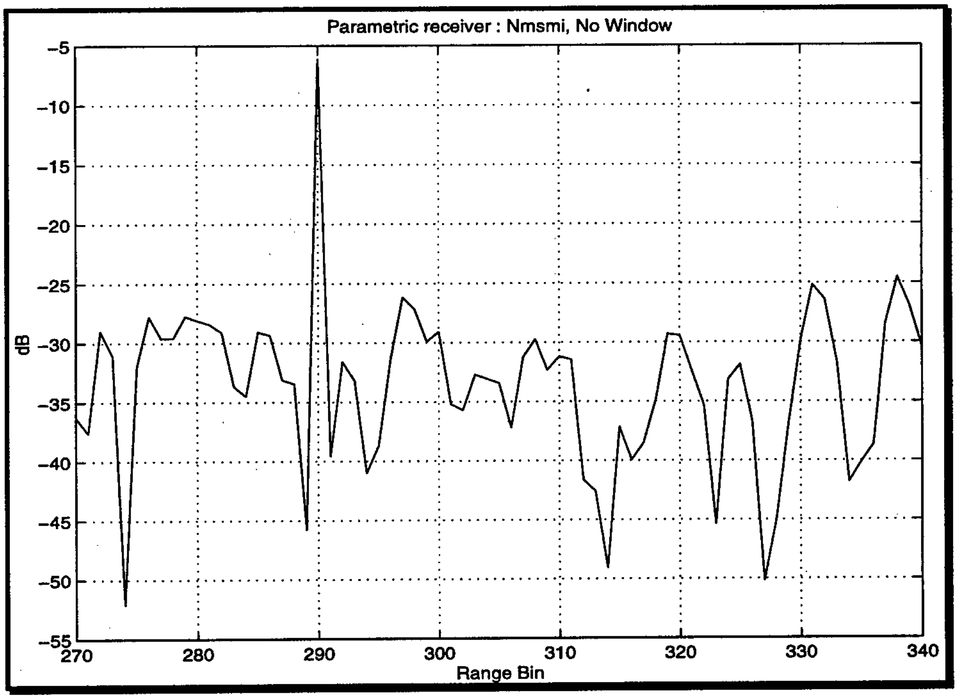

Figure 76. PAMF output, P = 10.

Again, the digital beamformer failed to identify the target at range bin 290. The PAMF output is shown in Fig. 76. From this figure, note that the PAMF uncovers the target with a target gain of 18.42 dB using a value of P equal to 10. Using the suggested value of P (P = 8), the target gain is only 16.17 dB, showing a loss of 2.25 dB.

Figure 77 shows the variations of the target gain as function of P.

