

Copyright

by

Yong Zhou

2005

The Dissertation Committee for Yong Zhou
certifies that this is the approved version of the following dissertation:

**High Frequency Electromagnetic Scattering Prediction
and Scattering Feature Extraction**

Committee:

Hao Ling, Supervisor

Sean M. Buckley

Mircea D. Driga

Ross Baldick

Edward J. Powers

**High Frequency Electromagnetic Scattering Prediction
and Scattering Feature Extraction**

by

Yong Zhou, BS, MSEE

Dissertation

Presented to the Faculty of the Graduate School of
the University of Texas at Austin
in Partial Fulfillment
of the Requirements
for the Degree of
Doctor of Philosophy

The University of Texas at Austin

May 2005

To my dear daughter Julia and my wife Hongxing

ACKNOWLEDGEMENTS

I would like to thank Professor Hao Ling, my supervisor, for his long time guidance in my graduate study and research. It is those frequent enlightening academic discussions with him that directly lead to this dissertation. His profound understanding in electromagnetics has deeply extended my ken and will continuously affect me for many years.

I would also like to extend my appreciation to the distinguished members of my dissertation committee: Professor Sean Buckley in Aerospace Engineering Department, Professors Mercea Driga, Ross Baldick, and Edward Powers in the Department of Electrical and Computer Engineering, for their generous giving of their time to serve on my dissertation committee and for their welcomed suggestions. I also want to appreciate Professor Tiejun Liu, who introduced me to electromagnetic scattering analysis and advised me for my master thesis.

I would like to thank Ms. Darla Rowan and Ms. Pam Campbell for their gracious help during my graduate studies.

My wife and my family deserve my special thanks for their love and constant care, which supports me through all of my graduate studies.

Yong Zhou

The University of Texas at Austin

December, 2004

High Frequency Electromagnetic Scattering Prediction and Scattering Feature Extraction

Publication No. _____

Yong Zhou, Ph.D.

The University of Texas at Austin, 2005

Supervisor: Hao Ling

Three related electromagnetic scattering problems, namely, high frequency electromagnetic (EM) ray tracing, scattering feature extraction, and inverse scattering are studied in this dissertation. New approaches are presented to advance the state of the art in each of the areas. The presented study in electromagnetic ray tracing leads to an alternative ray tracing algorithm which can outperform the traditional algorithms for complex targets. The performance of the proposed techniques demonstrates their potential application to the study of high-frequency EM scattering prediction. Second, a genetic algorithm (GA)-based algorithm with an adaptive-feeding technique is developed to simultaneously extract both scattering centers and resonances. Scattering feature extraction algorithms are then developed with the consideration of the visibility of scattering centers. Inverse scattering problems with strong multiple scattering effects are

also studied. A GA-based method is presented to invert the shapes with multiple scattering effects. An approach combining hybrid GA with the tabu list idea are then developed to further improve the performance of the GA-based inversion algorithms.

TABLE OF CONTENTS

Acknowledgement	v
Abstract	vi
Table of contents	viii

CHAPTER ONE INTRODUCTION

1.1 Background review	1
1.2 Objectives of the dissertation	8
1.3 Organization of the dissertation	9

CHAPTER TWO RAY TRACING FOR HIGH-FREQUENCY ELECTROMAGNETIC SCATTERING COMPUTATIONS

2.1 Introduction	10
2.2 MPZ algorithm and its implementation	14
2.3 Evaluation of the MPZ against the BSP	20
2.4 Multi-aspect MPZ (MAMPZ)	26
2.5 MPZ ray tracer for IGES parametric surface	32
2.6 Summary	37

CHAPTER THREE FEATURE EXTRACTION FROM ELECTROMAGNETIC SCATTERING DATA

3.1	Introduction	39
3.2	Sparse parameterization of EM scattering data	42
3.3	Feature extraction with visibility	50
3.4	Summary	80

CHAPTER FOUR RADAR IMAGING AND HIGH FREQUENCY ELECTROMAGNETIC INVERSE SCATTERING

4.1	Radar imaging for multiple scattering effect	82
4.2	Electromagnetic inversion for metallic objects	88
4.3	Speedup of shape inversion by HGA-Tabu	104
4.4	Summary	114

CHAPTER FIVE CONCLUSIONS

BIBLIOGRAPGY

VITA

CHAPTER ONE

INTRODUCTION

1.1 Background Review

1.1.1 Current Frontier in High-Frequency Electromagnetic Scattering Prediction

Computational electromagnetics (CEM) is the solution of boundary value problems governed by Maxwell's equations on a computer. The rapid progress in computational power in the past two decades has made possible the computer simulation of electromagnetic propagation and scattering phenomena in large, complex structures. In addition, advances in computational algorithms have also taken place in CEM. A number of methods have been developed based on different mathematical formulations of Maxwell's equations.

There exist two main classes of methods, numerical methods and high-frequency methods. Numerical methods, such as the method of moments (MoM) [1], the finite element method (FEM) [2,3] and the finite-difference time-domain method (FDTD) [4], are based on the rigorous, numerical formulation of Maxwell equations. Among these methods, the integral-equation based MoM is the most widely utilized in CEM because of its high accuracy and ease of implementation. The MoM divides the structural domain into small elements and uses a set of basis functions to represent the induced current distribution on each element. The problem is then converted into a matrix equation and

the solution is obtained by a linear equation solver. For MoM, an approximate requirement of at least 10 discretized elements per wavelength has to be satisfied in order to achieve a stable solution. Generally, the computational complexity of the MoM for three-dimensional (3-D) problem is $O(M^3)$, where M is the total number of basis functions. For electrically large targets or applications requiring broadband frequency responses such as in radar cross section (RCS) prediction, the MoM can easily become very time-consuming. In recent years, a number of methods including the fast Fourier transform (FFT) [5], the fast multipole method (FMM) [6] and the wavelet sparsification algorithm [7] have been proposed to speed up the MoM computation and enhance the capability of the MoM to solve electrically large problems. For example, the FMM can reduce the computational complexity for solving the discretized wave equation from $O(M^3)$ to $O(M \log M)$. However, most of these numerical approaches still become computationally quite expensive as the frequency of interest increases. Even with today's computational resources, it remains a difficult task to predict the scattering from a full-size ground vehicle for frequencies approaching X-band (10 GHz).

A second class of methods is termed high-frequency methods. They are based on the asymptotic expansion of the solution to Maxwell equations. As the name implies, these methods are applicable at high frequencies when the wavelength is very small compared to the size of the target. In fact, due to the nature of the asymptotic expansion, their accuracy is better at high frequencies, and gradually degrades as the scatterer size becomes smaller in terms of the wavelength.

The most basic of the high-frequency techniques is physical optics (PO) [8], which approximates the induced currents on the illuminated surface of a scatterer as proportional to the incident field. The PO approximation ignores the multiple interactions that can and often exist between different parts of a scatterer. In the early 1990s, the Z-buffer technique from the computer graphics community was exploited to enhance the performance of PO computations [9]. In particular, the hardware Z-buffer technique was adopted to quickly detect lit surfaces of the target and obtain the z-depth values of the lit surfaces. Several such computer codes such as GRECO [10] and Visage [11] had been developed. Although they can achieve real-time prediction for complex targets, this technique is limited in accuracy since it is based on PO and does not take into account multiple scattering effects.

Other high-frequency methods include the geometrical theory of diffraction (GTD) [12] and the physical theory of diffraction (PTD) [13], which are ray-based methods. In addition to the physical optics contribution, they also account for higher-order scattering effects such as multiple scattering and edge diffractions. A particular implementation of the ray-based algorithms is the shooting and bouncing ray (SBR) technique [14, 15]. It has gained wide popularity because of its applicability to complex computer models of realistic targets. In this technique, millions of rays are shot at a computer model of the target. Each ray is traced in space until it eventually exits the target and the scattered field is calculated according to ray optics. Since the rays can undergo multiple reflection/diffraction, multiple scattering can be accounted for. Ray tracing is a key step in the implementation of this method.

The standard ray tracing algorithm available from computer graphics is based on the binary space partition (BSP) tree algorithm [16]. The BSP tree is generated by recursively cutting the bounding box of the target along a spatial plane. For a target consisting of N facets, the computational complexity of the algorithm for each ray scales between the optimal lower bound of $O(\log N)$ to the upper bound of $O(N)$, depending on the actual spatial distribution of target facets. Although the BSP-tree based ray tracer is often considered the fastest algorithm for ray tracing, its time performance still increases with the complexity of the target. High-frequency prediction in applications such as RCS design and radar target recognition demands ray tracing algorithms with even higher performance.

1.1.2 Scattering Feature Extraction from Scattering Data

In addition to the prediction of electromagnetic scattering from complex targets, an equally important problem is the extraction, interpretation and exploitation of different scattering features from the scattered field data. The development of effective algorithms for scattering feature extraction is useful in scattering data analysis, diagnostics and compression. It is also a key step to the development of automatic target recognition (ATR) algorithms.

Many scattering feature extraction techniques have been developed in the radar signature community. One of the most standard models to describe scattering from complex targets is the scattering center model [17-23]. Basically, the scattering center model assumes that at high frequencies the scattering response of a target can be well

approximated as a sum of responses from individual point scatterers, or scattering centers, on the target. The model has been used with success by the radar signature community for over three decades. Two main approaches have been developed to locate the scattering centers: Fourier-based processing [24] and model-based processing [17-23]. The Fourier approach is fast and robust, but results in limited resolution because it computes the inverse Fourier transform from frequency samples over a limited radar bandwidth. Model-based processing can achieve higher resolution than that of the Fourier-based process, but at the cost of time and robustness.

While the scattering center model has been very successfully utilized in radar, it has two key drawbacks. First, the scattering center model does not take into account more complex frequency dependent phenomena that can be present in real targets. One example is the strong frequency resonances that can arise from partially open cavities on a target, such as engine inlets and cockpits on an aircraft. Therefore, improved models that combine both scattering centers and frequency resonances have been proposed in [25, 26]. Correspondingly, the parameterization problem to determine the model parameters (scattering center location/strength and resonant frequency/strength) also becomes more challenging for these more sophisticated models. CLEAN [27], genetic algorithms [22, 23] and evolutionary programming-based CLEAN [28] are among some of the parameterization techniques that have been proposed for these more complicated models. However, there still exist open problems including dynamic range limitation, model sparsity and convergence issues for these techniques.

The second major drawback of the scattering center model is that it does not take into account the complex angular behavior of real scattering mechanisms in targets. The simple scattering center model assumes that each point scatterer is isotropic and visible over all angles. Clearly this cannot be true in real targets, as scattering features tend to have finite regions of visibility. For example, a trihedral corner reflector is visible only over a 90 degree angular sector. Further complicating the matter is that a scattering feature can be shadowed over a range of angles by other parts of the target. Existing techniques can extract the spatial locations of the scattering centers one angle at a time [20]. However, the correspondence of the extracted scattering centers across aspect is still far from satisfactory [27, 29]. A method considering both the spatial locations and angular visibility of the scattering centers will provide much more insight into the scattering features on a complex target. However, this method does not currently exist.

1.1.3 Inverse Scattering

In addition to investigating the scattering phenomenology from known targets, an even more challenging problem is to determine information, e.g., the shape and material constituency, of an unknown target from the scattered fields. This class of problems is generally known as inverse scattering problems.

The most common electromagnetic inverse scattering technique is called inverse synthetic aperture radar (ISAR) imaging [24]. ISAR imaging is based on the Fourier transform and can be considered as a linearized form of electromagnetic inverse

scattering. While the algorithm is fast and robust in obtaining the approximate shape of an object, it suffers from resolution limitation and image artifacts due to multiple scattering phenomena [30, 31]. To overcome the shortcomings of ISAR imaging, researchers have also tried to attack the problem by more exact inverse scattering methods [32-42]. These methods are usually based on a forward electromagnetic solver and a model describing the target shape (or material). The problem is then cast into an optimization problem whereby the difference between the known scattered field from the target and the computed field from an assumed target model is minimized. If the forward electromagnetic solver can account for more complex scattering mechanisms, then the potential advantage of this type of approach is that better target shape reconstruction and higher resolution can be achieved. However, the price is the much larger computational cost to carry out the inversion.

Typically, the optimization can be carried out by either a local search or a global search approach. Local search techniques were adopted by the inverse scattering community in solving simple and small inverse problems [32]. While the local search algorithms could achieve satisfactory solutions for simple problems, they eventually failed when dealing with large targets or targets with more complex scattering mechanisms. In the past decade, the use of global search algorithms in electromagnetic inverse scattering became more popular. These algorithms include genetic algorithms (GA) [33-39], differential evolution (DE) [40] and hybrid global-local algorithms [41-42]. These global search approaches have a better chance at finding a satisfactory solution. However, they are usually plagued by slow convergence rates. In particular,

large conducting targets and targets with strong multiple scattering effects are particularly problematic [31].

1.2 Objectives of the Dissertation

This dissertation sets out to advance the state of the art in each of the areas we have reviewed above, namely, high-frequency electromagnetic scattering prediction, scattering feature extraction and inverse scattering algorithms. In particular, we first focus on the development of a new ray tracing algorithm based on the multi-platen Z-buffer (MPZ) concept. The objective is to improve on the performance of the traditional binary space partition tree algorithm, as speeding up the ray tracer will fundamentally impact the speed of high-frequency electromagnetic prediction.

The second focus of the dissertation is on the development of improved scattering feature extraction algorithms. One objective aims at developing a new scattering center extraction algorithm with high accuracy and sparsity. The algorithm will be based on a global model with both scattering centers and resonances to represent the scattering from complex targets. Another objective aims at developing a scattering feature extraction algorithm by considering not only feature location but also angular visibility. The visibility factor is an important component in identifying and interpreting scattering features in the data.

The third focus of the dissertation is on enhancing the performance of global inverse scattering algorithms. This research concentrates on inverse problems with strong multiple scattering effects, which cannot be well handled by the traditional ISAR imaging

technique. The objective is to explore a combination of algorithms including genetic algorithm, local search and tabu search to achieve a better convergence rate for this class of difficult inverse problems.

1.3 Organization of the Dissertation

This dissertation is organized as follows. In Chapter Two, the multi-platen Z-buffer ray tracing algorithm is introduced and its performance is evaluated against the binary space partition tree algorithm. A multi-aspect approach is also proposed to further improve its performance. In Chapter Three, a global model combined with genetic algorithm is presented to parameterize scattering data with both scattering centers and resonances. Next, a scattering feature extraction algorithm that accounts for angular visibility is developed and tested. Chapter Four focuses on inverse scattering of metallic objects with strong multiple scattering effects. A genetic algorithm based approach is first studied. A hybrid algorithm combining GA, local search and the tabu search concept is then proposed to enhance the convergence rate of the GA. Chapter Five contains the conclusions of this research.

CHAPTER TWO

RAY TRACING FOR HIGH-FREQUENCY ELECTROMAGNETIC SCATTERING COMPUTATIONS

This chapter focuses on the development, evaluation and improvement of a new ray tracing algorithm aimed at speeding up high-frequency electromagnetic scattering computations. The traditional ray tracing algorithm based on the binary space partition (BSP) tree is first introduced. The multiplaten Z-buffer (MPZ) ray tracing algorithm is then implemented and its performance is evaluated against the traditional BSP tree algorithm. Results indicate that the computational complexity of the MPZ is independent of the number of facets, making it potentially attractive for targets described by a large number of facets. A multi-aspect MPZ (MAMPZ) is also proposed to further enhance the speed performance of the MPZ algorithm. Finally, the extension of the MPZ algorithm to targets described by initial graphic exchange specification (IGES) surfaces is investigated.

2.1 Introduction

As discussed in Chapter One, frequency-domain electromagnetic solvers are usually divided into low-frequency and high-frequency methods. Low-frequency methods, such as the method of moments (MoM), are based on the exact formulation of Maxwell equations. Their solutions can converge to the exact solution if the

discretization requirement is satisfied. The main drawback of these methods is that they usually lead to a dense and complex matrix equation. In the high frequency regime where the scatterer is large in terms of the radar wavelength, the matrix system may contain millions of unknowns and very large computer memory and computation time are required to solve the system.

High-frequency methods are based on the asymptotic approximation to the rigorous solution of Maxwell's equations and are usually based on ray tracing. In high frequency scattering, the physical optics method is typically used in the analysis of the first order scattering. The shooting and bouncing ray (SBR) method is a popular technique to account for the multiple scattering. The major advantage of the SBR method is its applicability for very high-fidelity computer models of complex targets.

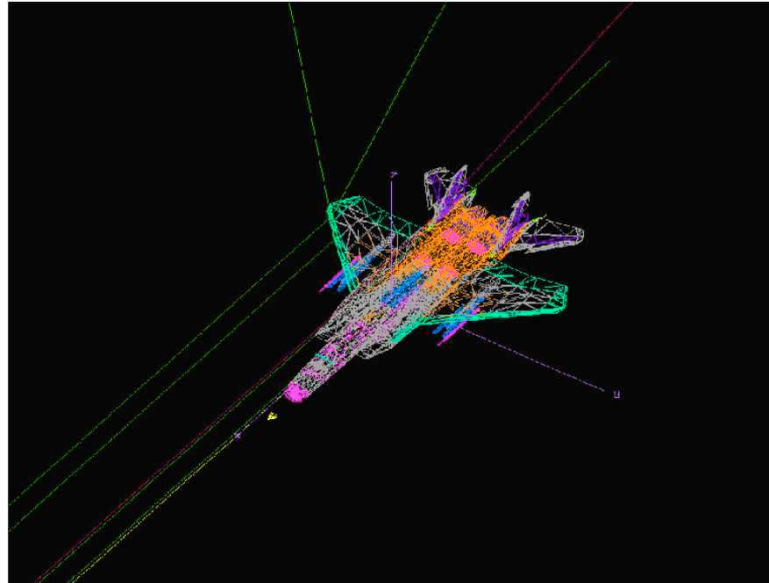


Fig. 2.1 Ray tracing implementation for EM scattering computation

In the SBR implementation, ray tracing is used to determine the ray trajectories launched from a transmitter to a given target geometry (Fig. 2.1). Each ray is traced in space and wave propagation is calculated according to ray optics. The trace terminates until it eventually exits the target bounding box. A large number of rays need to be launched from the transmitter to properly interrogate the target geometry. The ray density is typically on the order of ten rays per wavelength.

Ray tracing is a key step for the SBR method [14, 15]. The determination of where a ray hits the target is the most time consuming process in ray tracing. Although the SBR method is more accurate than the physical optics computation, it is also computational much more time consuming, especially for complex targets. The standard ray tracing algorithm, developed by the computer graphics community, is based on the binary space partition (BSP) tree algorithm [16]. In the standard BSP algorithm, a BSP tree is first built based on the facet model of the target by recursively cutting the bounding box of the object along a spatial plane. Fig. 2.2 illustrates a sample construction showing both the geometry space and its BSP tree at successive steps. A plane is first selected to partition the bounding box into two half-spaces (or voxels). Each half-space contains the facets located inside that half-space. The process is continued recursively by choosing another plane within each voxel until eventually each voxel contains only a small number of facets. Ray tracing is then performed by traversing the BSP tree for each ray. The traversal starts from the root of the BSP tree by finding if there is any intersection between the ray and the bounding box. If there is, the two voxels associated with the root voxel are checked to find which one has the intersection with the ray. The intersection is

recursively checked down to the bottom voxel in the BSP tree and all the facets contained in that voxel are then checked to find the exact facet with which the ray intersects.

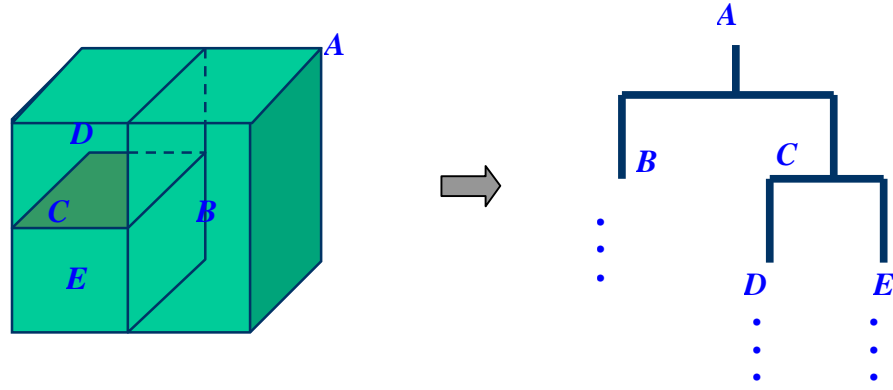


Fig. 2.2 Sample construction of a BSP tree at successive steps

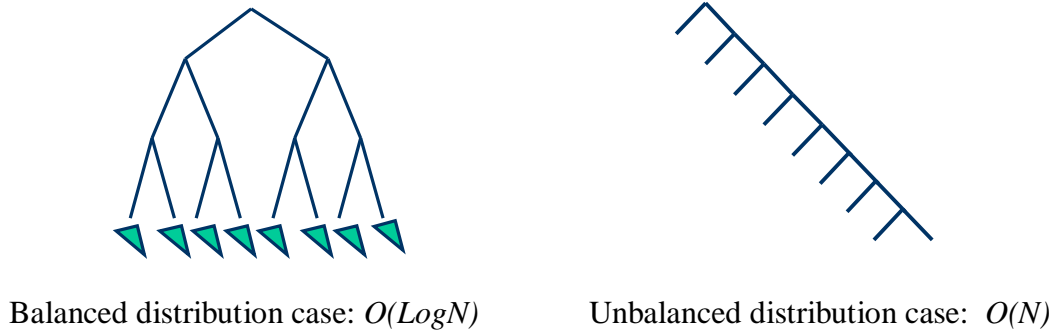


Fig. 2.3 Ray tracing performance vs. BSP tree structure

The performance of BSP tree ray tracing depends on the depth of the BSP tree. Fig.2.3 demonstrates two sample BSP trees associated with the object with a balanced facet distribution and the one with a highly unbalanced facet distribution (e.g., a cavity), respectively. For a target consisting of N facets, the computation time of the BSP tree ray tracing scales between the optimal lower bound of $O(\log N)$ for a balanced tree to the

upper bound of $O(N)$ for a highly unbalanced tree, depending on the actual spatial distribution of target facets. Although the BSP-tree based ray tracer is considered the fastest among all of the spatial subdivision approaches, its performance still increases monotonically with the number of facets (linearly in the case of a highly unbalanced facet distribution). A faster ray tracing algorithm is very much desirable in order to speed up the scattering prediction from complex targets with a large number of facets.