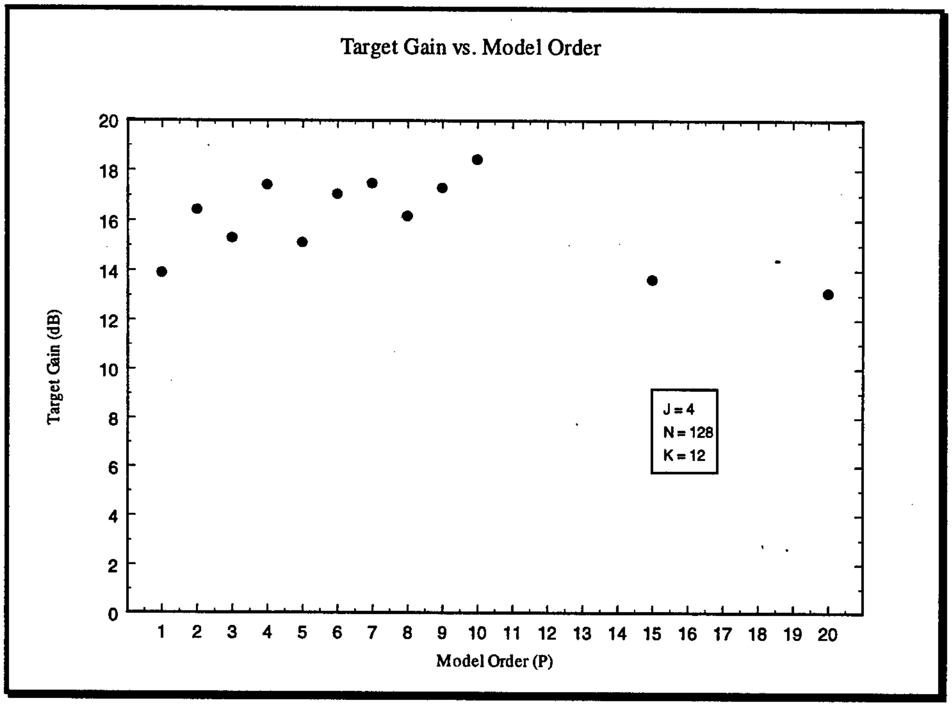

Figure 77. Target Gain vs. Model Order P.

As expected, the performance of the PAMF drops when P is greater than 10.

Finally, we study case 5. This is a more realistic case since the number of sensors used is increased to 14 (J = 14). However because of computational complexity, the number of pulses is only 64 (N = 64). The number of secondary data is assumed to be 3J (42). A target was injected again at range bin 290, azimuth angle 0, Doppler bin 55, and amplitude 0.02. Figure 78 shows the output of the PAMF when P = 1. Note that this is the only case where the suggested value of P yields the optimum performance.

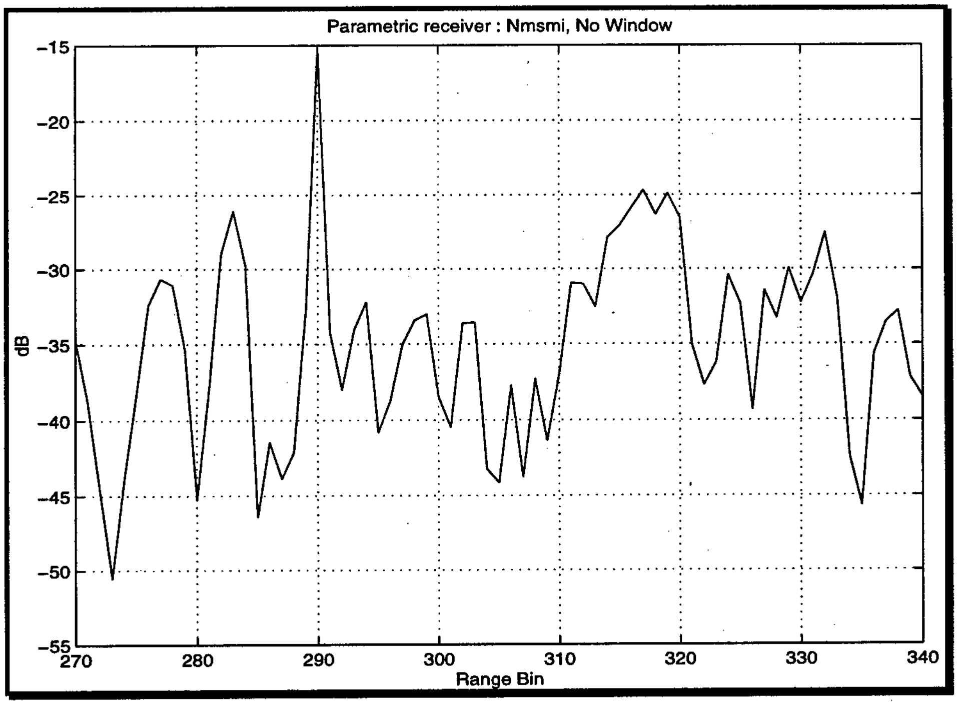

Figure 78. PAMF output, P = 1.

From Fig. 78, it is clear that the PAMF detects very well the target at range bin 290. Moreover the target gain is 9.46 dB.

Next, we show the variations of the target gain as a function of P.