This chapter is organized as follows. In Section 2.2, the multiplaten Z-buffer (MPZ) ray tracing algorithm recently proposed by Hu et al [43, 44] is first reviewed as an alternative to the traditional BSP tree algorithm and its implementation is described. In Section 2.3, the performance of the MPZ is evaluated and compared against that of the BSP algorithm based on the test results from a wide range of targets. A new multi-aspect approach is then proposed in Section 2.4 to further speed up the computation performance of the MPZ algorithm. In Section 2.5, the extension of the MPZ ray tracer to IGES parametric surfaces is investigated. Section 2.6 is the summary of this chapter.

2.2 MPZ Algorithm and Its Implementation

Z-buffers were originally developed to remove hidden surfaces in computer graphics displays. Rius et al first adopted the usage of the Z-buffer in the GRECO code in the early 1990s to carry out physical optics calculations [9]. To set up the Z-buffer, the target is first rotated so that the z-direction is aligned to the incident direction. A scan conversion is then processed facet by facet. Only the z depth closest to the observer (i.e., the visible surface of the target) is stored into the Z-buffer in a pixel form (Fig. 2.4, 2.5).

At the same time, the color value, which is associated with the normal component n_z of the illuminated surface, is stored in a corresponding "frame buffer".

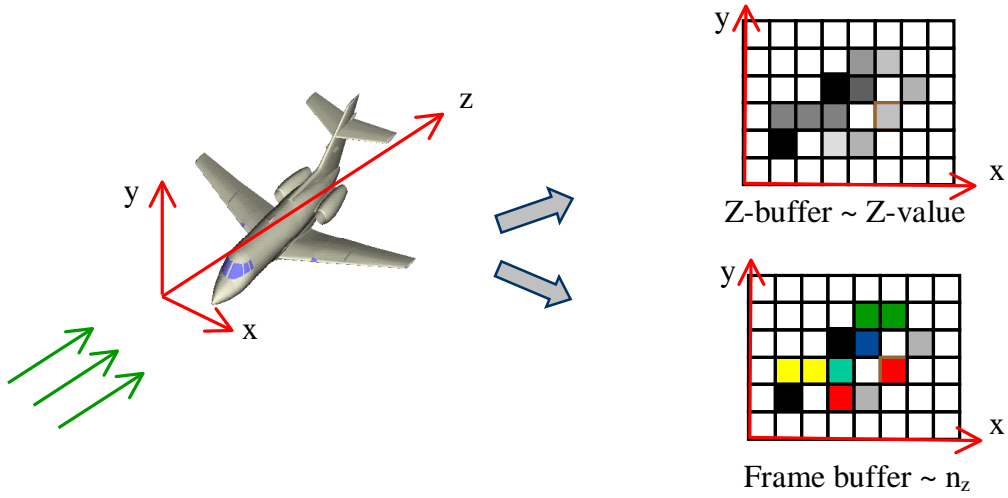


Fig. 2.4 Setup of single-layered Z-buffer

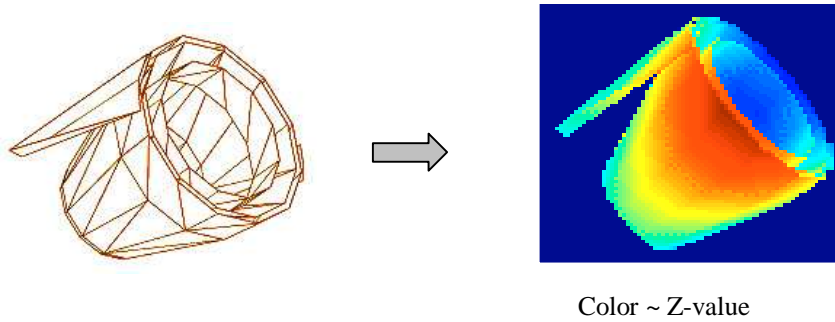


Fig. 2.5 Example of single-layered Z-buffer

The Z-buffer algorithm can take advantage of the high-performance graphics hardware for surface rendering to carry out PO calculations. Specifically, the radar cross

section (RCS) of a target can be quickly calculated by the equation [9]:

$$\begin{aligned}\sigma &= \frac{4\pi}{\lambda^2} \left| \int_{lit} (-\hat{k} \cdot \hat{n}) e^{i2kz} ds \right|^2 \\ &\cong \frac{4\pi}{\lambda^2} \left| \sum_{pixel} n_z e^{i2kz} \Delta x \Delta y \right|^2\end{aligned}\tag{Eq. 2.1}$$

However, higher order scattering contribution is totally omitted in this approach.

In the late 1990s, the multiplaten Z-buffer (MPZ) ray-tracing algorithm was proposed by Hu et al [43, 44] as an alternative to the traditional ray tracing algorithm. In the multiplaten Z-buffer approach, a multi-layered Z-buffer is first generated from the scan conversion process. The multi-layered Z-buffer is a natural extension of the traditional single-layer Z-buffer described above. Instead of just storing the z-coordinates of the visible pixels as in the traditional Z-buffer process, multi-layered Z-buffers are created to store the z-coordinates of all of the facets within each pixel during the scan conversion (Fig. 2.6). At the same time, a frame buffer is set up to store the corresponding facet numbers. The completed MPZ contains the three-dimensional coordinates of the target in a view-specific coordinate system. Fig. 2.7 shows an example of the multiplaten Z-buffer scan conversion for a simple facet model shown in Fig. 2.5. Here, besides the visible surface (the first layer Z-buffer), another four layers are used to store all the z values of the target.

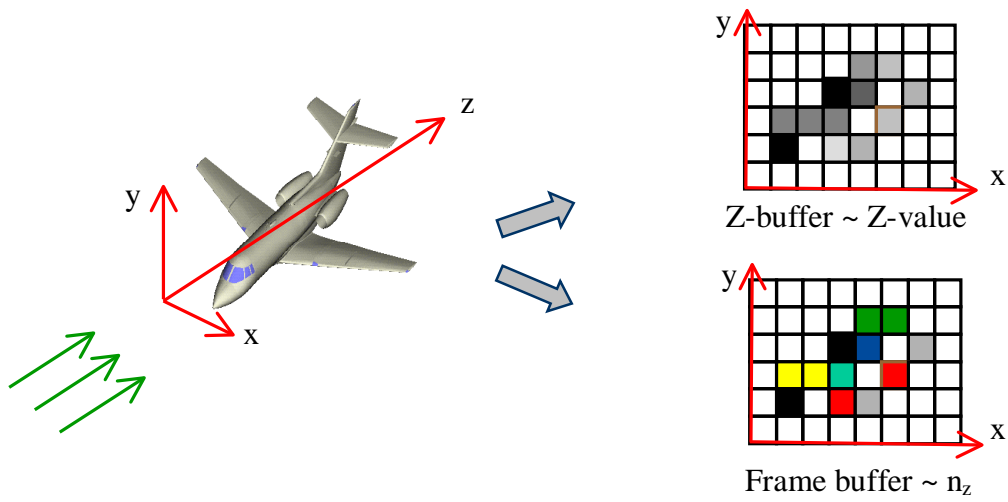


Fig. 2.6 Setup of the multiplaten Z-buffer

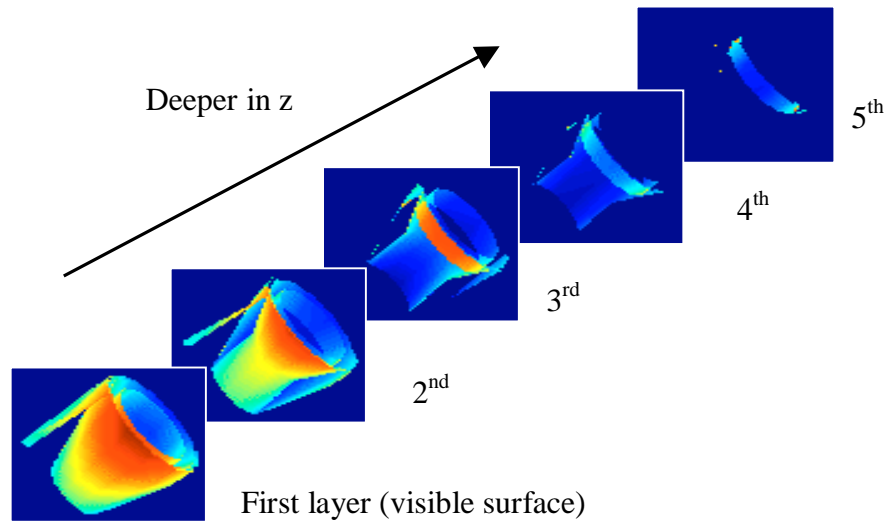


Fig. 2.7 Example of multiplaten Z-buffer

During the ray trace, the MPZ ray tracer propagates a ray along the ray direction

pixel by pixel (Fig. 2.8). Within every pixel tube, the z depths for both the entry and exit points are first calculated. These two z depths are then compared with all of the Z-buffer values within the pixel to check if there are any z values between them. If not, the ray is propagated to the next pixel tube. If yes, an intersection is reported. The closest Z-buffer value to that of the entry point is selected and the corresponding facet number is looked up in the frame buffer. An intersection routine is then executed to calculate the hit point and the reflected ray direction. To calculate the intersection between a ray and a facet, the ray is first expressed in parametric form as:

$$R(t) = P + \hat{R} \cdot t, \quad t > 0 \quad (\text{Eq. 2.2})$$

where P is the starting point of the ray and \hat{R} is the ray direction unit vector. The facet plane can be written as:

$$N_x \cdot x + N_y \cdot y + N_z \cdot z + D = 0 \quad (\text{Eq. 2.3})$$

where $N_x^2 + N_y^2 + N_z^2 = 1$ and (N_x, N_y, N_z) is the unit normal vector of the plane. The distance from the ray's starting point P to the plane is derived by simply substituting the expansion of equation (Eq. 2.2) in to the plane equation (Eq. 2.3) and solving for t:

$$t = \frac{-(N_x \cdot P_x + N_y \cdot P_y + N_z \cdot P_z + D)}{N_x \cdot R_x + N_y \cdot R_y + N_z \cdot R_z} = \frac{T_0}{T_1} \quad (\text{Eq. 2.4})$$

To use (Eq. 2.4) more efficiently, the denominator T_1 is first calculated. If $T_1 = 0$, then the ray is parallel to the plane and there is no intersection. Otherwise, if T_1 is non-zero, the ratio of T_0 and T_1 is calculated to get t. If $t < 0$, the ray intersects the plane behind the

ray starting point, and thus no actual intersection occurs. Else, the intersection point can be calculated by substituting the obtained t into (Eq. 2.2). Finally, the reflected ray direction is calculated by:

$$\vec{R}_{new} = \hat{R} - 2 \cdot (\hat{N} \cdot \hat{R}) \hat{N} \quad (\text{Eq. 2.5})$$

where \hat{N} is the unit facet normal vector and \vec{R}_{new} is the reflected ray direction.

This procedure is repeated from pixel to pixel until the ray eventually exits the bounding box, as shown in Fig. 2.9.

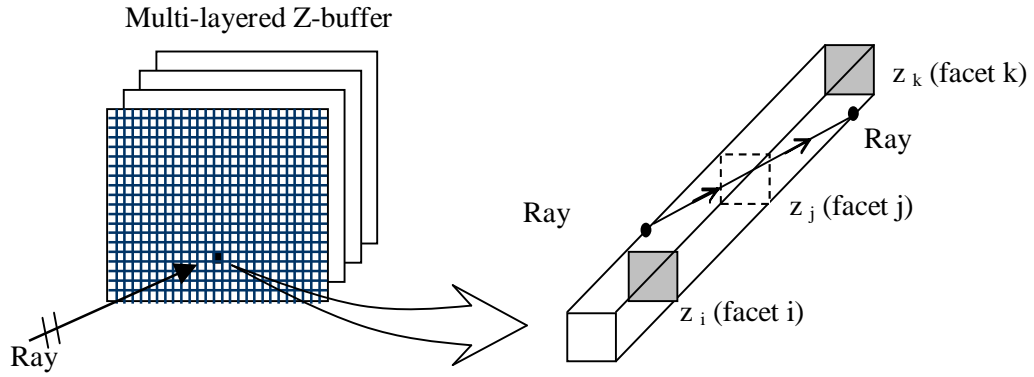


Fig. 2.8 Mechanism of the MPZ ray tracing algorithm

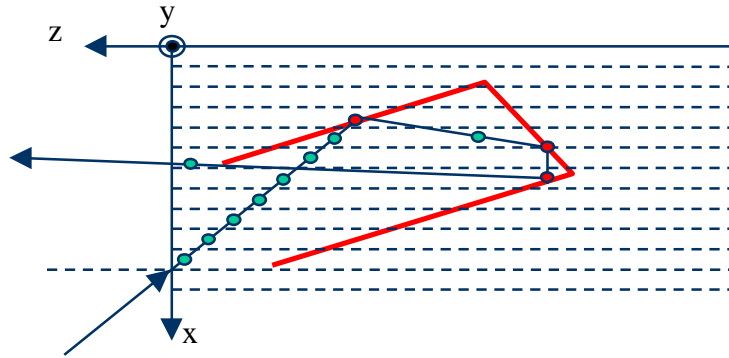


Fig. 2.9 Demonstration of a single ray tracing by MPZ

2.3 Evaluation of the MPZ Against the BSP

Although it was claimed that the MPZ algorithm should be superior to the traditional ray-tracing process, no evaluation about its time performance was reported in [43, 44]. Here an evaluation of the performance of the MPZ against that of the BSP algorithm is carried out. Multiple facet models are constructed for a straight rectangular cavity structure of size $2\text{m} \times 2\text{m} \times 20\text{m}$ (open at one end and closed at the other). The number of facets varies from 322 to 72002 (Fig. 2.10 shows two such models). A ray check is first carried out by running both the MPZ and the BSP ray tracer with the same starting points and directions on the rectangular cavity. A typical ray check is shown in Table 2.1. As shown, the check ensures that both algorithms trace the ray correctly.

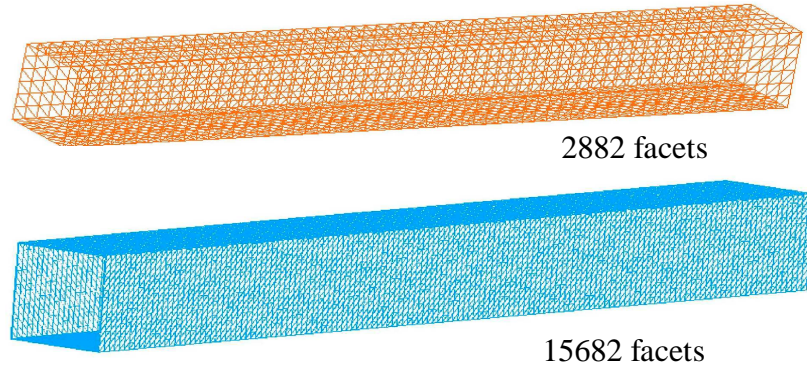


Fig. 2.10 Faceted rectangular cavity structures

Both the BSP and MPZ algorithms are then run on all of the facet models to compare and evaluate their computational performance. 300 rays are launched along the $\phi=135^\circ$ and $\theta=30^\circ$ direction. It should be noted that only ray tracing time of the two

algorithms are included in the comparison. The times for the BSP tree building and the MPZ scan conversion are not included. This is because in practice, the ray tracing time of millions of rays dominates the overhead time to set up the BSP tree or the MPZ scan conversion. Fig. 2.11 shows the result of the average ray tracing time per ray vs. the

Table 2.1 Ray check for MPZ algorithm

Bounce	BSP			MPZ		
i	X(i)	Y(i)	Z(i)	X(i)	Y(i)	Z(i)
0	0.2880	-2.2166	13.0071	0.2880	-2.2166	13.0071
1	-1.0000	-0.9286	9.8520	-1.0000	-0.9286	9.8520
2	0.9286	1.0000	5.1280	0.9286	1.0000	5.1280
3	1.0000	0.9286	4.9530	1.0000	0.9286	4.9530
4	-0.9286	-1.0000	0.2290	-0.9286	-1.0000	0.2290
5	-1.0000	-0.9286	0.0541	-1.0000	-0.9286	0.0541
6	0.9286	1.0000	-4.6699	0.9286	1.0000	-4.6699
7	1.0000	0.9286	-4.8449	1.0000	0.9286	-4.8449
8	-0.9286	-1.0000	-9.5689	-0.9286	-1.0000	-9.5689
9	-1.0000	-0.9286	-9.7439	-1.0000	-0.9286	-9.7439
10	-0.8954	-0.8240	-10.000	-0.8954	-0.8240	-10.000
11	0.9286	1.0000	-5.5321	0.9286	1.0000	-5.5321
12	1.0000	0.9286	-5.3571	1.0000	0.9286	-5.3571
13	-0.9286	-1.0000	-0.6331	-0.9286	-1.0000	-0.6331
14	-1.0000	-0.9286	-0.4582	-1.0000	-0.9286	-0.4582
15	0.9286	1.0000	4.2659	0.9286	1.0000	4.2659
16	1.0000	0.9286	4.4408	1.0000	0.9286	4.4408
17	-0.9286	-1.0000	9.1648	-0.9286	-1.0000	9.1648
18	-1.0000	-0.9286	9.3398	-1.0000	-0.9286	9.3398

number of facets for both of the ray tracers. Since the cavity facets are not distributed uniformly in space, the resulting BSP tree is highly unbalanced. Consequently, the ray

tracing time of the BSP algorithm increases nearly linearly with the number of facets. On the contrary, the performance of the MPZ algorithm is almost independent of the number of facets. Although the crossover point between the two algorithms does not occur until the number of facets is very large (>70000), it should be noticed that the MPZ algorithm was implemented without any optimization, while the BSP algorithm is a standard published routine [16].

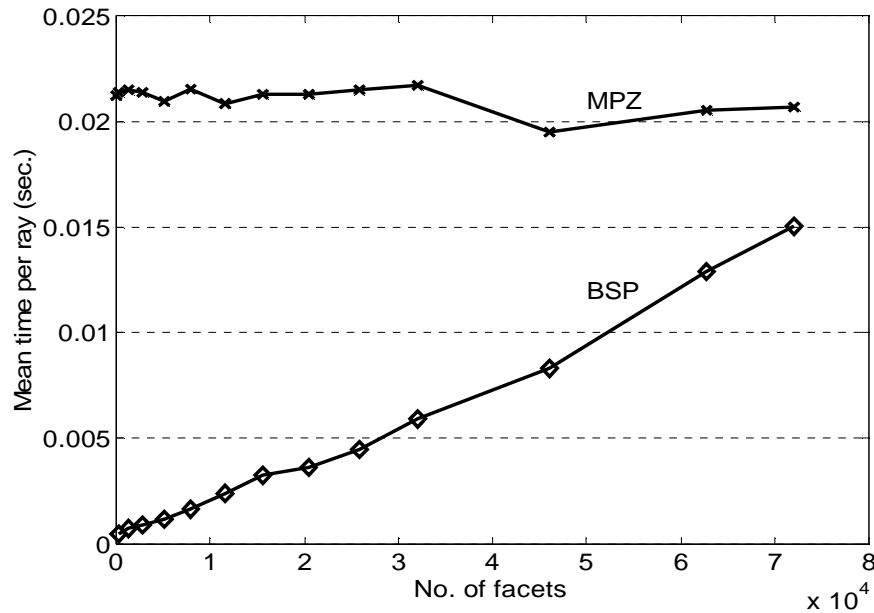


Fig. 2.11 Performance comparison of the BSP and MPZ algorithms for the faceted rectangular cavities

The MPZ performance should theoretically be independent of the number of facets, but the curve from the test result in Fig. 2.11 shows small undulations. Closer examination reveals that a small portion of the rays are not traced correctly. The missed intersections are predominantly caused by the z-value gaps shown in Fig. 2.12. These

gaps between neighboring pixels occur after the scan conversion of a facet, and gets larger for a facet that is nearly parallel to the Z-buffer direction. As a result, a ray can miss the facet, especially when the ray direction is nearly perpendicular to the z-direction. This problem will be further taken up in the next section. Another minor cause of the miss-traced rays is due to the pixel quantization at the facet edges. Incorrect intersections can occur due to the serrated edge approximation by pixels as oppose to the original linear edge of a facet. However, this problem is a minor one, as such errors decrease with a finer pixel size.

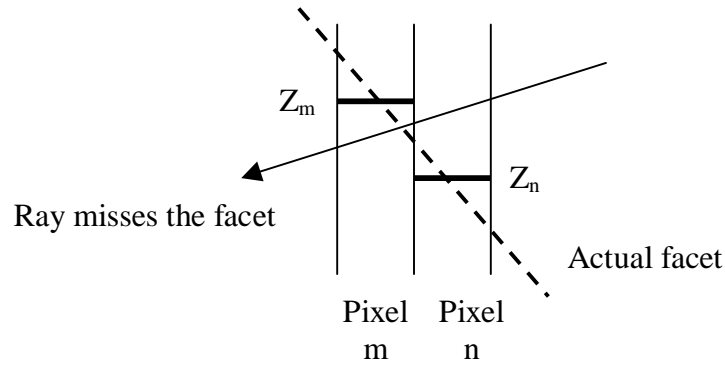


Fig. 2.12 Missed ray at large z value gap between two adjacent pixels

Next, the effect of the number of MPZ layers on the ray tracing time is investigated. The same rectangular cavity is used to evaluate the computation time. Since only two Z-buffer layers are enough to describe the rectangular cavity, dummy z-values (zeros) are used to pad the extra MPZ layers. This scheme guarantees that all the conditions are kept the same except for the number of MPZ layers. Fig. 2.13 shows the

relationship between the average MPZ ray tracing time per ray and the number of MPZ layers for the rectangular cavity. The result indicates that the ray tracing time only increases slightly with the number of layers. This result is not surprising, since the computational overhead to check for a ray intersection against the stored Z-buffer values within a pixel is very small.

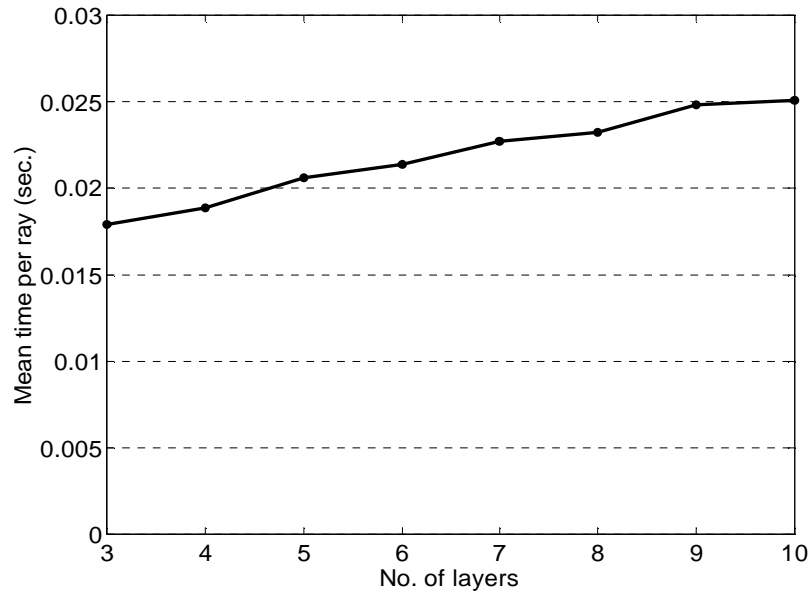


Fig. 2.13 Relationship between the averaged time per ray and the number of MPZ layers

Table 2.2 shows the performance comparison between the MPZ and BSP algorithms for nine complex targets(Fig. 2.14) where the number of facets varies from 126 to 56556. Because the complexity of the targets is different, the average number of ray bounces varies from target to target and no apparent performance trend can be found. However, the results do show a smaller performance gap between the MPZ and BSP algorithms as the number of facets increases.

Table 2.2 Time performance comparison between the BSP and MPZ algorithms for complex targets

Target	1	2	3	4	5	6	7	8	9
No. of Facets	126	800	958	3992	5974	7994	10090	23988	56556
BSP Average Time Per Ray (ms)	0.13	0.12	2.50	16.84	13.02	5.16	4.61	5.64	56.48
MPZ Average Time Per Ray (ms)	5.35	3.46	8.98	10.04	11.00	18.45	6.14	12.91	43.66

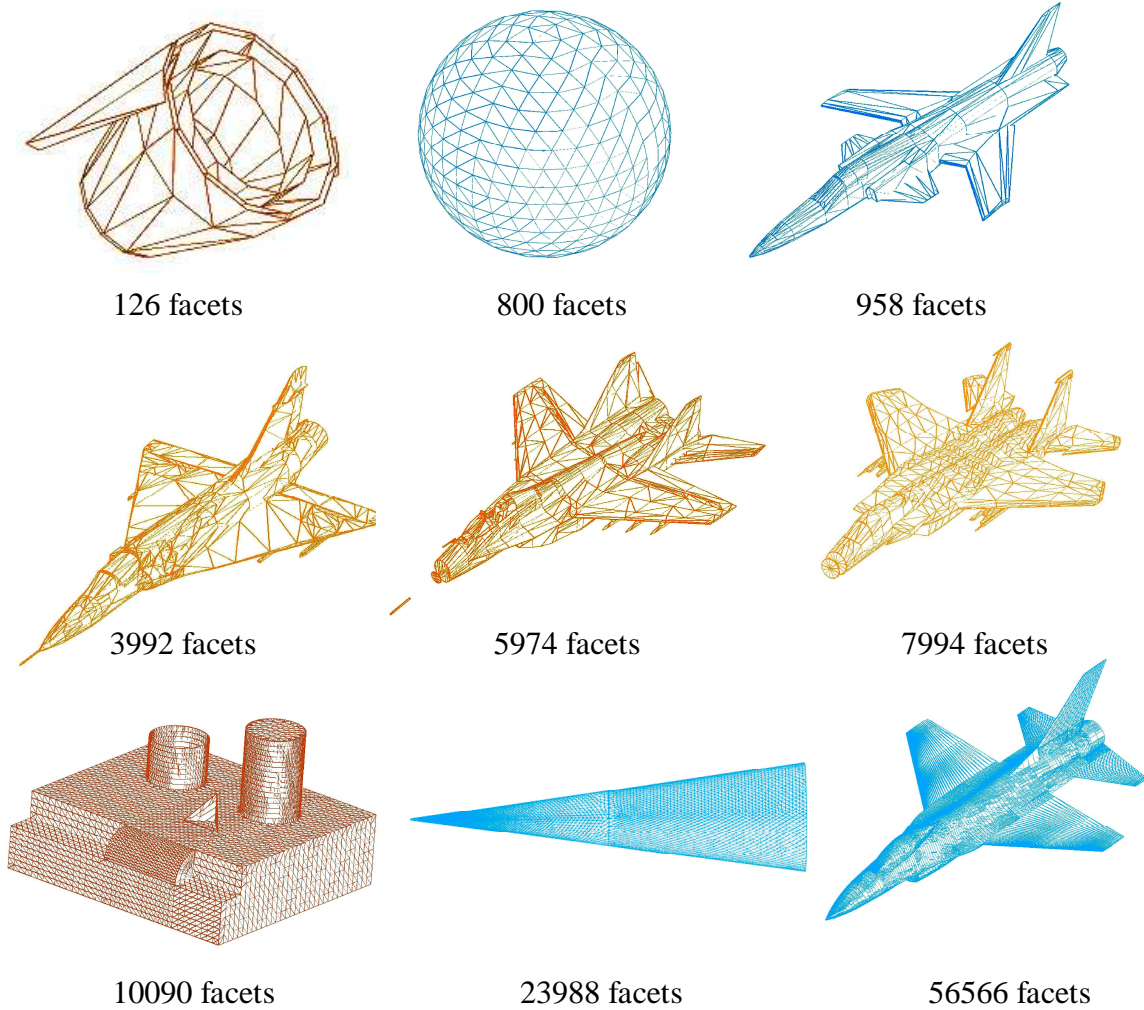


Fig. 2.14 Geometries of nine tested targets

2.4 Multi-aspect MPZ (MAMPZ)

While the performance of the MPZ ray tracing algorithm does not depend on the number of facets, it does depend directly on the number of pixels a ray traverses. Therefore, if the number of pixels a ray traverses between bounces can be reduced by decreasing the angle between the ray and the Z-buffer directions (Fig. 2.15), the MPZ time performance could be further improved. Here a multi-aspect MPZ (MAMPZ) approach is proposed. In the MAMPZ, multiple multi-layered Z-buffers are first generated from the scan conversion process along many aspect angles. The maximum number of multi-layered Z-buffers is limited only by the available computer memory. The more aspect angles that are stored, the fewer pixels a ray needs to traverse in one bounce, and the better the time performance. During the ray trace, the multi-layered Z-buffer structure that has the closest aspect to the ray direction is selected to carry out the ray tracing (Fig. 2.16). A ray is then tracked by moving along the ray direction inside this MPZ structure pixel-by-pixel to check for possible intersections.

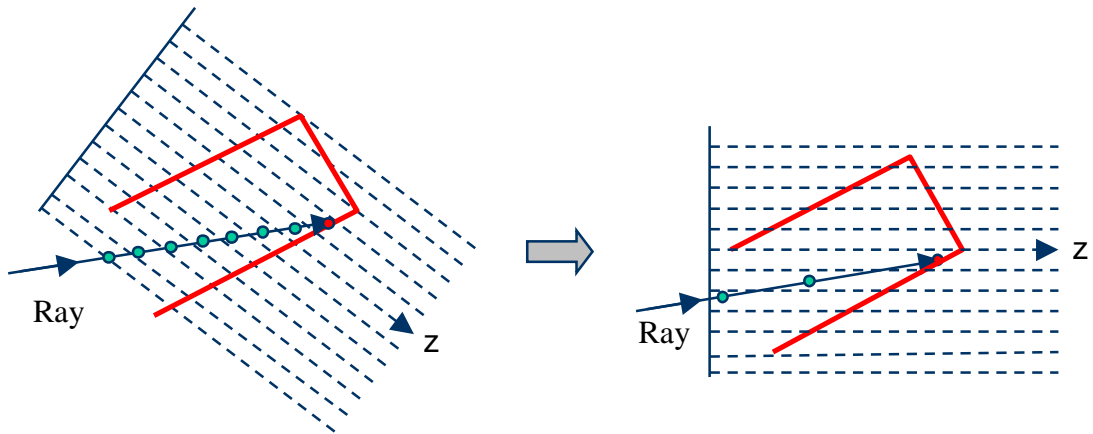


Fig. 2.15 Scheme for reduction of the number of pixels a ray travels across

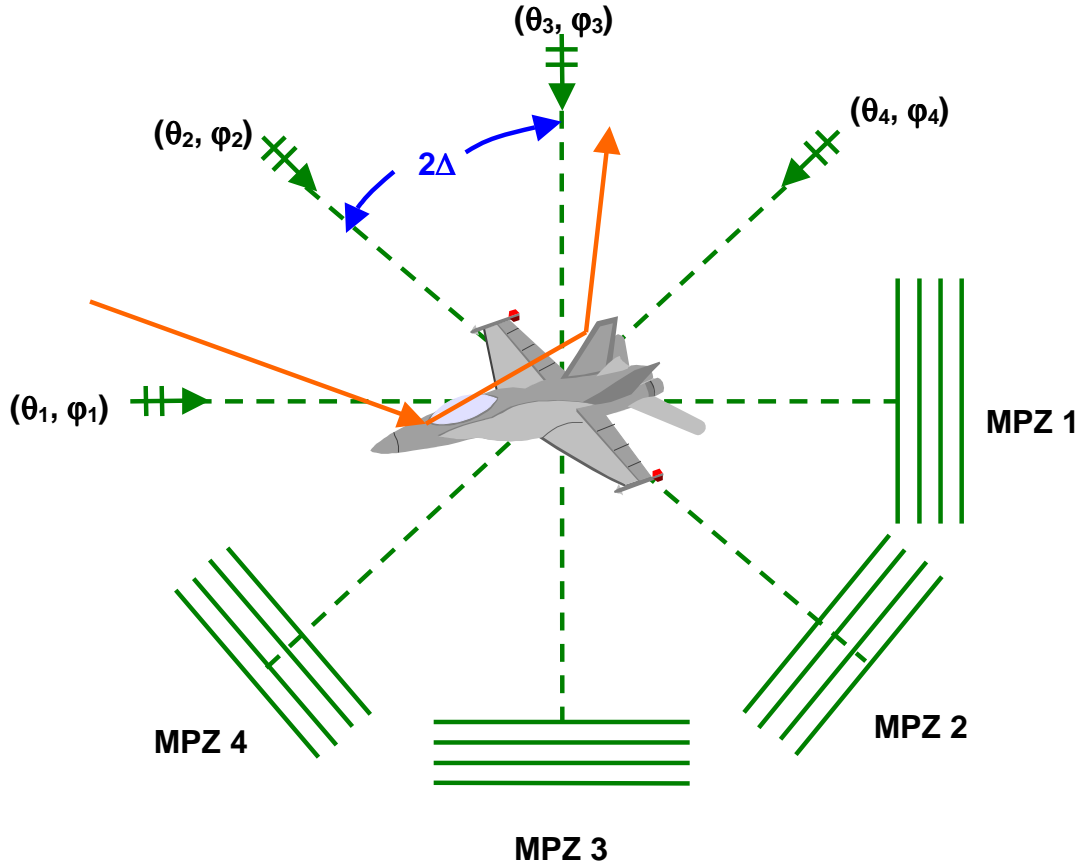


Fig. 2.16 Scheme of multi-aspect MPZ approach

The computation time performance of the MAMPZ ray tracer is evaluated against that of the single-aspect MPZ and the BSP tree algorithm on the rectangular cavity. Generally, the traced rays for an object can be along any direction. If multiple MPZs are set up with an angular interval Δ along both the θ and ϕ directions, it will guarantee that every ray can find its closest MPZ with a maximum cross angle of $\Delta/2$. This angular interval $\Delta/2$, as shown in Fig. 2.16, will be called the MPZ aspect resolution in the following text. A fine aspect resolution means a large number of MPZs, which require a

large amount of memory to store. Nonetheless, this concept can be tested easily for the straight rectangular cavity shown in Fig. 2.10 since the ray directions can be limited to only two distinct directions by choosing the incident angle to be along $\phi=0$ (θ is chosen to be 45°). Thus, by setting up MPZs at $\Delta/2$ away from these two ray directions (Fig. 2.17), the performance of the algorithm with different MPZ aspect resolutions can be evaluated.

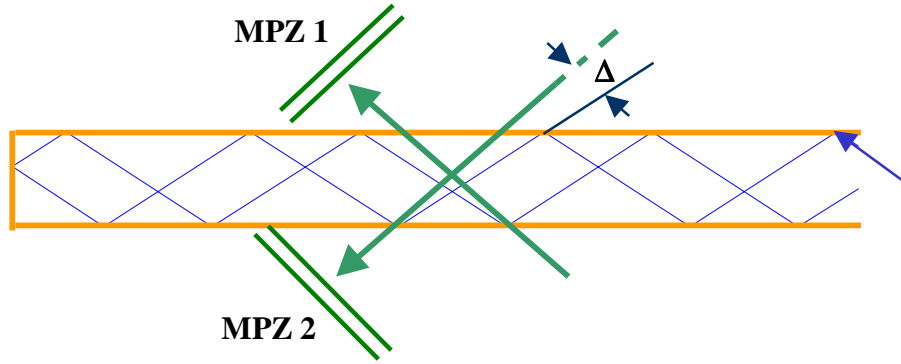


Fig. 2.17 Evaluation scheme for straight rectangular cavity

Fig. 2.18 shows the computational complexity of the MAMPZ algorithm as a function of the number of facets for two different MPZ aspect resolutions. The comparison indicates a 7-fold improvement in performance against that of the single-aspect MPZ even for a 20-degree aspect resolution. Another 14-fold improvement can be achieved for a 1-degree aspect resolution (for a 98-fold improvement over the single-aspect MPZ). Consequently, the new MAMPZ algorithm can outperform the BSP algorithm for targets with as small as a few thousand facets if the aspect resolution is fine enough. Fig. 2.19 shows the time performance of the MAMPZ algorithm for different aspect resolutions. The result displays a monotonic (and slightly nonlinear) decrease of

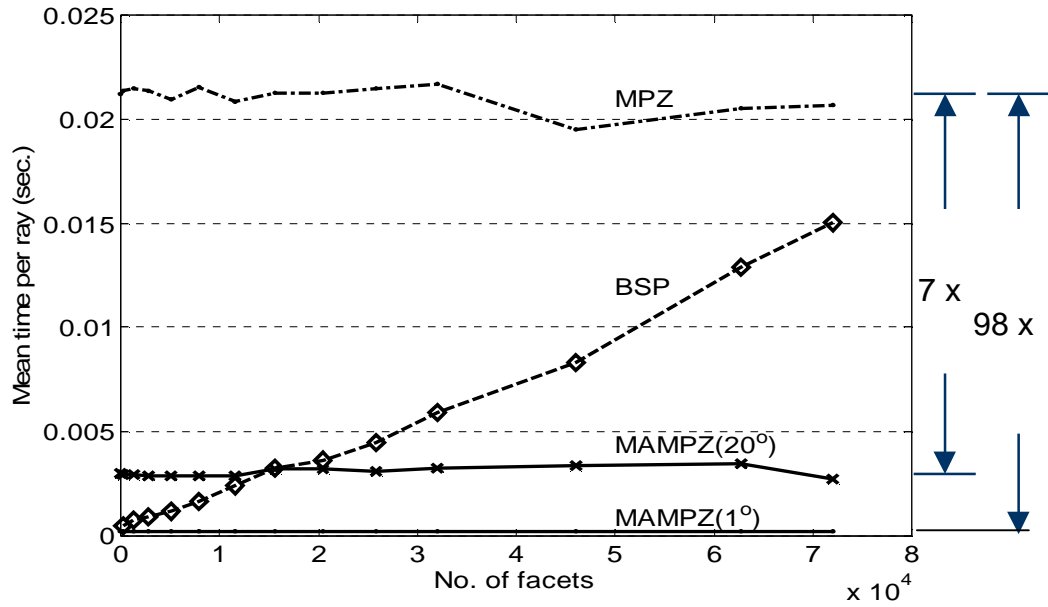


Fig. 2.18 Performance of the MAMPZ for two different aspect resolutions

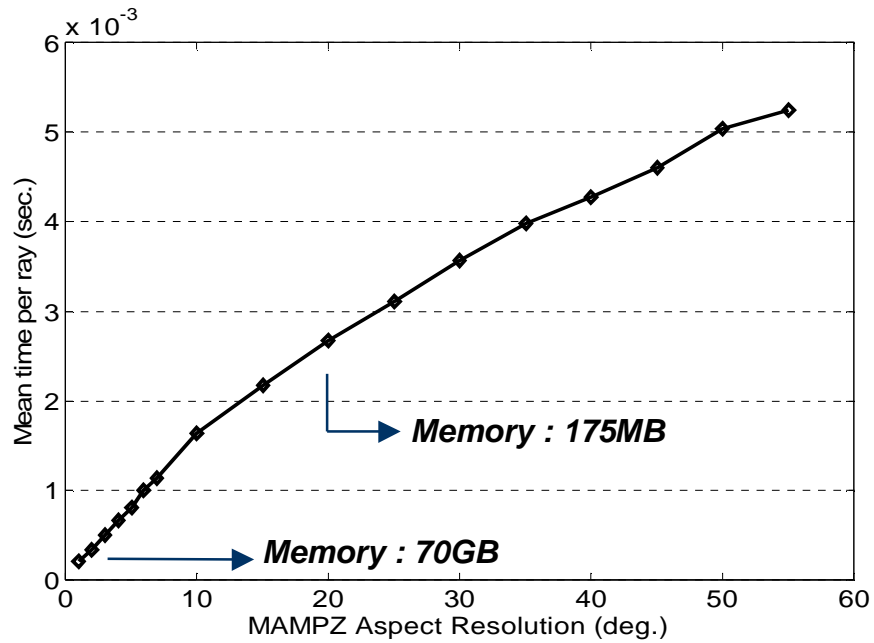


Fig. 2.19 Time performance for different aspect resolutions of the MAMPZ

ray tracing time versus the MAMPZ aspect resolution. The price for the improvement in performance is the large memory requirement to store the MPZs. For a single-aspect, 24-bit Z-buffer with 3 layers, the typical memory size is 8.6MB for a 1000×960 pixel resolution. If an aspect resolution of 20 degrees is used to set up uniformly distributed MPZs over all directions (note that only a hemispherical coverage is needed to cover all angles), the required memory is 175MB. For a resolution of 1 degree, the number becomes 70GB.

Another performance check for the MAMPZ approach is carried out on a Y cavity structure shown in Fig. 2.20. The cavity is open at the two bottom inlets and blocked at the upper end. Similarly, the ray directions can be limited to only six distinct directions by choosing the incident angle to be along $\phi=0$. Thus, by setting up MPZs at $\Delta/2$ away from these six ray directions, the performance of the algorithm with different MPZ aspect resolutions can be easily evaluated. The Y cavity structure is discretized into different number of facets ranging from 528 to 74496. 1100 rays are launched and traced by the BSP and MAMPZ algorithms, respectively. For the MZMPZ, two different aspect resolution sets (5° and 10°) are checked. Fig. 2.21 shows the performance comparison between the BSP and the MAMPZ for two different resolution cases. The performance gain of the MAMPZ over that of the single aspect MPZ is similar to the straight rectangular cavity case. Therefore, the MAMPZ can outperform the BSP algorithm for targets with as few as several thousand facets if the MAMPZ aspect resolution is fine enough.

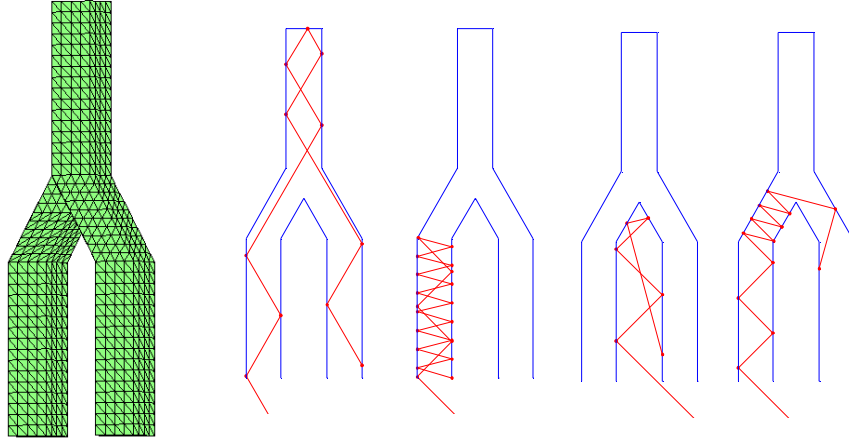


Fig. 2.20 Geometry of Y cavity and typical rays in $\theta=0$ plane

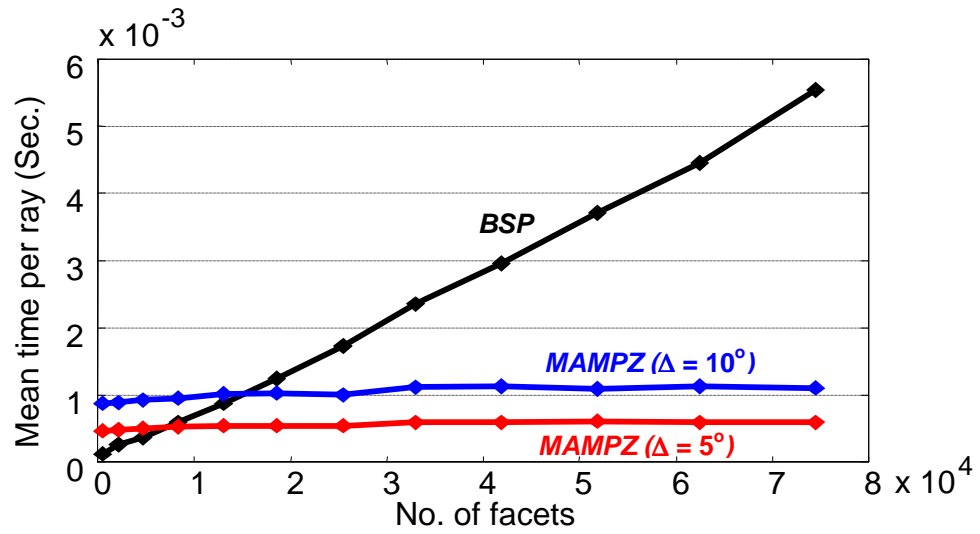


Fig. 2.21 Performance comparison for Y cavity

One important benefit of the MAMPZ algorithm is a decrease in the number of miss-traced rays. The results in Fig. 2.18 and Fig. 2.21 show the disappearance of undulations in the timing curves. As mentioned in the previous section, the miss-tracing

usually occurs when the ray direction is nearly perpendicular to the z-direction. In the MAMPZ algorithm, since each ray is traced by choosing the closest Z-buffer, the ray direction is always close to the z-direction (Fig. 2.22). For the straight cavity, the percentage of miss-traced rays decreases from above one percent for the single-aspect MPZ algorithm to below 0.2 percent for the 20-degree MAMPZ algorithm. This is an important advantage of the multi-aspect algorithm.