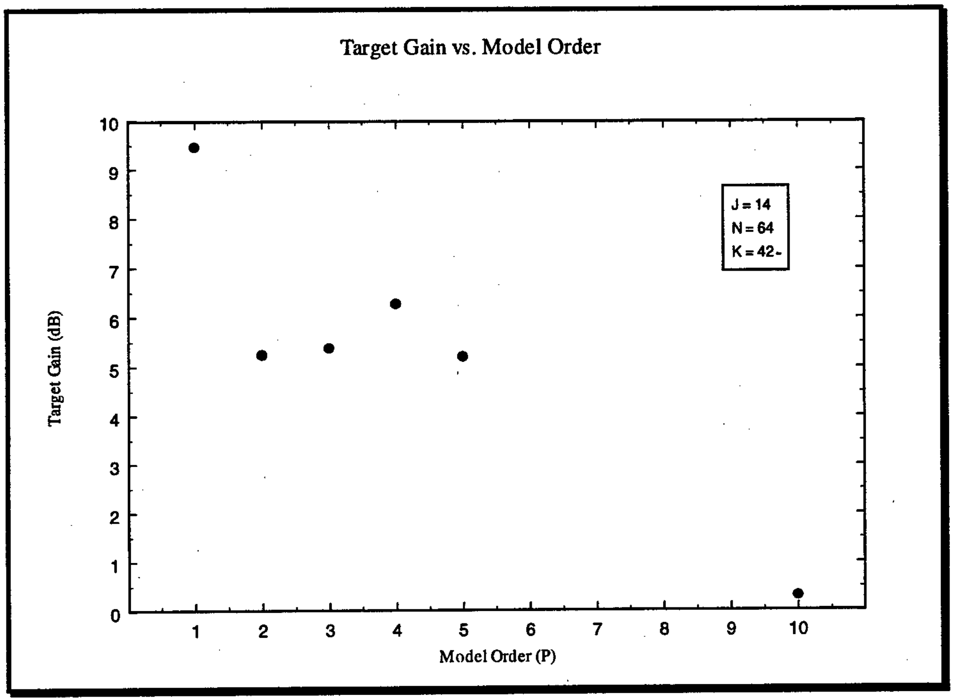

Figure 79. Target Gain vs. Model Order P.

From Fig. 79, it is clear that the PAMF performance drops significantly when the model order is over-estimated. For instance, if a value of 10 is chosen for P, the target gain is only 0.3 dB. This results in a drop of 9.16 dB. This represents a huge loss when dealing with low flying and/or low cross-section targets.

When compared to other techniques, the PAMF performs extremely well [24]. Moreover, the fact that the number of secondary data needed is very small (on the order of 3J), this makes it a possible candidate for a fielded approach. Further, note that no covariance matrix inversion is involved with this algorithm. However, it must be noted that since the data vectors involved are space-time vectors, the computational burden is increased when dealing with large values of J and N.

6. Conclusions and Recommendations

In this effort, we processed raw data obtained through the MCARM program. This data has undergone extensive pre-processing before it was made available to researchers over the Internet. To date, we have processed all the files provided to RADC. Our discussions with key AFRL personnel about the pre-processing of the data leads us to believe that it was done properly. So far, we pre-processed and made available through the Internet 569 data files. It is our belief that the MCARM database is the best available airborne data to date. It provides researchers with a wide variety of clutter data involving several regions such as land, sea, forest, urban as well as a mixture of all the above. From the BSC log files, useful information could be extracted and used in the processing of the data. As we have shown, flight trajectories could be plotted and overlaid actual terrain maps to provide researchers with better understanding of their findings.

RADC carried several STAP analyses involving both artificial targets and true targets. (We refer to the MTS tones as true targets). We have successfully canceled the underlying clutter and observed good detection results with most techniques. The following observations are made.

- Adaptive processing improves the detection performance of airborne radar significantly over conventional approaches. Recall that the digital beamformer failed to detect the target(s) in all cases.
- All proposed processors performed well with adequate sample support size. Considering the computational complexity of the algorithm as one requirement in selecting the right technique and especially in cases where the number of pulses is high, FTS is a good candidate for a fielded approach. It offered improved detection with only 2J secondary data needed for the covariance estimation.
- Covariance estimation and window selection are two primary issues to be considered in each case. The non-homogeneous nature of the clutter returns violates the iid assumption made about the secondary data. Non-homogeneity detectors have shown improved target detection as well as a good disturbance cancellation.
- Depending on how the algorithm operates on the data, applying a window may severely degrade the performance of the algorithm.

- The PAMF approach seems to be a very promising approach in STAP processing. Attractive features of this technique include the low number of sample support size involved and low model order. However, more testing is needed before it can be proposed for a fielded implementation.

7. References

1. D. Sloper et al, MCARM Final Report, Rome Lab Tech. Rept., RL-TR-96-49, April 1996.
2. A. D. Jaffer et al, "Adaptive Space-Time Processing Techniques for Airborne Radars", RL-TR-91-162, July 91.
3. L. E. Brennan and I. S. Reed, "Theory of adaptive Radar," IEEE Trans. AES, Vol. 9 No. 2, pp.237-252, Mar. 73.
4. Robey, F.C., Fuhrmann, D. R., Kelly, E.J., Nitzberg, R., "A CFAR adaptive matched filter detector," IEEE Transactions on Aerospace and Electronic Systems, pp. 208-216, Jan. 1992.
5. R.C DiPietro, "Extended Factored Space-Time Processing for Airborne Radar," Proc. 26-th Asilomar Conf., Pacific Grove, CA, pp. 425-430, Oct. 92.
6. H. Wang et al., "On Adaptive Spatial-Temporal Processing for Airborne Surveillance Radar Systems," IEEE Trans. AES, Vol. 30, No. 3, pp. 660-670, July 94.
7. R. S. Blum et al., "An Analysis of Adaptive DPCA", Proc. IEEE Natl. Radar Conf., Ann Arbor, MI, pp. 303-308, May 96.
8. I. P. Kirsteins and D. W. Tufts, "Adaptive Detection Using Low Rank Approximation to a Data Matrix", IEEE Trans. AES, Vol. 30, No. 1, pp. 55-67, Jan. 94.
9. B. D. Van Veen, "Eigenstructure Based Partially Adaptive Array Design", IEEE Trans. AP, Vol. 36, No. 3 Mar. 88.
10. A. M. Haimovich, "The Eigencanceler: Adaptive Radar by Eigen-analysis Methods", IEEE Trans. AES, Vol. 32, No. 2, pp. 660-670, April 96.
11. J. S. Goldstein and I. S. Reed, "Subspace Selection for Partially Adaptive Sensor Array Processing," IEEE Trans. AES, Vol. 33, No. 2, pp. 539-544, April 97.
12. Rangaswamy, M., Michels, J.H., "A Parametric Multi-Channel Detection Algorithm for Correlated Non-Gaussian Random Processes," IEEE National Radar Conference, Syracuse, NY, May 1997.
13. I.S. Reed, J. D. Mallett and L.E. Brennan, "Rapid convergence in adaptive arrays," IEEE Trans. AES, Vol. 10, no. 6, pp. 853-863, November 1974.
14. Haykin, S., Adaptive Filter Theory, 3rd Edition, Prentice Hall, 1996.
15. Nuttall, A., "Multivariate Linear Predictive Spectral Analysis Employing Weighted Forward and Backward Averaging," Tech. Rep. TR-5501, Naval Underwater Systems Center, New London, CT, October 1976.