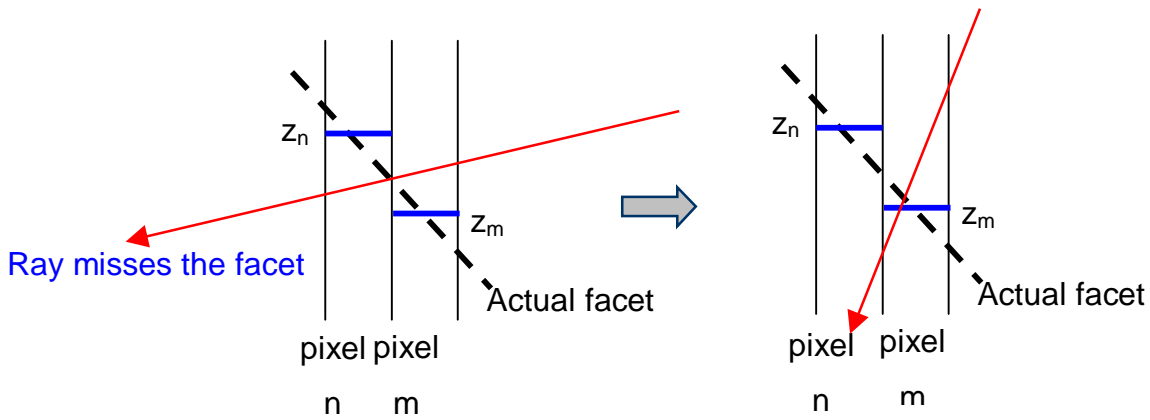


Fig. 2.22 Decrease of miss-tracing for MAMPZ

2.5 MPZ ray tracer for IGES Parametric Surfaces

High-fidelity computer models of targets are often described not by planar facets, but by parametric curved surfaces. The MPZ ray tracer can be extended to parametric curved surfaces by modifying the Z-buffer scan conversion and intersection calculations. As an example, the industry standard IGES114 [45] parametric surface is expressed as:

$$\left\{ \begin{array}{l} x = x(s, t) = [s^3 \quad s^2 \quad s \quad 1] \begin{bmatrix} c11 & c12 & c13 & c14 \\ c21 & c22 & c23 & c24 \\ c31 & c32 & c33 & c34 \\ c41 & c42 & c43 & c44 \end{bmatrix} \begin{bmatrix} t^3 \\ t^2 \\ t \\ 1 \end{bmatrix} = S^T C_x T \\ y = y(s, t) = S^T C_y T \\ z = z(s, t) = S^T C_z T \end{array} \right. \quad 0 \leq s \leq 1 \quad \text{and} \quad 0 \leq t \leq 1 \quad (\text{Eq. 2.6})$$

According to the above expression, the inverse mapping from a point (x,y,z) in 3-D coordinate system to its parameters (s, t) can not be solved analytically. Thus, the scan conversion from the parametric surface patch into the Z-buffer values will take longer time compared to the planar facet. In addition, the intersection between a ray and the parametric curved surface has to be calculated numerically.

The procedure of the MPZ ray tracer for IGES parametric surface is similar to that of the planar facet type, except for the special scan conversion for parametric patches and the structure of the frame buffer. To accomplish these steps, the following procedure is proposed:

- A. *Scan convert IGES parametric surface patches one by one, store z values into the z-buffer while storing the patch number and (s, t) pair into the corresponding frame buffer (Fig. 2.23);*
- B. *The MPZ ray tracer propagates along the ray direction pixel by pixel to detect possible intersection;*
- C. *When the possible intersection is detected, the algorithm looks up the corresponding frame buffer to obtain the patch number and (s, t) pair*

to start the numerical search for the exact intersection point and reflection direction (Fig. 2.24);

D. The process is repeated until the ray eventually exits the bounding box of the target.

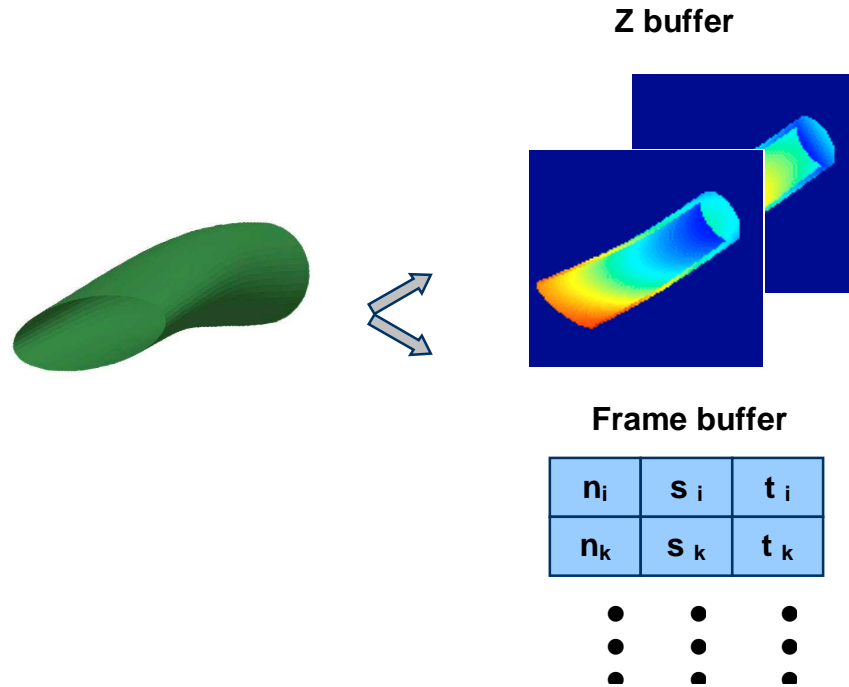


Fig. 2.23 Scan conversion for an inlet shape with 129 IGES114 parametric patches

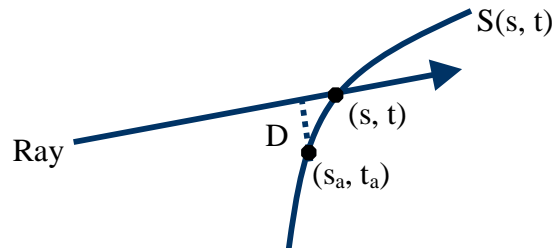


Fig. 2.24 Intersection detection between a ray and the parametric patch

It should be noted that in step A, every patch is first finely scanned in (s, t) space. The resulting point array in (x, y, z) space is then converted into the Z-buffer. The two-step scan conversion for IGES surface patch is different from that for the planar facet, which is done directly in the (x, y, z) space. In step C, the MPZ ray tracer first detects the possible intersection inside the multi-platen Z-buffer. A numerical search routine is then called to achieve the exact intersection point (s, t) based on the detected approximate location (s_a, t_a) . The numerical search tries to detect the point on the parametric surface which has the shortest distance to the ray, as shown in Fig. 2.24. The intersection occurs only when the shortest distance D falls within a small tolerance. It is a minimization problem requiring an iterative search process in (s, t) space to obtain the solution. Several algorithms, such as the Newton iteration [46] and quasi-Newton iteration [47], have been developed and implemented to detect an intersection on a parametric surface. In the MPZ implementation, the quasi-Newton iteration algorithm from [47] is adopted to search for the exact intersection point. As a numerical example, a ray trace is carried out for an inlet cavity described by 129 IGES114 patches (Fig. 2.23). The resulting ray history agrees well with that of the BSP tree based ray tracer, as shown in Table 2.3.

Table 2.3 Ray check for IGES parametric model ray tracing

Bounce No.	BSP			MPZ		
i	X(i)	Y(i)	Z(i)	X(i)	Y(i)	Z(i)
0	79.91	-98.39	-18.78	79.91	-98.39	-18.78
1	25.76	5.64	-55.76	25.73	5.65	-55.75
2	19.18	23.00	-41.71	19.13	23.00	-41.71
3	4.59	-21.92	-42.85	4.60	-22.00	-42.59
4	3.06	-1.49	-28.13	2.85	1.31	-28.06
5	-5.72	21.45	-39.49	-5.43	21.43	-39.88
6	-11.08	-14.67	-47.08	-10.70	-15.08	-46.88
7	-10.53	-18.81	-31.77	-10.04	-18.80	-31.81
8	-14.07	20.49	-34.69	-13.42	20.45	-34.53
9	-15.22	-0.51	-50.41	-14.74	1.64	-50.41
10	-13.69	-20.43	-34.48	-13.35	-20.65	-35.22

Ray tracing for parametric curved surfaces is typically much more time consuming than planar facets. A comment is therefore in order in comparing the MPZ ray tracer to other developed ray tracers for parametric surfaces. Existing ray tracers for parametric surfaces are based either on the space subdivision technique [48, 49] or the iterative search process. The former type facetizes the parametric surface into smaller planar facets and uses the standard planar facet ray tracer for the ray tracing, such as the BSP tree. The latter type is similar to the MPZ procedure described above. Usually, the convergence rates of the iterative search algorithms are sensitive to the initial guess. Here the MPZ ray tracing algorithm for IGES parametric surfaces has the potential advantage in its ability to detect a very close approximate intersection point (within a pixel) as the

initial guess for the local search. Thus, it can speed up the local search by saving the number of iterations required.

2.6 Summary

In this chapter, a ray tracer based on the multiplaten Z-buffer algorithm has been implemented and its performance has been evaluated against the standard BSP tree algorithm. Numerical results show that the computational complexity of the MPZ algorithm is independent of the number of facets and is only weakly dependent on the number of Z-buffer layers. The algorithm has been further extended by a multi-aspect MPZ approach. Results show that the extended algorithm can dramatically improve the ray tracing speed in comparison to the single-aspect MPZ by cutting down the number of pixels each ray has to traverse. The MAMPZ also reduces the chance of miss-traced rays, which are intrinsic to the Z-buffer quantization process. One price for the improvement in speed is the large memory requirement to store the MPZs. For a single-aspect, 24-bit Z-buffer with 3 layers, the typical memory size is 8.6MB for a 1000×960 pixel resolution. If an aspect resolution of 20 degrees is used to set up uniformly distributed MPZs over all directions (note that only a hemispherical coverage is needed to cover all angles), the required memory is 175MB. For a resolution of 1 degree, the number becomes 70GB. Provided that such memory resources are available, the MAMPZ algorithm can achieve a two order of magnitude acceleration in ray tracing time. This makes the algorithm quite attractive in certain modeling and simulation applications

where speed is critical. The MPZ algorithm was then extended to the ray tracing of IGES parametric surfaces. The MPZ IGES parametric surface ray tracer is potentially attractive since the numerical search for the intersection can be seeded with a very good initial guess.

CHAPTER THREE

FEATURE EXTRACTION FROM ELECTROMAGNETIC SCATTERING DATA

3.1 Introduction

In addition to the prediction of electromagnetic scattering from complex targets, the problem of extraction, interpretation and exploitation of different scattering features from the scattered field data is currently an area of great interest in radar signal processing. Analyses of these problems require that target scattering features be extracted from the radar data. The most commonly used model to describe the scattering characteristics of a target at high frequencies is the scattering center model. In this model, it is assumed that the scattering response of a target can be well approximated as a sum of responses from individual point scatterers [17]:

$$E(f) = \sum_{i=1}^N a_i(f) e^{j2\pi f t_i} \quad (\text{Eq. 3.1})$$

where t_i is the time delay of the i^{th} point scatterer and a_i is its strength. $a_i(f)$ is usually a complex function of frequency f . Physical scattering mechanisms derived from the geometrical theory of diffraction (GTD) indicate that $a(f)$ has an f^α dependence [17]. In most cases, the bandwidth of a radar system is less than 10% of the central frequency, and the frequency dependence is slow compared to the exponential function in Eq.3.1. In such cases, $a(f)$ can be approximated by a complex constant.

Two main approaches have been developed to locate the scattering centers: fast Fourier transform (FFT)-based algorithms and model-based techniques. The FFT-based technique is the traditional method of performing scattering center extraction. The technique requires uniform samplings in frequency, and the sampling rate must be dense enough to satisfy the Nyquist sampling rate. Raw radar data are first obtained in the frequency domain by either measurements or by means of EM prediction algorithms. The time-domain response (or the range profile) can be generated by a 1-D FFT of the radar data. The scattering centers are found by extracting the locations and amplitude of peaks in the range profile.

Though the FFT technique is computationally efficient, it has several disadvantages. Besides having strict requirement for the data sampling, the FFT technique can only achieve a range resolution limited by the Fourier limit. To extract high-resolution scattering features, large frequency bandwidth must be used. The disadvantages of the FFT technique can be reduced by model-based algorithms. Generally, model-based algorithms do not require even sampling. Moreover, they are found to be superior to the conventional FFT technique in resolution and dynamic range [18, 19]. However, the resolution advantage of model-based algorithms comes at the price of speed and robustness.

The scattering center model has been used with success by the radar signature community for over three decades. However, it has two main drawbacks. First, it does not take into account more complex frequency dependent phenomena, such as strong frequency resonances that can arise from partially open cavities on a target. Though

improved models that combine both scattering centers and frequency resonances have been proposed in [25, 26], the parameterization problem to determine the model parameters also becomes more challenging for these more sophisticated models.

The second drawback of the scattering center model is that the standard scattering center model assumes that each point scatterer is isotropic and visible over all angles and does not take into account the complex angular behavior of real scattering mechanisms in targets. Clearly, scattering features tend to have finite region of visibility. For example, a trihedral corner reflector is visible only over a 90 degree section in azimuth and elevation. Moreover, a scattering feature can be shadowed over a range of angles by other parts of the target. A useful methodology for extracting the 3-D scattering centers of a complex target was reported in [20]. However, the scattering centers in this approach are extracted one angle at a time, and information on the visibility of a scattering center across angles is not available. Later, research to rectify this problem was carried out but the results were not fully satisfactory [27, 29].

In this chapter, a model-based method is first presented in Section 3.2 to parameterize scattering data from complex targets. Based on a global model with both scattering centers and resonances, a genetic algorithm with adaptive feeding is proposed for a sparse representation of the data. The algorithm, when tested with measurement data, shows better performance than non-global parameterization methods. In Section 3.3, several extraction and clustering methods are investigated by considering both the spatial locations and angular visibility of the scattering centers. The results show that it is possible to categorize scattering centers by their angular visibility. Furthermore, the

developed methodology is capable of capturing scattering mechanisms that are not necessarily stationary in space, but move in a continuous fashion from angle to angle.

3.2 Sparse Parameterization of EM Scattering Data

3.2.1 Parameterization based on combined model and community GA

Obtaining a sparse, physical representation of electromagnetic scattering data from a complex target is a problem of fundamental importance in radar signature analysis [18-23, 25, 26, and 28]. A number of techniques, including super-resolution [18, 19], CLEAN [20], genetic algorithms (GA) [21-23], and evolutionary programming-based CLEAN [28], have been reported for determining the model parameters based on the scattering center model. For targets containing convex, interior structures such as cavities, a model combining scattering centers and resonances has been shown to be a more efficient and physically meaningful representation of both the exterior and interior scattering features [25, 26]. However, finding the model parameters in such cases is more challenging, because the scattering center and resonance bases have complementary behaviors in time and frequency. In [25], a CLEAN-based algorithm was used iteratively to extract one scattering center and/or resonance at a time. In [26], Prony's method was first used to extract all the scattering centers and then all the resonances. One drawback of these methods is that the parameterization results are not very sparse because the scattering centers and resonances are not extracted simultaneously.

For this research, the scattering model is assumed to comprise of responses from both scattering centers and resonances in the following form [26]:

$$E(f) = \sum_{m=1}^M a_m f^{\alpha_m} e^{-j2\pi t_m f} + \sum_{n=1}^N b_n \frac{1}{j2\pi(f - f_n) + \beta_n} e^{-j2\pi f \tau_n} \quad (\text{Eq. 3.2})$$

where M and N represent the number of scattering centers and resonances, respectively, and f is the frequency. For each scattering center, t_m is the time delay, α_m is the frequency dependency coefficient, and a_m is the complex amplitude. For each resonance of complex amplitude b_n , f_n is the resonant frequency, τ_n is the turn-on time, and β_n is the Q-factor. The parameterization process can be formulated as a minimization problem as follows:

$$\{\alpha_m, t_m, f_n, \beta_n, \tau_n\} = \arg \min \|E(f) - E^m(f)\|_2 \quad (\text{Eq. 3.3})$$

where E^m denotes the measurement data to be parameterized. The amplitudes a_m and b_n are not included in the bracket as they can be derived from other unknowns using minimum least squares fitting.

To improve the sparsity of the extracted parameters, the genetic algorithm (GA) approach is researched to parameterize complex scattering data using the combined scattering center and resonance model. The GA has been used in many engineering applications as a global optimization scheme [21-23, 33-37, 39, 41, 42]. In general, the GA does not operate directly on the unknown parameters, but on the coding representation of the unknown parameters, which is called the chromosome. Several operators are then performed on the code according to the concepts of selection, crossover, and mutation. A cost function must be defined in advance to evaluate the difference between the GA member and the real solution. A lower cost function value

means the solution it represents is a better approach to the real solution. Members with lower cost function values will have more chance to participate the creation of next generation. A simple description of the GA procession is as follows:

- a) Decide the unknown parameters of the problem and encode them to be chromosome.*
- b) Randomly create the initial generation members.*
- c) Evaluate the cost function value of each member.*
- d) Perform selection, crossover and mutation procession based on the cost function value of each member to produce the next generation.*

The process is then continued until a global solution, which satisfies the termination condition, is achieved.

For the combined scattering center and resonance model, however, it is found that the standard simple GA (SGA) has difficulty in converging to the desired global optimum. Since the energy in a resonant term is typically much lower than that in a scattering center, the resonant terms are easily missed in the SGA process. To overcome this problem, a parameterization based on a community GA (CGA) concept, which was previously reported in [50], together with a new adaptive feeding scheme is devised. In CGA, the entire population is broken up into subpopulations, known as communities. All of the members of a given community share the same community chromosome. For the problem of the combined scattering center and resonance model, different communities have the different composition of the number of scattering centers and resonances.

Fig. 3.1 illustrates the approach in extracting M scattering centers and N resonances. In this figure, each solid box represents a community using a different parameterization order number. For example, the highest-order community uses M scattering centers and N resonances, while the lowest-order community uses one scattering center and no resonance. The parameterization consists of an iteration process involving three steps. First, for each community, the standard GA operations, including selection, crossover and mutation, are utilized to reproduce members in the next generation for better solutions. Second, at the end of each generation, the residual signal of each community is calculated as the error between the best solution in the community and the original data $E^m(f)$. The residual signal is also parameterized with the GA. The order number for the residual parameterization, which is specified in the dashed box, is the difference between the order number of the current community and that of the next higher community. Third, the parameters from the best solution of a lower-order community and its residue are combined to form a candidate solution in the next higher-order community. A zero-mean Gaussian perturbation is added during this step to create a community-level mutation. By adaptively feeding the solutions from the lower-order communities forward to the higher-order communities, the convergence of the highest order community is significantly accelerated without sacrificing the optimality of the final solution.

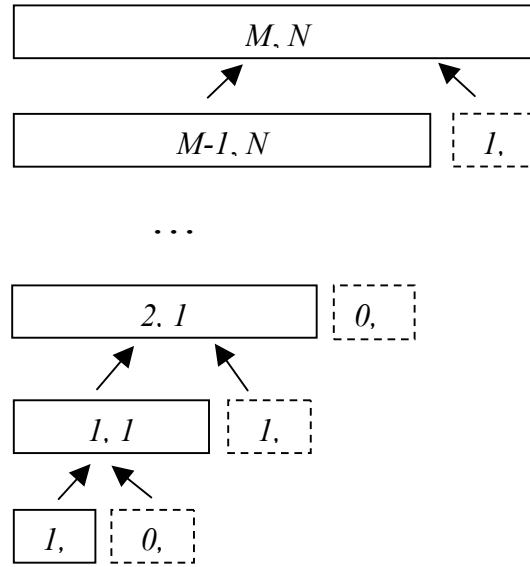


Fig. 3.1 GA with adaptive feeding. The best solution from a lower-order community (solid box) and that from the residue (dashed box) are combined and fed into the next higher-order community. The convergence of the highest-order community with M scattering centers and N resonances is accelerated.

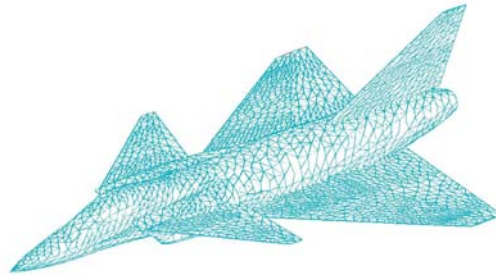


Fig. 3.2 1:30-scale VFY 218 model

3.2.2 Examples

The algorithm is applied to the VFY 218 measurement data [51]. The scattering data came from a 1:30-scale model aircraft (Fig. 3.2) using horizontal polarization in the

8 to 16 GHz frequency band. The look angle was at 19.6 degree from nose-on, so that the inlet contribution is prominent in the return.

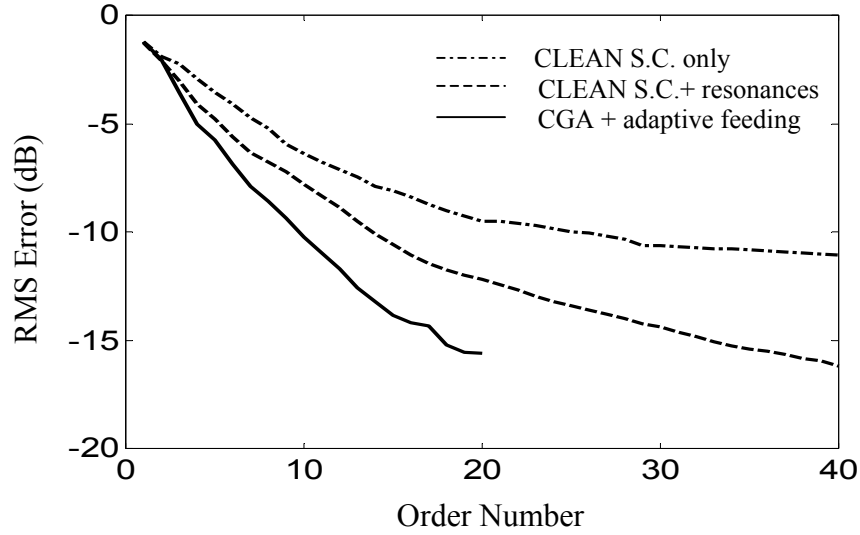


Fig. 3.3 Comparison of three parameterization results of the VFY 218 measurement data

Fig. 3.3 is the plot of RMS error vs. total model order number ($M+N$) for the three different methods tried: 1) CLEAN with scattering center only (dash-dotted line), 2) CLEAN with both scattering center and resonances (dashed line), and 3) the CGA with adaptive feeding (solid line). The CGA results are averaged over six runs. The CLEAN curve with the scattering-center-only model decreases very little after 30 terms, indicating the scattering center model is inefficient in modeling the resonance part of the data after the scattering centers are extracted. The CLEAN curve with both scattering center and resonance model is better; however, the rate of convergence still slows down considerably after the first 18 terms. The CGA with adaptive feeding curve shows the best sparsity. It requires only 20 terms to achieve the same accuracy as the CLEAN

approach with 40 terms. The model orders used in the CGA with adaptive feeding are $M=14$ and $N=6$. It is found, however, that the results are not sensitive to the model order selection.

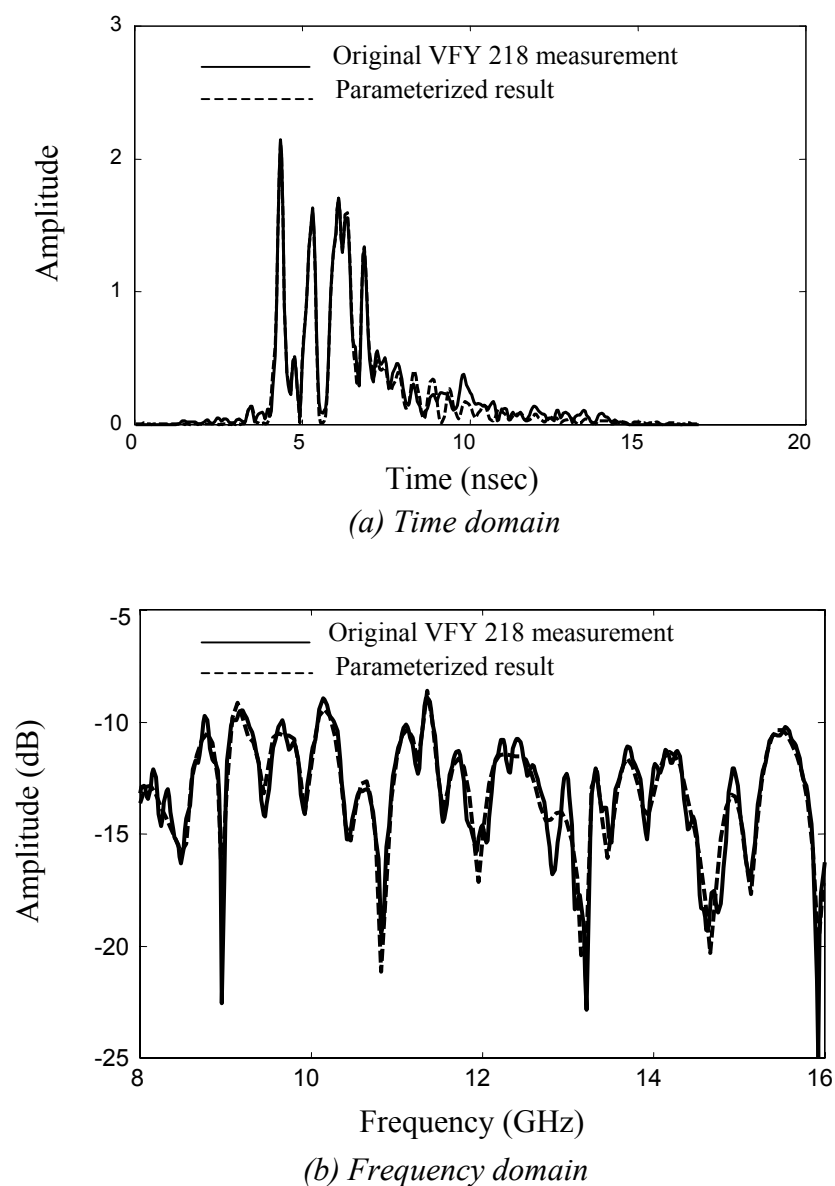


Fig. 3.4 Accuracy of the parameterization in time and frequency domains

To further interpret the physical significance of the GA-parameterized data, the extracted scattering center positions are correlated with the peaks in the target range profile, and they are found to line up well. Furthermore, the two strongest resonances extracted are at frequencies of 9.8 GHz and 11.3 GHz. This is consistent with the size of the rectangular engine inlet openings, which have dimensions of 2.5cm by 1.5cm. (The cutoff frequencies of the TE_{01} and TE_{11} modes are estimated at 10 GHz and 11.7 GHz, respectively.) The other four resonances at 8.6 GHz, 9.1 GHz, 9.4 GHz, and 13.3 GHz, are harder to interpret given the complex shape of the actual inlet structure.

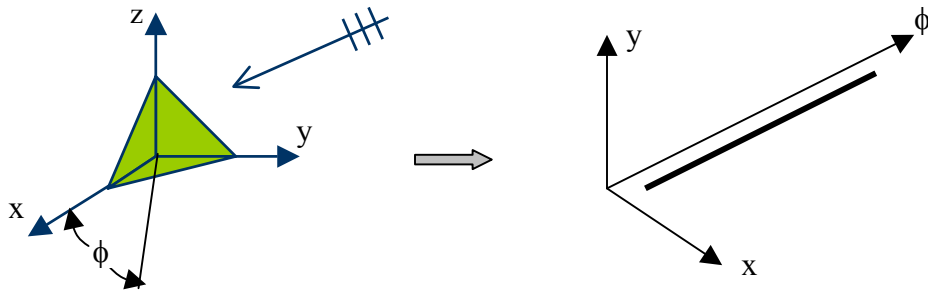
The comparisons of the parameterized result (dashed line) with the original measurement data (solid line) in the time and frequency domains are shown in Fig. 3.4(a) and 3.4(b), respectively. The two results agree fairly well. The small parameterization error is probably due to the model mismatch of Eq. 3.1 to the complex measurement data. Therefore, increasing the model order for this data does not reduce the error significantly. The data from 0 to 180 degrees from nose-on in 5-degree increments are also processed, and the CGA with adaptive feeding consistently outperforms CLEAN at all angles.

To summarize, the testing results indicate that the proposed method can achieve sparser results than other non-global based methods. The resulting sparse model facilitates target feature interpretation and can be used for signature reconstruction in modeling and simulation applications. However, this algorithm processes the data one angle at a time. The correspondence of the scattering features from angle to angle is not considered. In the following section, the visibility of scattering features across aspect angles is investigated.

3.3 Feature Extraction with Visibility

3.3.1 Typical angular behaviors of scattering centers

For complex targets, most scattering features are visible only for a limited aspect span. Moreover, the scattering centers associated with different scattering mechanisms differ in their angular behaviors. Fig. 3.5 shows the different angular behaviors of several canonical scattering structures: a corner reflector, a tophat structure, and a plate. For the convenience of illustration, the 3-D spatial location of a scattering center is plotted only in the XY projection plane. As demonstrated, the angular behaviors of these different scattering mechanisms differ from each other. The angular behavior of the scattering center contributed by the corner reflector is stable. For the tophat structure in Fig. 3.5(b), the scattering center contributed by the circular dihedral moves along its base from angle to angle. Although this feature is not strictly a point scatterer over a wide angle, it should be considered as a single scattering mechanism. As for the specular flash of the plate, it is visible only in a narrow angular window.



(a) Angular behavior of scattering centers for a trihedral

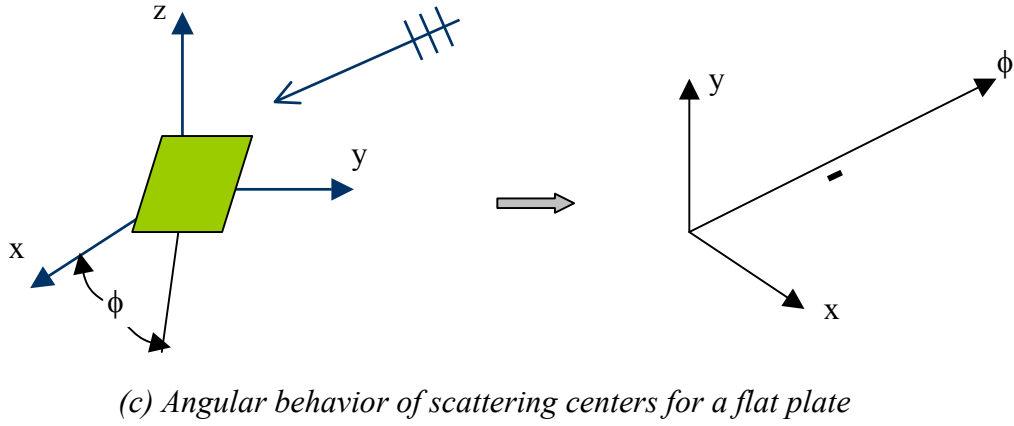
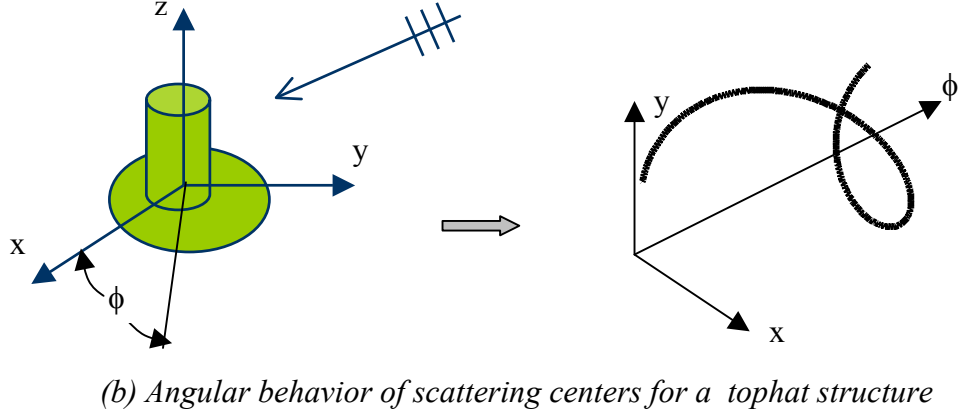
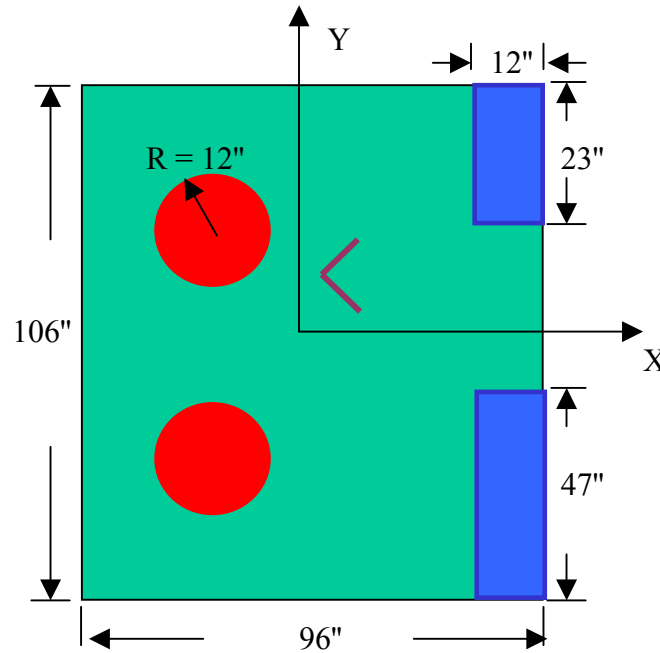


Fig. 3.5 Angular behavior for typical scattering mechanisms

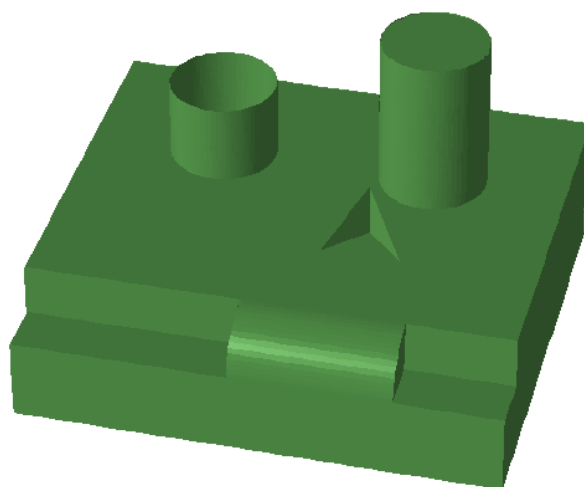
In the work by Bhalla and Ling [20], a 3-D scattering center extraction algorithm was developed. However, the scattering centers are extracted one angle at a time. Therefore, information on the visibility of a scattering center across angles is not identified. Such visibility information would be very important for feature identification. In the following sections, the specific angular behaviors of different scattering mechanisms are first investigated. Algorithms for identifying scattering center visibility are then developed to categorize the scattering centers. In Section 3.3.2, a parameterization algorithm based on the sinogram is described and analyzed. The

algorithm can successfully extract the stable features and features with circular behaviors. An improved segmentation algorithm in multi-dimensional space is then developed in Section 3.3.3. The segmentation algorithm is more general and can successfully extract most features with different angular behaviors.

For convenience of analysis, the benchmark target "Slicy" is taken as a sample target to test the developed algorithms. The Slicy model (Fig. 3.6) consists of two circular cylinders on the top of a square platform. The taller cylinder has a top cover and the other one does not. The Slicy model is relatively smooth compared to realistic targets, but it includes many scattering mechanisms, such as specular reflections from flat surfaces, retro-reflection from trihedral and dihedral corner reflectors, and high-order multiple scattering from the open cavity.



(a) Size of the Slicy model



(b) *View of Slicy 3-D model*

Fig. 3.6 *Geometry of the Slicy model*

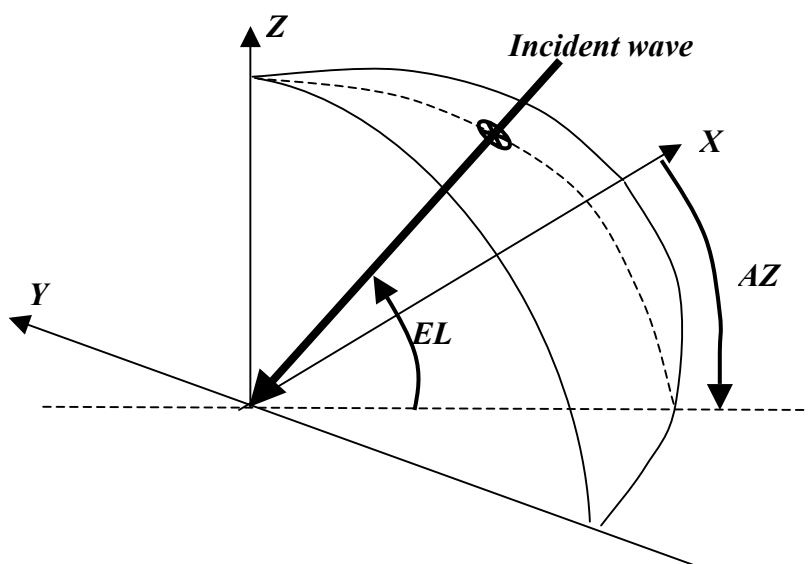


Fig. 3.7 *Definition of azimuth and elevation angles*

The raw scattering data are generated by an SBR simulator [14, 20]. The data are collected by setting the azimuth angle from 0° to 360° and the elevation angle to 30° , as defined in Fig. 3.7. The frequency band is from 4.5 GHz to 14.5GHz, with a bandwidth of 10 GHz at the central frequency of 9.5GHz. The 3-D scattering centers are then extracted from angle to angle across all 360 azimuth angles by a CLEAN extraction algorithm with a dynamic range of 40 dB. The number of bounces and the facets that contribute to the specific scattering center are also recorded during the CLEAN extraction based on the ray history.

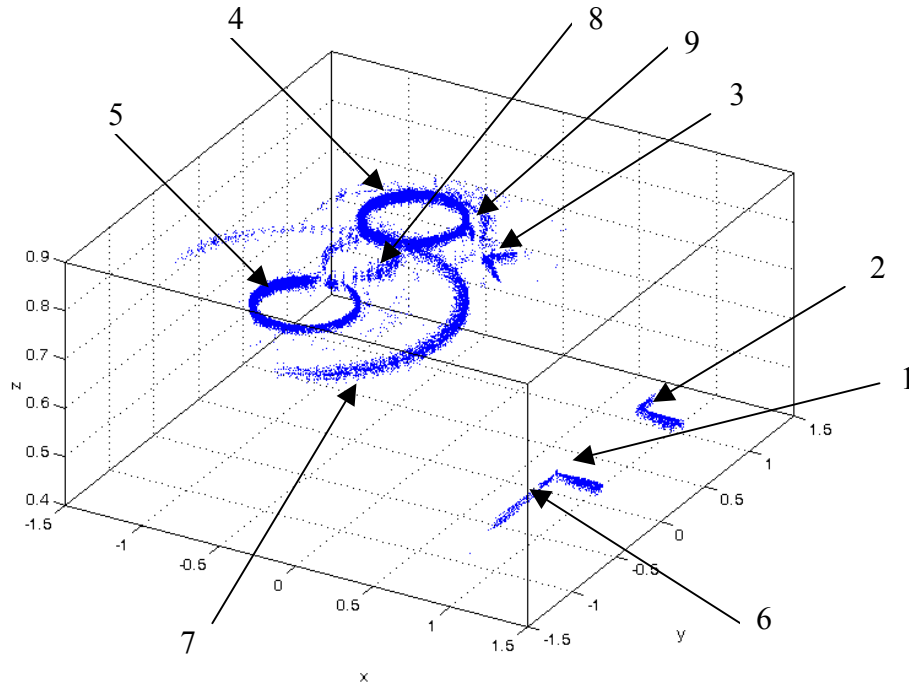


Fig. 3.8 3-D spatial location of scattering centers for the Slicy model

The total number of scattering centers resulting from the above process is very large (more than 38,000), which makes the direct analysis of the scattering features difficult. A 3-D spatial location display of all the extracted scattering centers is shown in

Fig. 3.8. A number of scattering mechanisms can clearly be observed, including point features from the trihedrals (1, 2, and 3), circular features from the tophats (4 and 5), line features from the edge flash (6), and higher-order features from multiple scattering (7, 8, and 9). However, an automated algorithm needs to be developed to separate the different mechanisms.

3.3.2 Feature separation based on the sinogram

Scattering feature separation and parameterization is first attempted in the sinogram domain by using a parametric model to extract features iteratively. A sinogram is the range profiles of a target collected over many aspect angles. It is generated by first creating the scattered field data in frequency for each angle and then

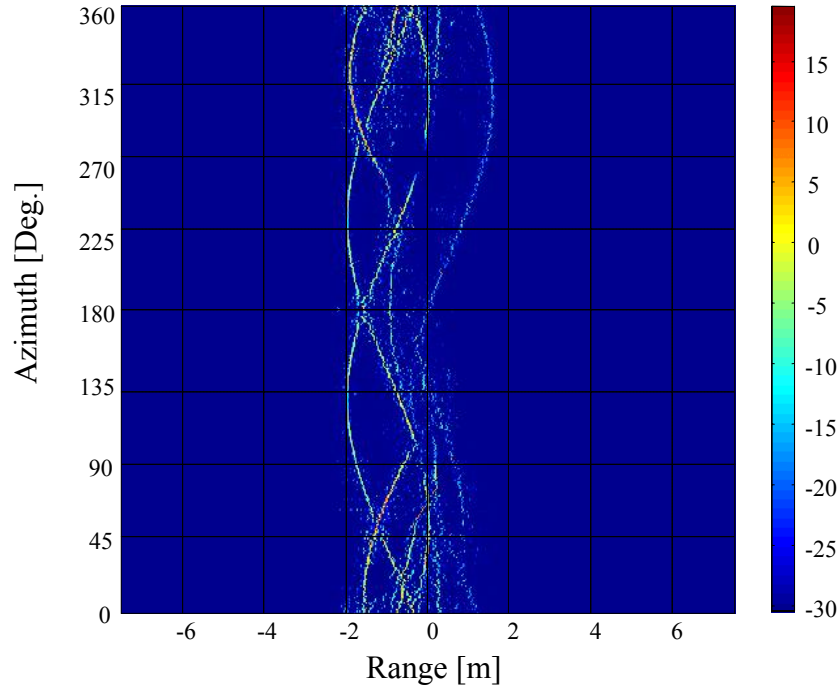


Fig. 3.9 Sinogram for the Slicy model

taking the inverse Fourier transform to obtain the range profile. The range profile can be considered as a projection of the 3-D scattering centers onto the radar line-of-sight (LOS). As an example, the sinogram for the Slicy model is shown in Fig. 3.9. The color represents the strength of the range return in dB scale. A number of range tracks with different angular behaviors can be observed.

The separation and parameterization of the range tracks in the sinogram is next tried. The following model is used:

$$R(\phi) = r_0 \cos(\phi + \phi_l) + r_l \quad (\text{Eq. 3.4})$$

where ϕ represents the azimuth angle and (r_0, ϕ_l, r_l) are the parameters to be extracted. Basically, the model described by Eq. 3.4 assumes that each scattering center has a circular behavior with the azimuth angle in 3-D space. When projected from the 3-D space to the range domain, the trajectory of the scattering center described by Eq. 3.4 is a sinusoidal curve, as shown in Fig. 3.10 for four different scattering centers.