16. Strand, O.N., “Multi-Channel Complex Maximum Entropy (Auto-Regressive) Spectral Analysis,” IEEE Trans. Automatic Control, vol. AC-22, pp. 634-640, Aug. 1977.
17. E. Conto et al., "Adaptive Matched Filter Detection in Spherically Invariant Noise", IEEE Sig. Proc. Let., Vol. 3, No. 8, pp. 248-250, Aug. 96.
18. W. L. Melvin et al., "Assessment of Multi-Channel Airborne Radar Measurements for Analysis and Design of Space-Time Processing Architectures and Algorithms", Proc. IEEE Natl. Radar Conf., Ann Arbor, MI, pp. 130-135, May 96.
19. W.L. Melvin and M.C. Wicks, “Improving practical space-time adaptive radar," *Proc. IEEE Natl. Radar Conf.*, pp. 48-53, May 1997.
20. W. L. Melvin and B. Himed, “Comparative Analysis of Space-Time Adaptive Algorithms with Measured Airborne Data,” presented at the 7th International Conference on Signal Processing Applications & Technology, Oct. 7-10, 1996.
21. B. Himed and W. L. Melvin, “Analysis of Reduced-Dimension Space-Time Adaptive Processors using Monostatic MCARM Data”, Presented at the 1997 ISDSP Conf., London, July 1997.
22. B. Himed and W. L. Melvin, “Analyzing Adaptive Space-Time Processors Using Measured Data”, Presented at the 31-st ASILOMAR Conf. on Signals, Systems and Computers, Monterey, CA, Nov. 1997.
23. J. H. Michels, T. Tsao, B. Himed and M. Rangaswamy, “Space-Time Adaptive Processing (STAP) in Airborne Radar Applications”, To be presented at the IASTED Int. Conf. On Signal Processing and Communications, Canary Islands, Spain, Feb. 11-14 1998.
24. T. Tsao, B. Himed and J. H. Michels, “Effects of Interference rank estimation on the detection performance of rank reduced STAP algorithms” to be presented at the IEEE 1998 National Radar Conf., Dallas, TX, May 12-13 1998.
25. P. Sanyal, W. L. Melvin and M. C. Wicks, “Space-Time Adaptive Processing for Advanced Airborne Surveillance (AAS) Bistatic Radar,” 43-rd Tri-Service Radar Symposium, Boulder, CO, June 1997.
26. P. Sanyal, “Application of Space-Time Adaptive Processing to MCARM Bistatic Data”, MITRE Technical Report, MTR 97B0000068, September 1997.

Appendix A

MCARM Header Information

➢ AcquisitionNumber:	This integer should agree with the acquisition number portion of the filename.
➢ ApplyRxMatchOn:	N/A.
➢ AttenChannels:	These floating point values are the receiver attenuation settings for each of the MCARM channels for the beam position and
➢ BeamSteeringType:	This integer in conjunction with Rcvr Mode determines the mode of the radar. See table in MCARM Data Reduction document.
➢ BlockSize:	N/A.
➢ CPI1:	Baseband, Digital I/Q, Pulse Compressed, Channel balanced samples of multi-channel coherent processing interval (CPI) radar data. This data is in the form of an S row by M column matrix of integer values, where S is the number of samples per CPI and M is the number of receiver channels. For MCARM data, S equals RangeCellsPerIPP * IppsPerCpi For MCARM data, M is always 24. Channels (columns) 1 and 9 are the outputs of the sum and delta manifolds respectively. The other 22 channels are array sub-aperture outputs that depend on which array configuration was used. See Array Configuration section. The sample values integers representing 12 bit, signed A/D converter samples with two additional bits added as the least significant bits. Bit 0 is the range zero bit that is 1 only during the first A/D sample of each IPP. Bit 1 is the A/D saturation bit that is 1 only if the input is outside of the A/D input range. For more information see the MCARM Data Reduction document.
➢ CfarThreshold1:	N/A.
➢ CfarThreshold2:	N/A.
➢ CfarThreshold3:	N/A.
➢ ChannelMatchCoeffs:	N/A.
➢ ChannelSumOn:	N/A.
➢ ChannelPerCPI:	For MCARM data is value is always 24
➢ CheckROs:	N/A.
➢ ClutterFrequency:	For MCARM data is value is normally 0, however, it represents a frequency shift in hertz applied to the transmit pulse. The purpose of this allow the radar to move the Doppler frequency f the main-lobe clutter to zero.

- ClutterTestOn: N/A.
- ComputeRxMatchOn: N/A.
- CpisPerAquisition: This integer is the number of CPI data acquisitions included in the file. Currently only one CPI is recorded per file and is stored in variable CPI1. In future enhancement will allow multiple CPIs, with different PRFs, to be recorded. In that case variables CPI1, CPI2, etc. will be stored in the file and other variables that could have different values for the different CPIs will be vectors.
- DiqDecimation: This integer indicates amount of over-sampling that exists in the data relative to a 0.8 μsec range cell. Typically this value will be four indicating that data was sampled at 0.2 μsec intervals can be decimated four to one to obtain one sample per range cell.
- DiqOn: A value of zero indicates that the data is real A/D sample data. A value of one indicates that I/Q processing was performed prior to writing this file. See the Digital IQ and Pulse Compression sections.
- DisplavChannelSelec: N/A.
- DisplayIppOrRgSelec: N/A.
- DisplaySelection: N/A.
- DolphdB: N/A.
- EnvelopeDetectSelec: N/A.
- FlightNumber: This variable indicates the flight number.
- FTAAHARSCompassRead: N/A.
- FTAAHARSPitch: N/A.
- FTAAHARSRoll: N/A.
- FTAAHARSValidity: N/A.
- FTABAPressureAlt: Target plane barometric pressure altitude in feet.
- FTADataValid: Indicates the validity of target plane navigational data. A value of one indicates that the data is valid. A value of zero indicates that the data is invalid or that navigation data is not being received.
- FTAGPSGroundSpeed: Target plane ground speed in knots calculated from GPS data.
- FTAGPSGroundTrack: Target plane ground speed true North heading in 0. I degrees.
- FTAGPSLatitude: Target plane latitude or Rome Laboratory Multiple Target Simulator (MTS) latitude in degrees (see Note 2).

➤ FTAGPSLongitude:	Target plane longitude or Rome Laboratory Multiple Target Simulator (MTS) longitude in degrees. (See Note 2.)
➤ FTAGPSTime:	Target plane current time from GPS data in milliseconds from midnight GMT.
➤ FTAGPSAltitude:	Target plane altitude from GPS data in feet above mean sea level.
➤ FTARAHeight	Radar At -- zero
➤ FftOn:	For MCARM data, this value is normally zero. If nonzero, it indicates that the radar performed Doppler processing of the data.
➤ FftWeightSelect:	????
➤ FilterIncrement:	N/A.
➤ FilterRangeGate:	N/A.
➤ FilterStart:	N/A.
➤ FilterStop:	N/A.
➤ GPSAltitude:	MCARM platform altitude from GPS data in feet above mean sea level.
➤ GPSLatitude:	MCARM platform latitude from GPS data in degrees.
➤ GPSLongitude:	MCARM platform longitude from GPS data in degrees.
➤ GPSNavDataValid:	Binary coded status from GPS data. A value of 8192 (2^13) indicates valid data. Other values indicate GPS errors.
➤ GPSSystemTime:	MCARM platform current time from GPS data in seconds from midnight GMT.
➤ GPSTimeOfXmit:	Value is normally zero. If used, it is an (16-bit) integer representing the time that the GPS data was transmitted measured in 64 microsecond periods since system initialization.
➤ GPSTimeValid:	A value of zero indicates that the GPSSystemTime value is valid.
➤ GPSVelocityEast:	The east component of MCARM platform from GPS data in knots.
➤ GPSVelocityNorth:	The north component MCARM platform velocity from GPS data in knots.
➤ GPSVelocityVert:	The vertical component MCARM platform velocity from GPS data in knots where positive indicates upward direction.
➤ GatesToDump:	N/A.
➤ GatesToRecord:	N/A.