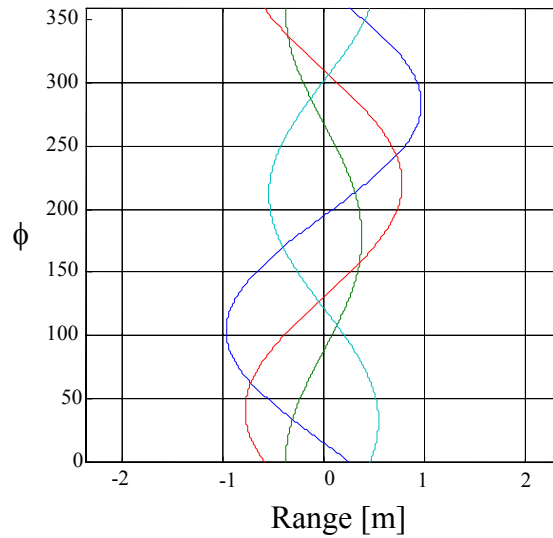


Fig. 3.10 Trajectories of different scattering centers in the sinogram

For each scattering feature, its parameters (r_0, ϕ_l, r_l) are searched by finding the maximum projection of the sinogram to the basis function $R(\phi)$. Here the projection value is calculated by simply accumulating the absolute value of the strength along the trajectory $R(\phi)$ in the sinogram. Once the parameters (r_0, ϕ_l, r_l) are obtained, the inverse process of finding the trajectory of the 3-D location $(x(\phi), y(\phi), z(\phi))$ can be obtained as follows:

$$\begin{aligned} x(\phi) &= \frac{r_0}{\sin \theta} \cos \phi_l + R_l \cos \phi \\ y(\phi) &= -\frac{r_0}{\sin \theta} \sin \phi_l + R_l \sin \phi \\ z(\phi) &= (r_l - R_l \sin \theta) / \cos \theta \end{aligned} \quad (\text{Eq. 3.5})$$

where θ is the elevation angle and R_l is an unknown. Here the unknown R_l comes from the loss of information during the projection from 3-D space into the 1-D range direction. It can be decided by a second search to match the maximum number of scattering centers along $(x(\phi), y(\phi), z(\phi))$. The scattering centers along the trajectory $(x(\phi), y(\phi), z(\phi))$ are then extracted from the sinogram and categorized as a scattering feature. The process is then iterated to extract and separate the other features.

The detailed procedure of the algorithm is described by the flow chart in Fig. 3.11. As indicated in the flow chart, the algorithm identifies the movers and non-movers according to the variance of the tracked scattering centers. Here a non-mover represents the stable scattering feature whose location does not change appreciably with the aspect angle. A mover represents the scattering feature whose location changes with the angle.

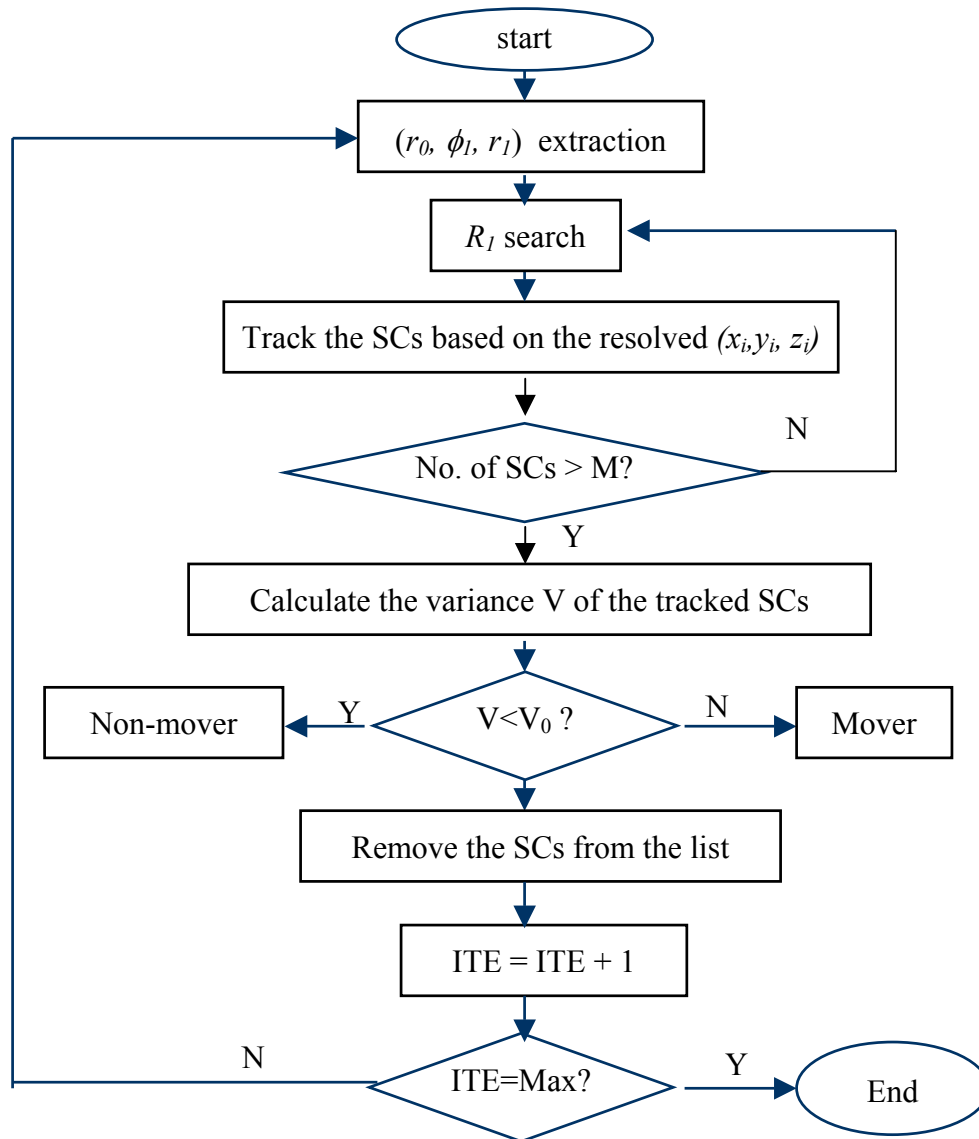


Fig. 3.11 Flow chart of the sinogram-based algorithm

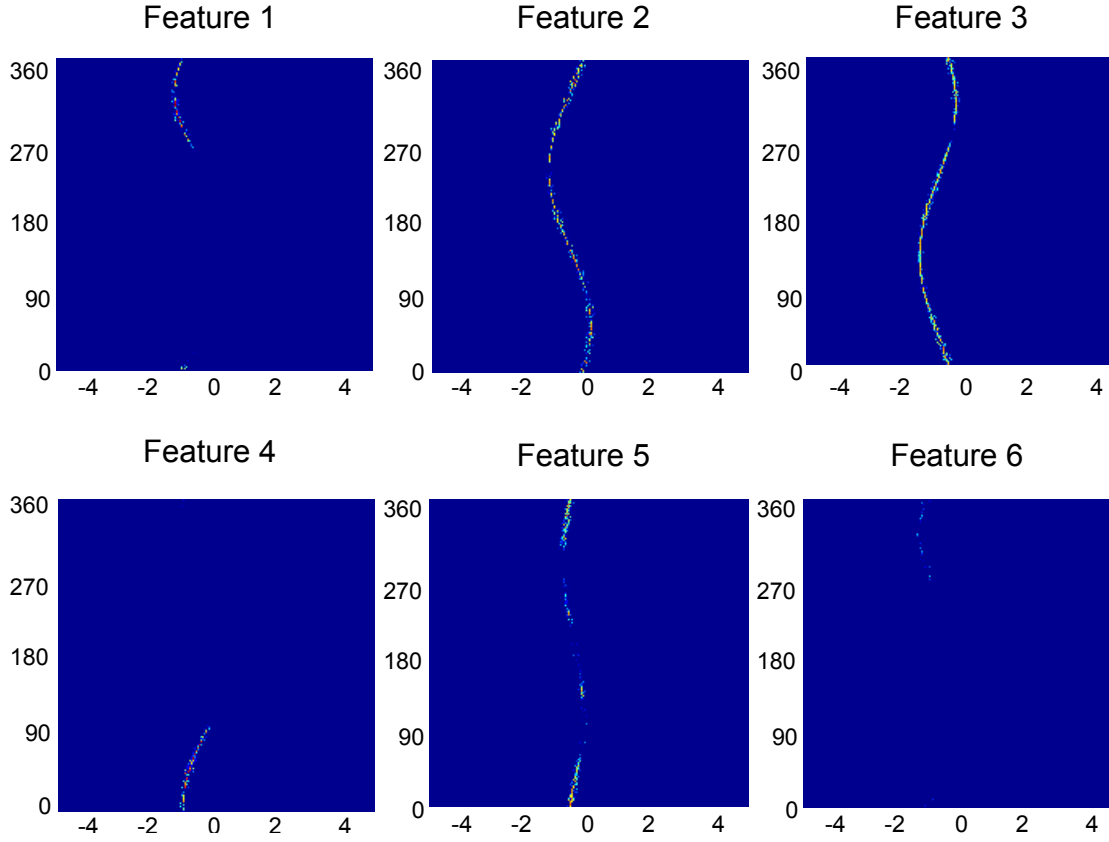


Fig. 3.12 Scattering feature extraction by sinogram-based algorithm for the Slicy

The algorithm is tested on numerical simulation data from the Slicy model shown in Fig. 3.6. Fig. 3.12 shows the first six features extracted by the algorithm for the Slicy model. Compared to the original sinogram in Fig. 3.9, it can be observed that all the main features are successfully extracted. Fig. 3.13 shows the residual sinogram. The residual scattering features include the edge flash, multiple scattering from inside the cavity, and other weak features. The parameterized scattering features are displayed in Fig. 3.14, where the two tophat structures and three corner reflectors are clearly depicted.

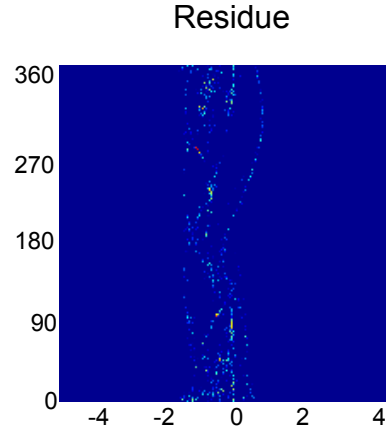


Fig. 3.13 Residue sinogram after seven iterations of scattering feature extraction by sinogram-based algorithm for the Slicy model

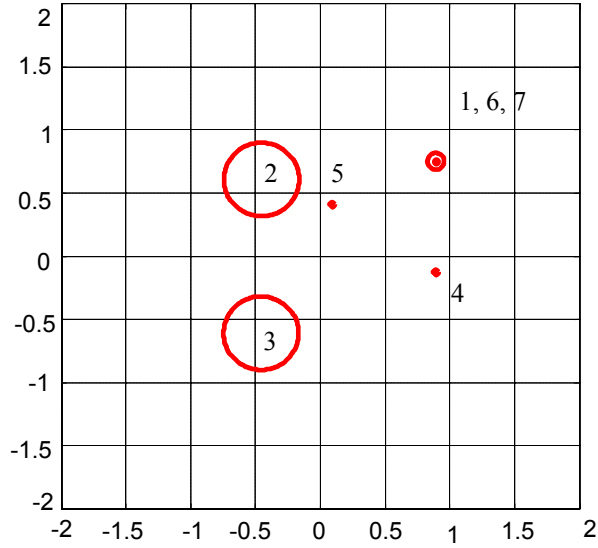


Fig. 3.14 Parameterized scattering features extracted for the Slicy model

Table 3.1 lists the extracted parameters for the first seven iterations. Besides the feature parameters, the percentage energy for each extracted feature is listed in the last column. In calculating this number, the total energy is calculated from the incoherent sum of the returns from all of the scattering centers over all angles. The energy for each feature is calculated from only those scattering centers corresponding to the feature. For

the seven extracted features, the sum of the energy is 88.23%. It should be noted that the residue contains some edge flash features, which are strong but only visible in a very narrow angular window. Therefore, the near 90% energy for the extracted features indicates a good feature extraction for the Slicy model.

Table 3.1 Extracted parameters of the first seven features for the Slicy model

No.	r_0	r_1	$\phi_1(\text{deg.})$	R_l	Energy%
1	1.0103	0.2436	320.1803	0.02	23.89
2	0.6574	0.6459	233.2750	0.29	20.06
3	0.6591	0.6447	127.0300	0.29	12.17
4	0.7804	0.2565	8.1173	0.02	23.28
5	0.3426	0.4112	285.2530	0.02	8.77
6	0.9997	0.2522	320.1726	0.09	0.01
7	1.0001	0.2510	320.0716	0.00	0.05

The model in Eq. 3.4 is not general in that it can only describe the stable scattering centers (non-movers) and features with circular tracks. Therefore, while this approach worked adequately for the test target Slicy, it would have trouble with more general targets, which may include some non-circular features. In the following section, a more general algorithm is presented to separate and categorize the scattering features with more general angular behaviors.

3.3.3 Segmentation algorithm in multi-dimensional space

The spatial separability of scattering centers is next investigated in 3-D target-centered space. While the features shown in Fig. 3.8 for Slicy are quite revealing, their spatial locations alone do not offer sufficient separability to allow for a successful extraction of these mechanisms. The dimensionality of the feature space needs to be increased to enhance mechanism separability. To accomplish this, the feature space is extended to higher dimensions by utilizing the bounce numbers and angular persistence of the scattering centers. The goal is to move to a sufficiently high-dimensional feature space such that the scattering mechanisms become naturally separated and can thus be extracted more easily.

A multi-dimensional segmentation algorithm is next developed to implement the tracking and separation of scattering mechanisms. The algorithm starts at the discretization of the multi-dimensional space, including 3-D location (x, y, z) , ray bounce number, and angular persistence. For the three dimensions corresponding to the location of scattering centers, the grid size is decided by the range resolution. Investigation indicates that twice the resolution cell is a good choice. To conserve memory and increase computational speed, the scattering center data are first processed and separated into different groups by ray bounce number. The 5-dimensional space is then reduced to 4-D after the process, namely, 3-D space (x, y, z) and angular persistence. A segmentation algorithm is then applied to automatically separate the different mechanisms. The objective of the presented segmentation algorithm is to cluster a collection of points (scattering centers) in multi-dimensional space into different groups.

The procedure of proposed multi-dimensional segmentation algorithm can be described as follows:

1. *Discretize the 4-D feature space into small cells.*
2. *Set up an empty search list and an empty feature list.*
3. *Randomly select a non-zero cell (Cell 1); put it into the feature list and set it to zero.*
4. *Find Cell 1's non-zero neighbor cells; put them into the feature list as well as the search list and then set them to be zeros.*
5. *Find Cell 2 in sequence from the search list (it has been set to zero in Step 3); find its non-zero neighbor cells; put them into both the feature list and search list; set them to zeros and then delete Cell 2 from the search list;*
6. *Find the next cell in sequence from the search list. Repeat Step 4 until the search list is empty. The first feature is finished.*
7. *Repeat Steps 2-5 to find the next feature. Eventually, all the cells in the space will be zeros and the algorithm exits.*

Basically, this algorithm uses the continuity of the cells in the discretized space to segment the scattering centers into groups. It can be considered as a discretized version of the K-means clustering algorithm [52]. Because it is difficult to display a 4-D space, an example procedure of the algorithm is demonstrated in Fig. 3.15 for the 2-D case. The assumed 2-D space is first discretized into small cells, as shown in Fig. 3.15(b). The segmentation algorithm is then applied to separate the discretized space into different groups, as shown in Fig. 3.15(c). The result is a feature list, which includes the number of

features and the grid locations corresponding to each feature (cluster). The separation of different features is then completed by finding the corresponding scattering centers for each cluster based on the obtained grid-location information.

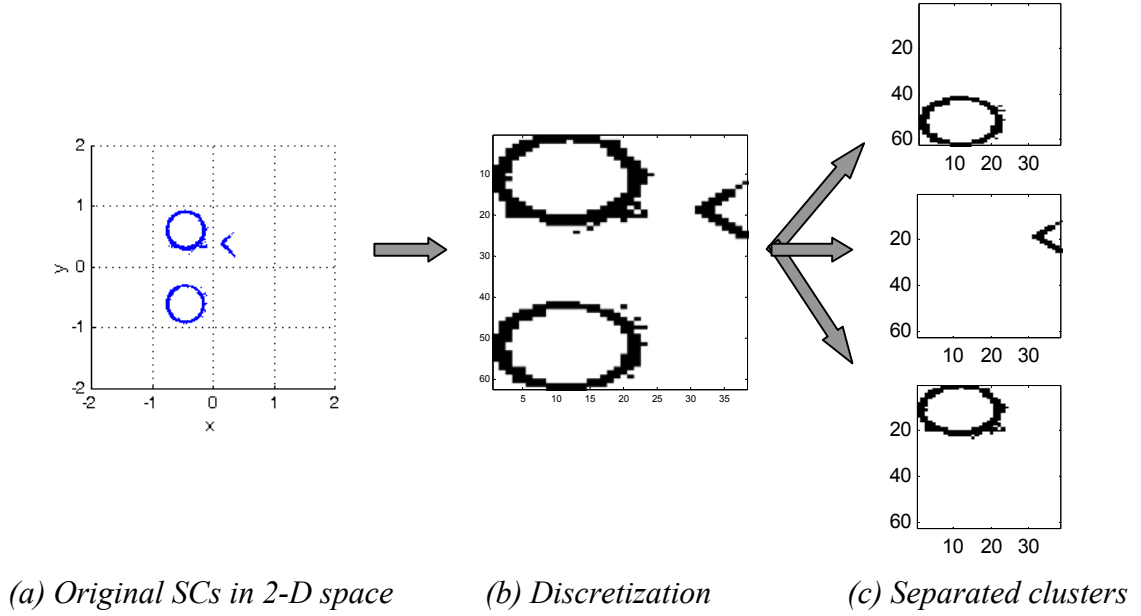


Fig. 3.15 Example procedure of the segmentation algorithm for the 2-D case

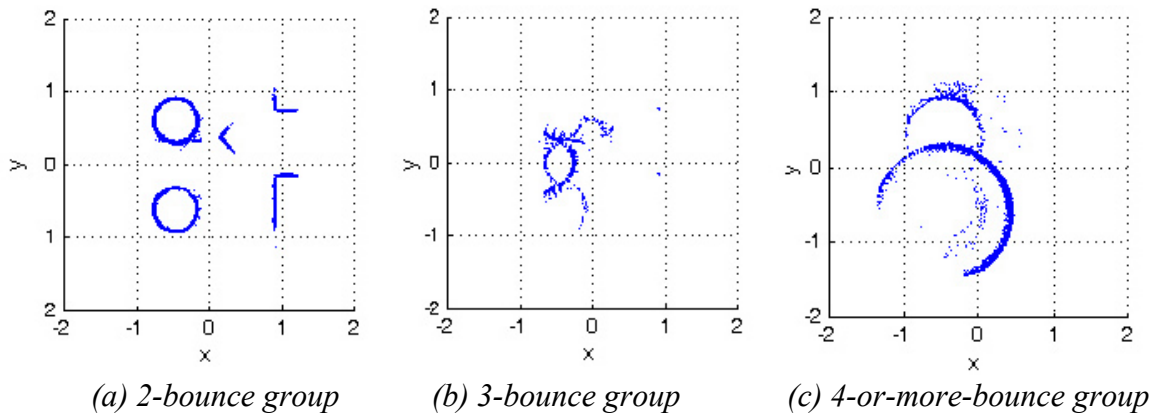


Fig. 3.16 Projection of scattering centers on the XY plane after separation by bounce number for the Slicy data

The algorithm is tested using the data from the Slicy model. For the Slicy model, the original scattering centers are first separated by bounces, as shown in Fig. 3.16. The algorithm is then applied on each group to separate the different features.

Table 3.2 lists the first 21 features and the corresponding number of scattering centers, as well as the percentage energy for each feature. The x-span, y-span, z-span, and angle-span for each feature are also indicated. Here the x-, y-, and z-spans represent the extent of the feature in the x, y, and z directions, respectively. The angle-span means the angular window (in degrees) where the feature is visible. While the number of bounces indicates the scattering order of the feature, the x-, y-, and z-spans indicate the spatial extent of the feature. A feature like No. 1, which has small x-, y-, and z-spans, indicates that the scattering centers are focused around a point. Moreover, the 80° angle-span, the number of bounces (3), and the large energy content further substantiate that the feature corresponds to a trihedral corner reflector. Similar characteristics can be found for feature No. 3 and No. 5, which are also due to trihedral corner reflectors. On the other hand, feature No. 2 and No. 4 have large x- and y-spans and angle-spans close to 360° , implying that these features are visible almost from all directions. Also, they are two-bounce features. Indeed they correspond to the returns from the circular tophat structures formed by the circular cylinders and the flat top surface. The analysis can be applied for other features to relate each of them to the physical location on the target. The locations of several important features for the Slicy model are shown in Fig. 3.17.

Table 3.2 Scattering feature separation by multi-dimensional segmentation algorithm for the Slicy model

Feature No.	No. of bounce	x-span (m)	y-span (m)	z-span (m)	Angle span (°)	No. of SCs	Energy %
1	3	0.0126	0.0209	0.0162	80.0	132	24.27
2	2	0.6752	0.6710	0.0506	356.0	15671	24.25
3	3	0.0115	0.0166	0.0203	80.0	130	22.24
4	2	0.6306	0.6365	0.0466	348.0	11903	13.80
5	3	0.0294	0.0165	0.0206	73.0	143	6.49
6	2	0.2955	0.0368	0.0292	32.0	890	1.48
7	2	0.2939	0.0311	0.0275	37.0	845	1.45
8	2	0.2028	0.3808	0.0170	80.0	366	1.18
9	2	0.0010	0.0016	0.0032	1.0	4	0.61
10	2	0.0029	0.0032	0.0020	2.0	10	0.57
11	2	0.0254	0.6214	0.0197	16.0	147	0.47
12	2	0.0325	0.4606	0.0189	5.0	151	0.47
13	3	0.1825	0.5955	0.0391	89.0	291	0.46
14	3	0.4558	1.0082	0.0557	51.0	1344	0.40
15	3	0.0691	0.0777	0.0404	30.0	220	0.39
16	4	1.8028	1.7706	0.0461	223.0	3975	0.16
17	2	0.0000	0.0004	0.0003	1.0	4	0.12
18	4	1.0337	0.6087	0.0444	104.0	577	0.11
19	2	0.0912	0.0944	0.0230	15.0	101	0.10
20	2	0.0312	0.2799	0.0203	17.0	209	0.06
21	2	0.2155	0.2180	0.0261	46.0	263	0.02

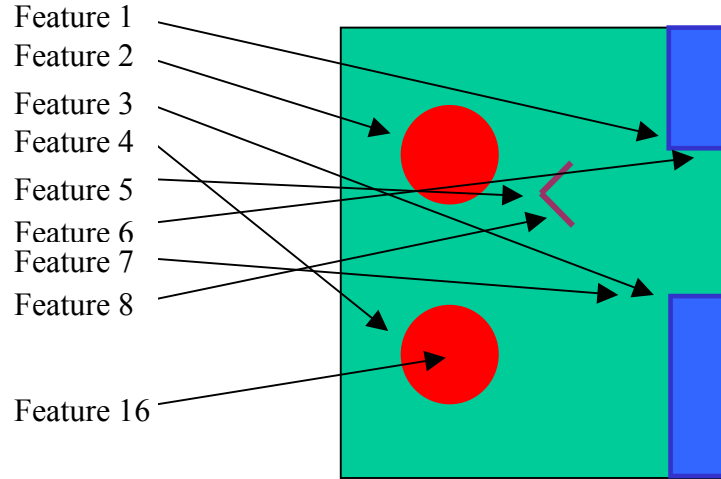


Fig. 3.17 Locations of typical scattering features for the Slicy model

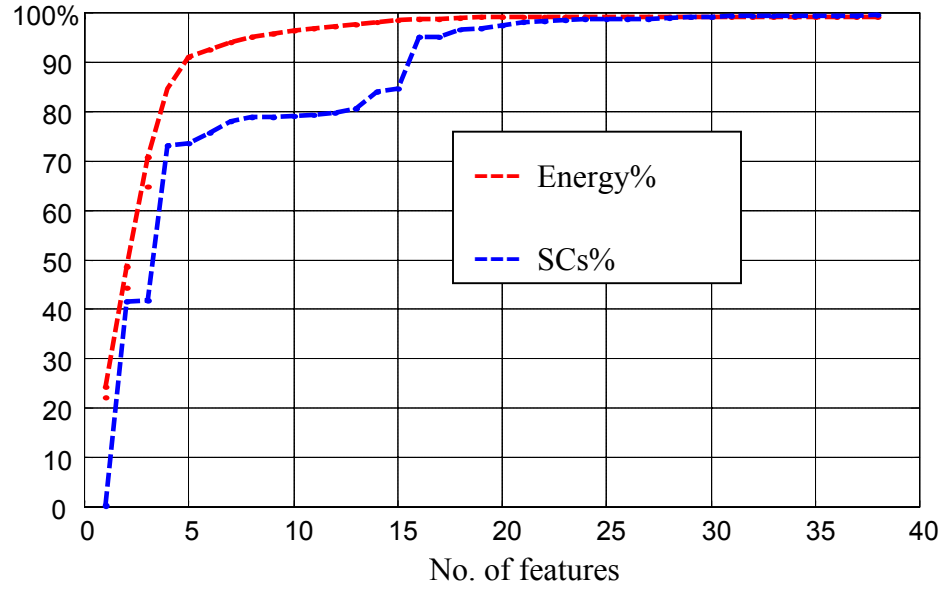


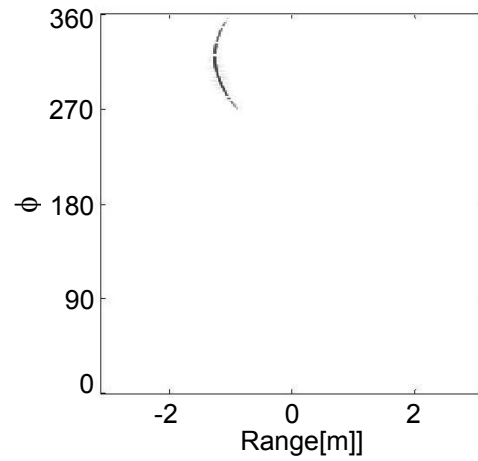
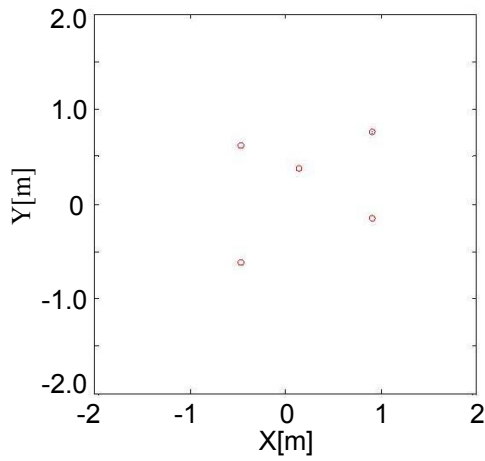
Fig. 3.18 Percentage of scattering centers and energy for scattering features extracted for the Slicy model

Fig. 3.18 shows the percentage of scattering centers and the percentage energy for the extracted scattering features from the Slicy model. The first 20 features include about 97% of the scattering centers and 99% of the total energy, indicating that the multi-dimensional segmentation algorithm yielded a very successful separation of the scattering

features. The first 20 features are shown from Fig. 3.19 to Fig. 3.38. For each figure, the upper left subplot shows the locations of the scattering centers for each feature in blue. The upper right subplot shows the corresponding sinogram for the separated feature. The bounce number, energy percentage, and the number of scattering centers are displayed at the bottom left. Finally, in the bottom right subplot, those facets with contribution to the separated feature are marked by color. While Table 3.2 gives an estimate to each feature based on some characteristic numbers, these more detailed figures provide a clearer understanding to the extracted features. In Fig. 3.19, the locations of the scattering centers and the contributing facets indicate Feature 1 is related to the corner reflector on the right step. The sinogram shows that the feature is visible from around 270° to 360° . In Fig. 3.20, Feature 2, which is due to the right tophat structure, is displayed. The color facets in the bottom right subplot indicate that the scattering is contributed by the double reflection between the right cylinder and the top of the base. The sinogram shows that the feature is almost visible in all the directions except for the blockage by the left cylinder around 90° . The next two features are similar to the first two discussed: Feature 3 in Fig. 3.21 is due to the corner reflector on the left step, and Feature 4 in Fig. 3.22 is due to the left tophat structure. Fig. 3.23 shows Feature 5, which is due to the corner reflector on the top base. The sinogram indicates that the scattering from the reflector is visible from the front side of the target from -45° to $+45^\circ$. Among the weaker features, the most interesting one to point out is Feature 16 in Fig. 3.34. It is due to the left open cylinder. Since the structure is a cavity, the number of bounces is high (4). However, the location of the return moves in a very predictable manner in 3-D space and the return is visible

over a wide range of angles. The feature is blocked due to the taller right cylinder around 270° . These results show that all the features shown in Fig. 3.8 can be satisfactorily separated and identified on the model.

The testing results for the multi-dimensional segmentation algorithm indicate it is a very effective method to extract and separate the scattering features from the scattering data. Compared to the sinogram algorithm, this algorithm is more general and can extract features with very different angular behaviors. The algorithm has also applied to a more complex ground vehicle with satisfactory results.



No. of bounces = 3
 Energy = 24.27%
 Cumulative energy = 24.27%
 No. of SCs = 132

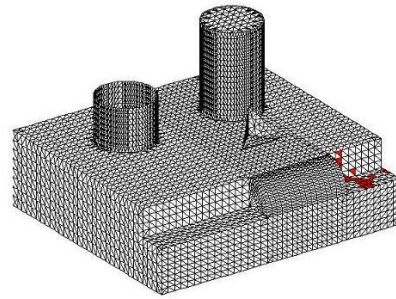
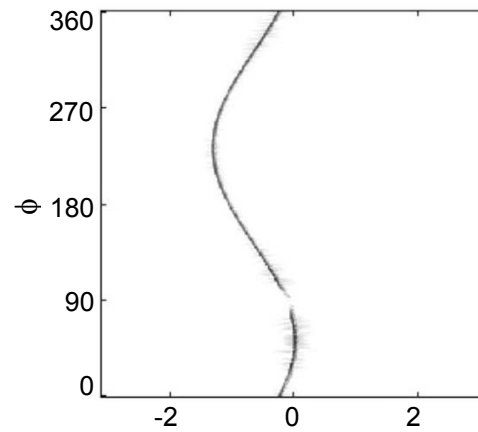
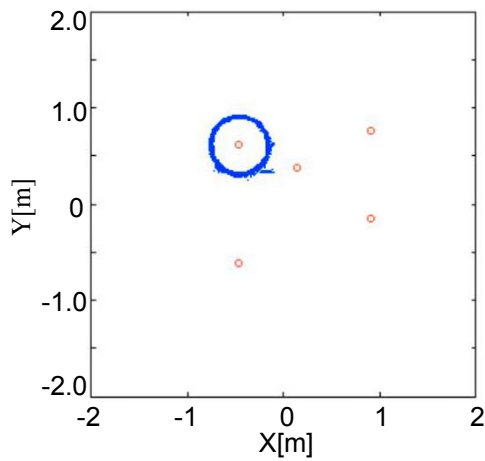


Fig. 3.19 Feature 1



No. of bounces = 2
 Energy = 24.25%
 Cumulative energy = 48.52%
 No. of SCs = 15671

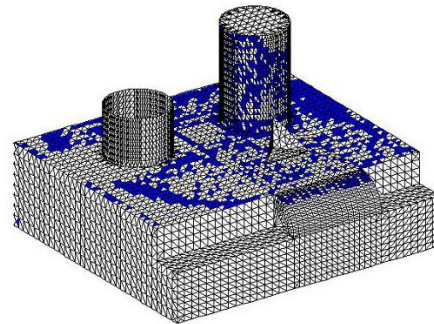
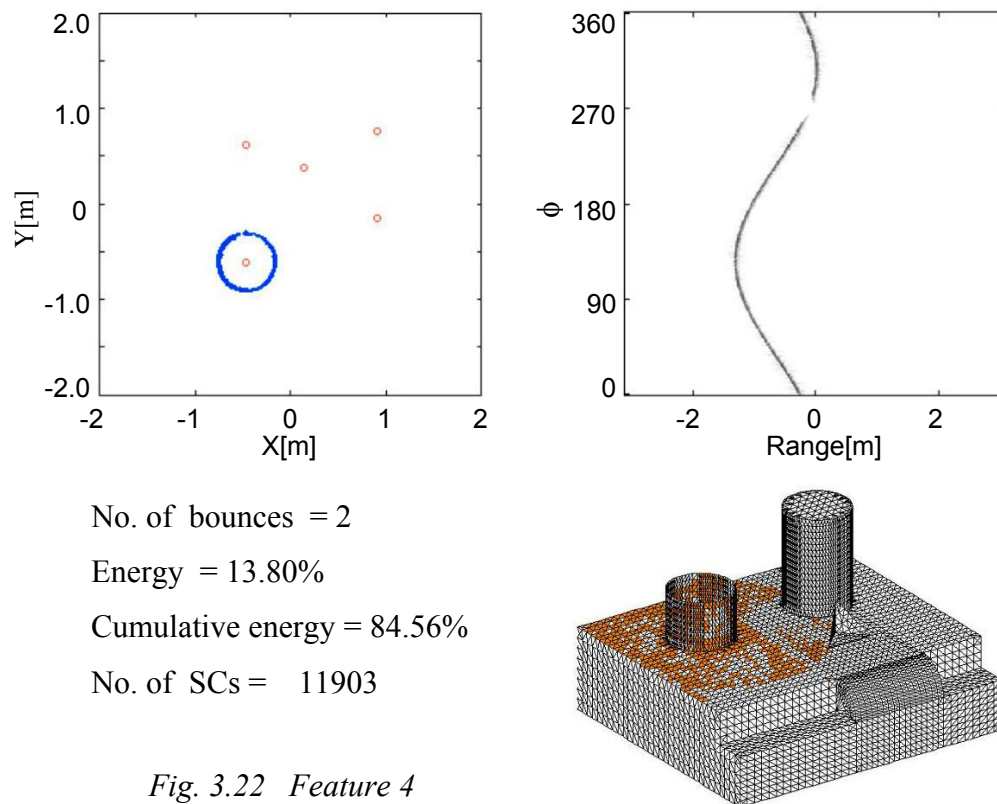
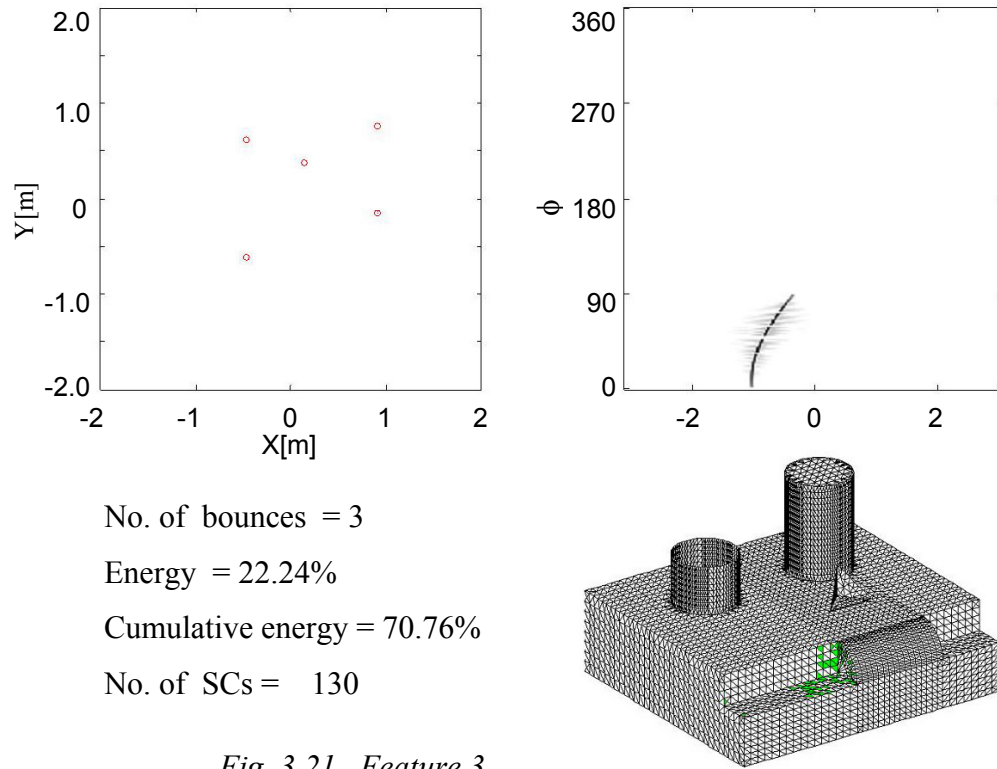
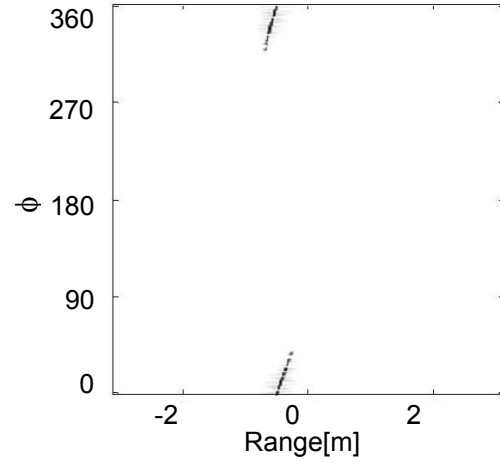
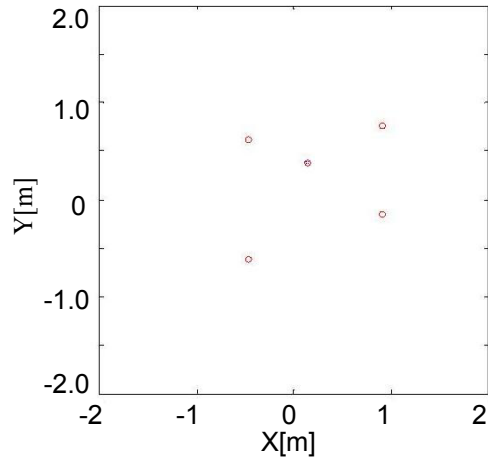


Fig. 3.20 Feature 2





No. of bounces = 3
 Energy = 6.49%
 Cumulative energy = 91.05%
 No. of SCs = 143

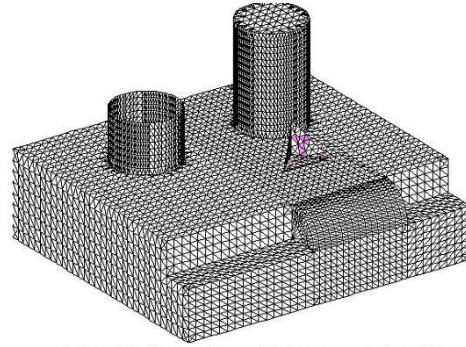
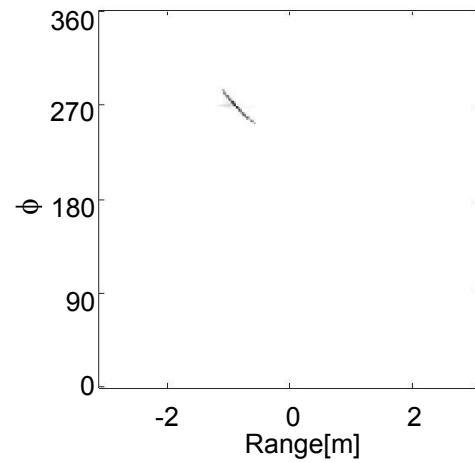
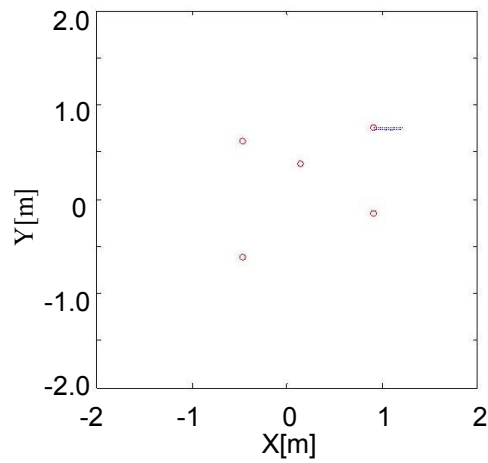


Fig. 3.23 Feature 5



No. of bounce = 2
 Energy = 1.48%
 Cumulative energy = 92.53%
 No. of SCs = 890

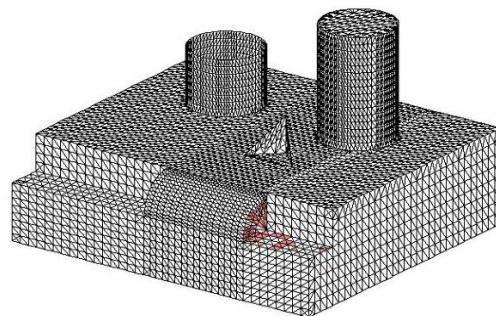
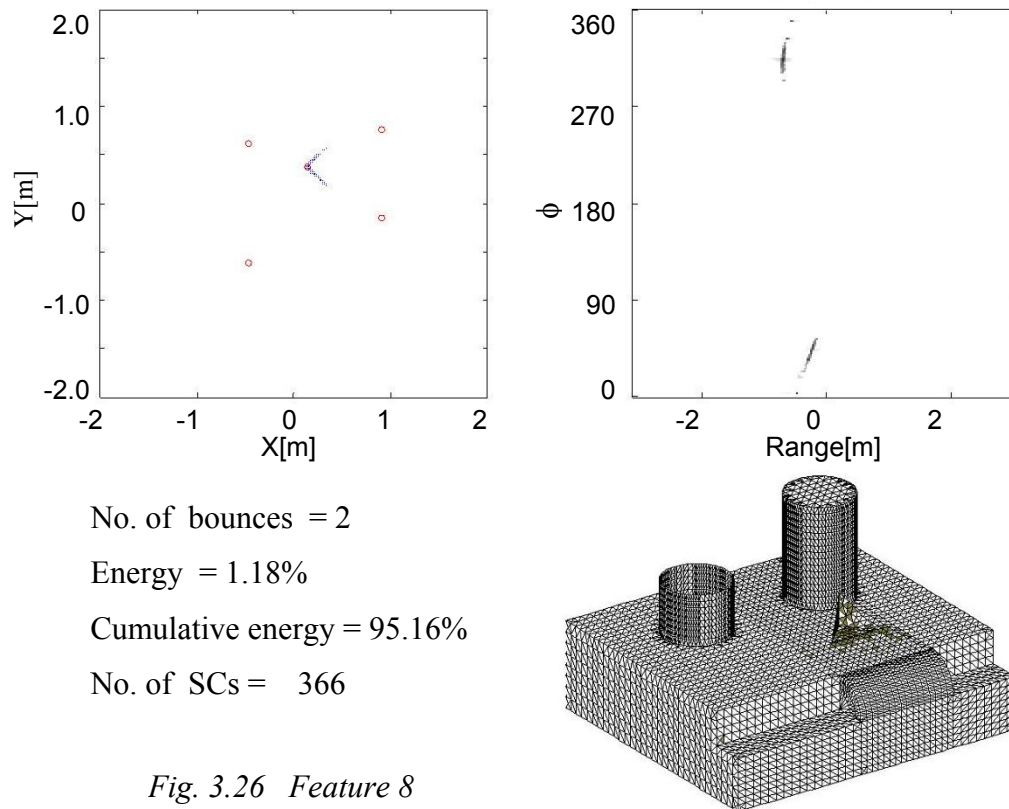
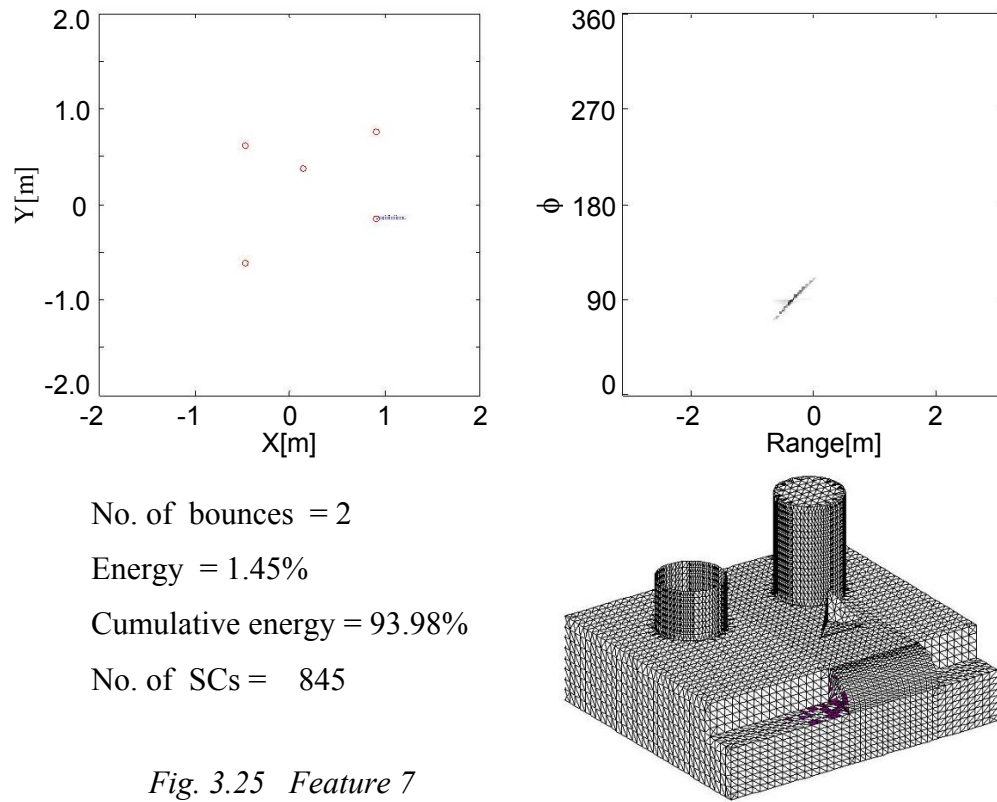
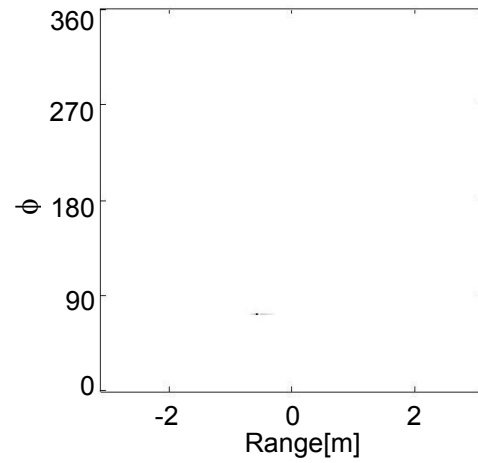
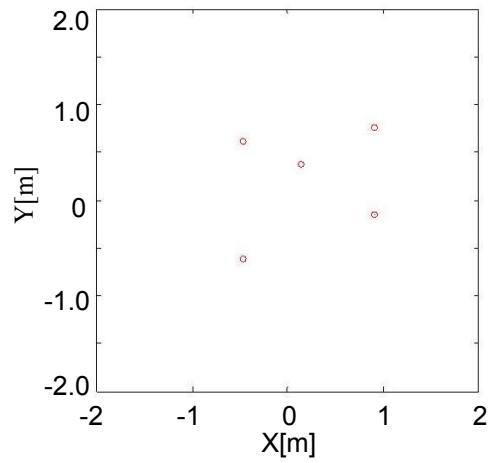