➤ GuardThresholdl;	N/A.
➤ GuardThreshold2:	N/A.
➤ HoleSize:	N/A.
➤ INUAltitude:	MCARM platform altitude in feet from INU data. The altitude is calculated from inertial navigation data and subject to long term error. (See Note 1.)
➤ INUAltitudeDelta:	This integer is the time when attitude data was sampled by the INU relative to the time when velocity data was sampled by the INU measured in 14.75 microsecond periods. The time when the velocity data was sampled is recorded in INUVelocityTime. (See Note 1.)
➤ INUDataValid:	A value of zero indicates that INU data is valid. A nonzero value of indicates invalid INU data. (See Note 1.)
➤ INUDriftAngle:	The angle between the MCARM platform center line and the ground velocity direction in radians from INU data. (See Note 1.)
➤ INUGroundSpeed:	MCARM platform ground speed in feet per second from INU data. (See Note 1.)
➤ INULatitude:	MCARM platform latitude in radians from INU data. Positive angles are north of the equator. (See Note 1.)
➤ INULongitude:	MCARM platform longitude in degrees from INU data. Negative angles are west of the prime meridian. (See Note 1.)
➤ INUPitch:	MCARM platform pitch in radians from INU data. Positive angles are nose up. (See Note 1.)
➤ INUPlatformAzimuth:	This value is the angle between the navigation system's inertial x-axis and the MCARM platform center line in radians from INU data. (See Notes 1 and 3.)
➤ INUPlatformRoll:	MCARM platform roll angle in radians from INU data. Positive angles are right wing down. (See Note 1.)
➤ INUVelocityTime:	This integer is the number of 64 µsec periods since initialization of MCARM platform INU system. (See Note 1.)
➤ INUVelocityX:	MCARM platform velocity along the navigation system's inertial x-axis in feet per second from INU data. (See Note 1.)
➤ INUVelocityY:	MCARM platform velocity along the navigation system's inertial y-axis in feet per second from INU data. (See Note 1.)
➤ INUVelocityZ:	MCARM platform velocity along the navigation system's inertial z-axis in feet per second from IW data. The z-axis is vertical relative to the center of the earth with positive values up. (See Note 1.)

➤ INUWanderAngle:	This value is the angle between the navigation system's inertial x-axis and true north in radians from INU data. (See Notes 1 and 3.)
➤ IppsPerCpi:	This integer value is the number of pulses (inter-pulse periods) that occurred during this acquisition. The number of pulses recorded is IppsPerCpi - NumSpaceChargePulse. (See discussion of CPI1.)
➤ ManifoldSwitch:	A value of zero indicates that the receivers are connected to the test manifold during the transmit pulse.
➤ McarmMesaMode:	This integer value will be zero for all MCARM data.
➤ MesaOptions:	N/A.
➤ MesaProcessing:	N/A.
➤ ModeName:	This is a MATLAB string variable containing the name assigned to the mode of the MCARM radar when this data was recorded.
➤ ModuleMapChannel:	This variable is a 1 by 24 vector of binary encoded integers. Each element of the vector corresponds to a MCARM receiver channel. Element ModuleMapChannel(i) indicates which of the 32 T/R modules in the MCARM array are connected to receiver channel i. Each element of ModuleMapChannel should be treated as a 32 bit 2's complement integer where each bit corresponds a T/R module. The MSB corresponds to module 31 and the LSB corresponds to module 0. A bit value of one indicates that the corresponding module is connected and zero indicates that it is not connected. Typically, elements 1 and 9 have the value -1, i.e., all bits are 1. This is because channels 1 and 9 are the sum and difference manifolds and are connected to all modules.
➤ NoiseLevelTestOn:	N/A.
➤ NumRangeGateZero:	This integer represents the number of range cells (0.8 μsec) during an inter-pulse period (IPP) during which the receivers are not being recorded. When this value is zero, the receivers are being recorded during the entire inter-pulse period, including during the transmit pulse.
➤ NumSpaceChargePulse:	This integer value represents the number of inter-pulse periods at the beginning of a coherent processing interval (CPI) that were not recorded. The intent of this feature is to delay recording so that all range ambiguous echo pulses are being recorded. The number of pulses recorded is: IppsPerCpi - NumSpaceChargePulse. (See discussion of CPI 1.)
➤ Passthru:	N/A.

- PcWeightSelect: ???
- PlaybackControl: N/A.
- PrfCombineOn: N/A.
- Prf: Pulse repetition frequency.
- PulseCompressSelect: A value of one indicates that I/Q processing was performed prior to writing this file. See the Digital IQ and Pulse Compression sections.
- PulseToPulseSelect: N/A.
- PulseWidth: This value is the width of the transmit pulse which corresponds to 0.8 μsec range cells.
- QuickLookOn: N/A.
- RangeCellsPerIPP: This integer is the number of range cells recorded during this CPI. (See discussion of CPI 1.)
- RangeGateFilter: N/A.
- RangeGateIncrement: N/A.
- RangeGateSelectOn: N/A.
- RangeGateStart: N/A.
- RangeGateStop: N/A.
- Rawdata: Indicates which data file is being processed
- RcvArrayWeight: N/A.
- RcvAzBWShape: This integer indicates which array tuning file was used for receive beam shape and steering.
- RcvAzPointAngle: N/A.
- RcvAzPointIndex: This integer is the beam position index selected for the receive mode.
- RcvElPointAngle: N/A. (Elevation steering is determined by azimuth steering.)
- RcvElPointIndex: N/A. (Elevation steering is determined by azimuth steering.)
- RcvError: N/A.
- RcvFrequency: The receiver center frequency used during receive mode in MHz.
- RcvrMode: This integer in conjunction with BeamSteeringType determines the mode of the radar. See table in MCARM Data Reduction document.
- RecordOn: N/A.
- Rxtest: Indicates which test file was used for channel balancing

➤ RxExpectedWeight:	N/A.
➤ RxSensitivityTstOn:	N/A.
➤ RxStabTestOn:	N/A.
➤ StcOn:	N/A.
➤ TxExpectedWeight:	N/A.
➤ TxPowerTestOn:	N/A.
➤ TxStabTestOn:	N/A.
➤ WESTINGHOUSE:	This is a MATLAB string variable containing a post-flight entered comment describing the data acquisition in the file.
➤ WaveForm:	This integer indicates the transmit waverforn type. A value of 0 indicates continuous wave transmission, 1 indicates 14.4 µsec LFM transmit pulse, 2 indicates 50.4 µsec LFM transmit pulse, 3 indicates 100 µsec LFM transmit pulse.
➤ XmitArrayWeight:	N/A.
➤ XmitAzBWShape:	This integer indicates which array tuning file was used for transmit beam shape and steering.
➤ XmitAzPointAngle:	N/A.
➤ XmitAzPointindex:	This integer is the beam position index selected for the transmit mode.
➤ XmitElPointAngle:	N/A. (Elevation steering is determined by azimuth steering.)
➤ XmitElPointlndex:	N/A. (Elevation steering is determined by azimuth steering.)
➤ XmitFrequency:	The transmitter center frequencies used during transmit mode in MHz.

<u>Notes</u>:

Inertial Navigation Unit (INU) data is sampled before and after each acquisition. For this reason, each variable is a 1 by 2 vector. Time delay between samples can be calculated from INUVelocityTime as follows:

$$(\text{INUVelocityTime}\ (1,2) - \text{INUVelocityTime}\ (l, 1)) * 64\ \mu\text{sec}$$

When the target plane navigational data is not being recorded and the Rome Laboratory Multiple Target Simulator (MTS) is being received, the MCARM radar inserts the latitude and longitude coordinates of the MTS in place of the target plane latitude and longitude.

Geographic boresight angle of the MCARM radar, i.e., relative to true north, is calculated as follows:

$$(\text{INUPlatformAzimuth} - \text{INUWanderAngle}) * 180/\pi - 90 \qquad \text{(degrees)}$$

where east is +90 degrees.

☆U.S. GOVERNMENT PRINTING OFFICE: 1999-610-130-81127