Fig. 3.24 Feature 6





No. of bounces = 2

Energy = 0.61%

Cumulative energy = 95.77%

No. of SCs = 4

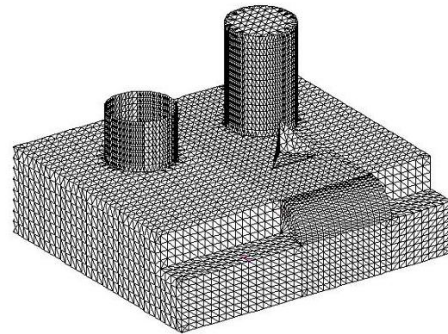
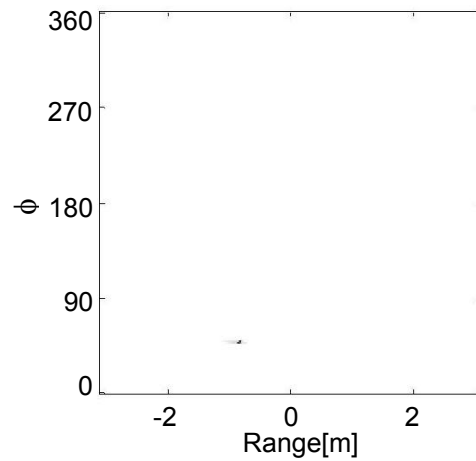
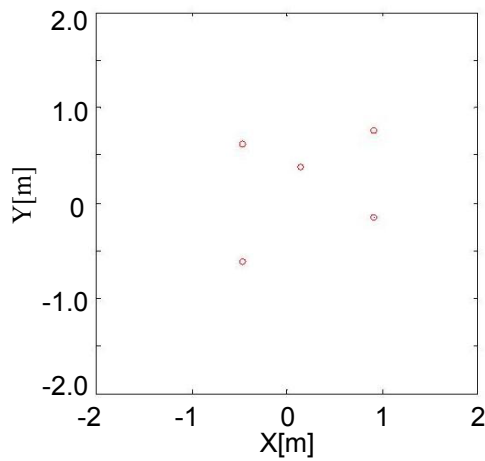


Fig. 3.27 Feature 9



No. of bounces = 2

Energy = 0.57%

Cumulative energy = 96.34%

No. of SCs = 10

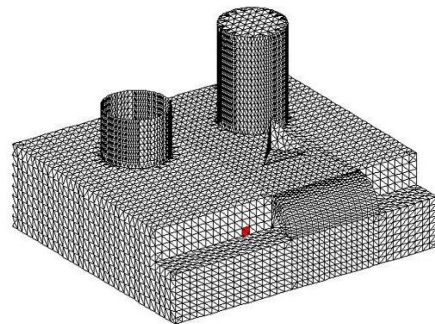
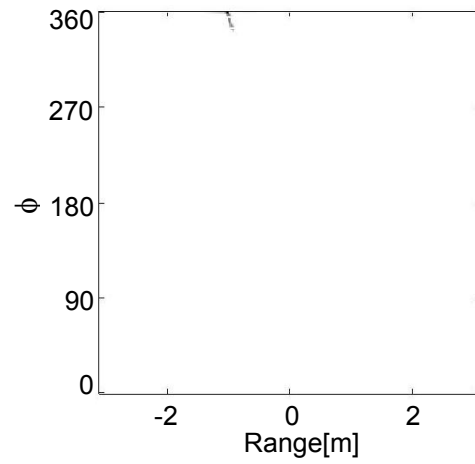
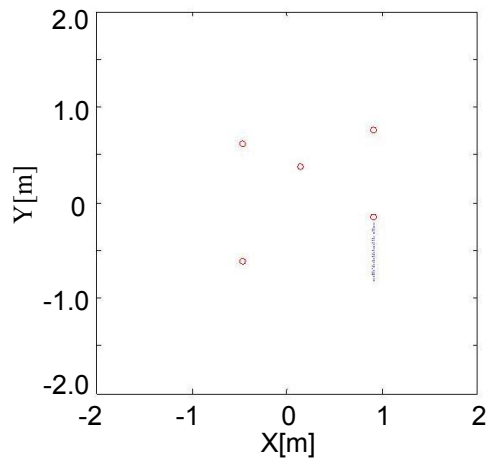


Fig. 3.28 Feature 10



No. of bounces = 2
 Energy = 0.47%
 Cumulative energy = 96.81%
 No. of SCs = 147

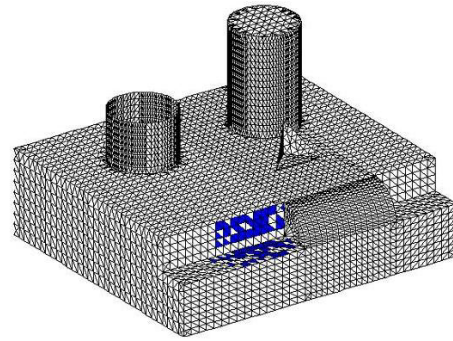
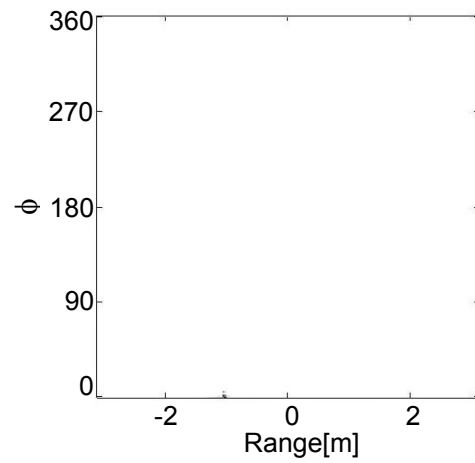
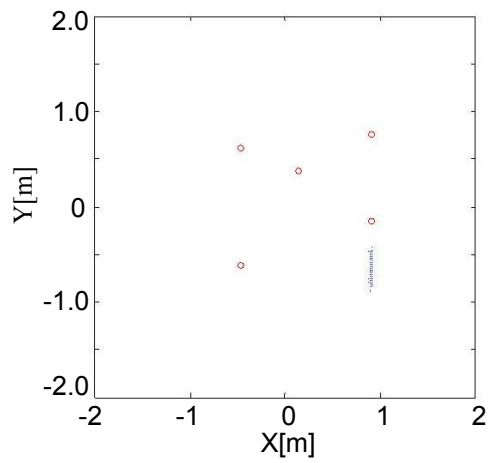


Fig. 3.29 Feature 11



No. of bounces = 2
 Energy = 0.47%
 Cumulative energy = 97.28%
 No. of SCs = 151

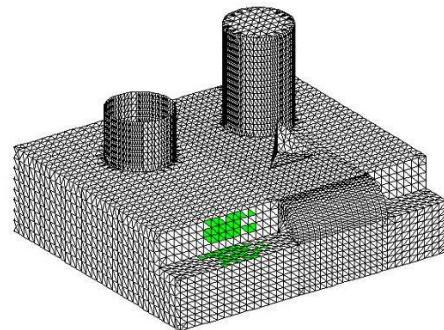
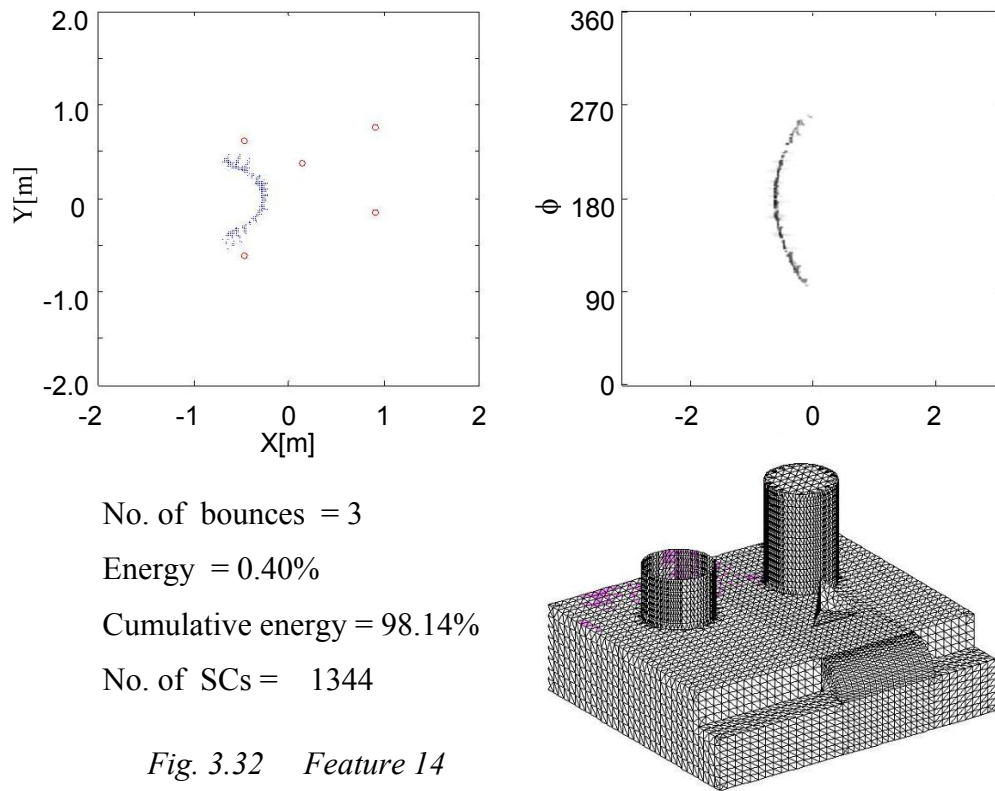
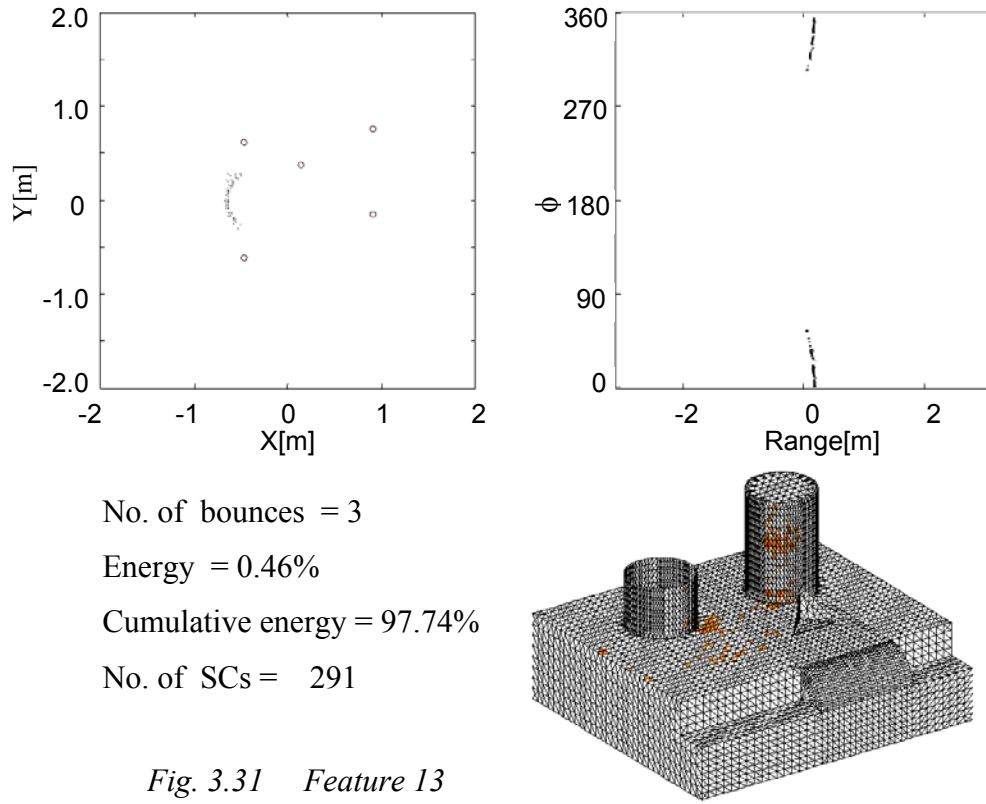
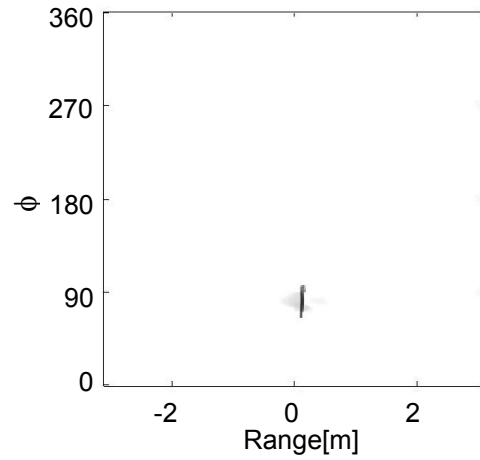
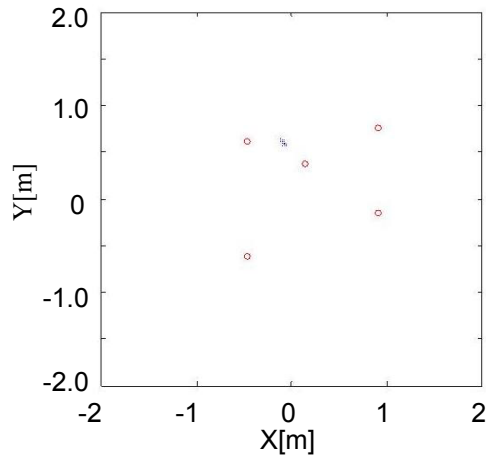


Fig. 3.30 Feature 12





No. of bounces = 3
 Energy = 0.39%
 Cumulative energy = 98.53%
 No. of SCs = 220

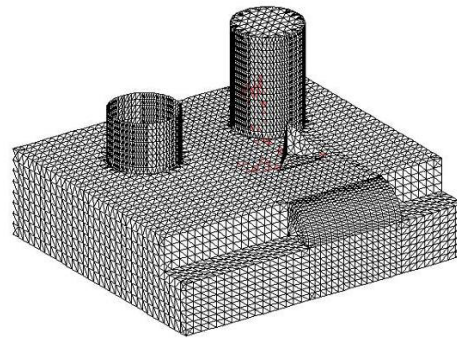
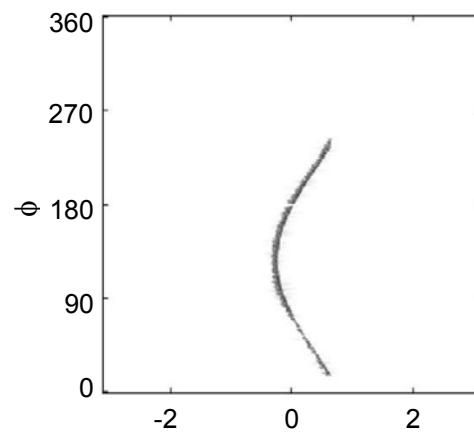
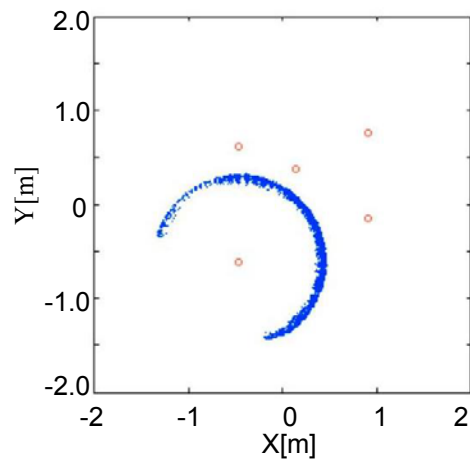


Fig. 3.33 Feature 15



No. of bounces = 4
 Energy = 0.16%
 Cumulative energy = 98.69%
 No. of SCs = 3975

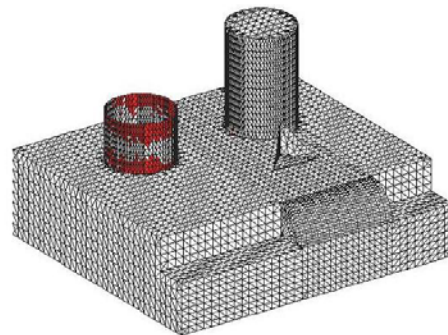
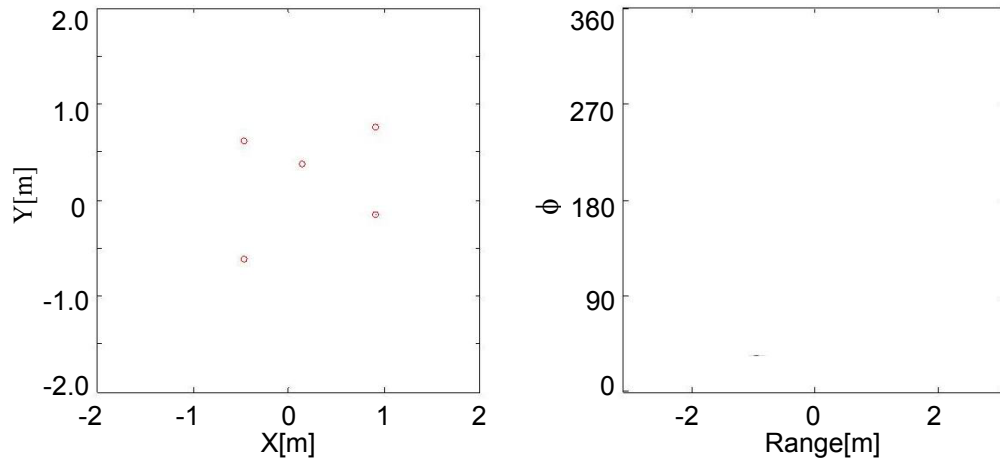
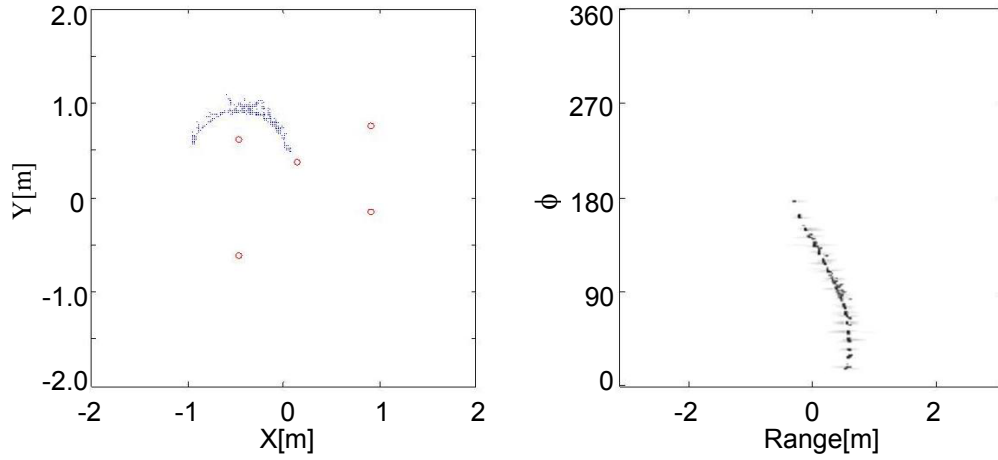


Fig. 3.34 Feature 16



No. of bounces = 2
 Energy = 0.12%
 Cumulative energy = 98.81%
 No. of SCs = 4

Fig. 3.35 Feature 17



No. of bounces = 4
 Energy = 0.11%
 Cumulative energy = 98.92%
 No. of SCs = 577

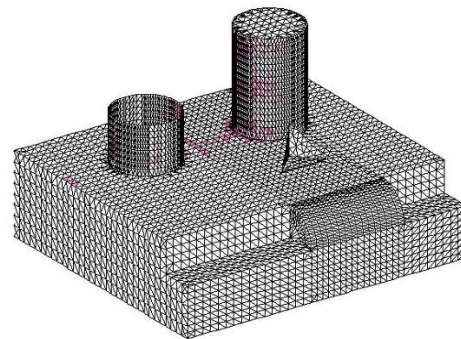
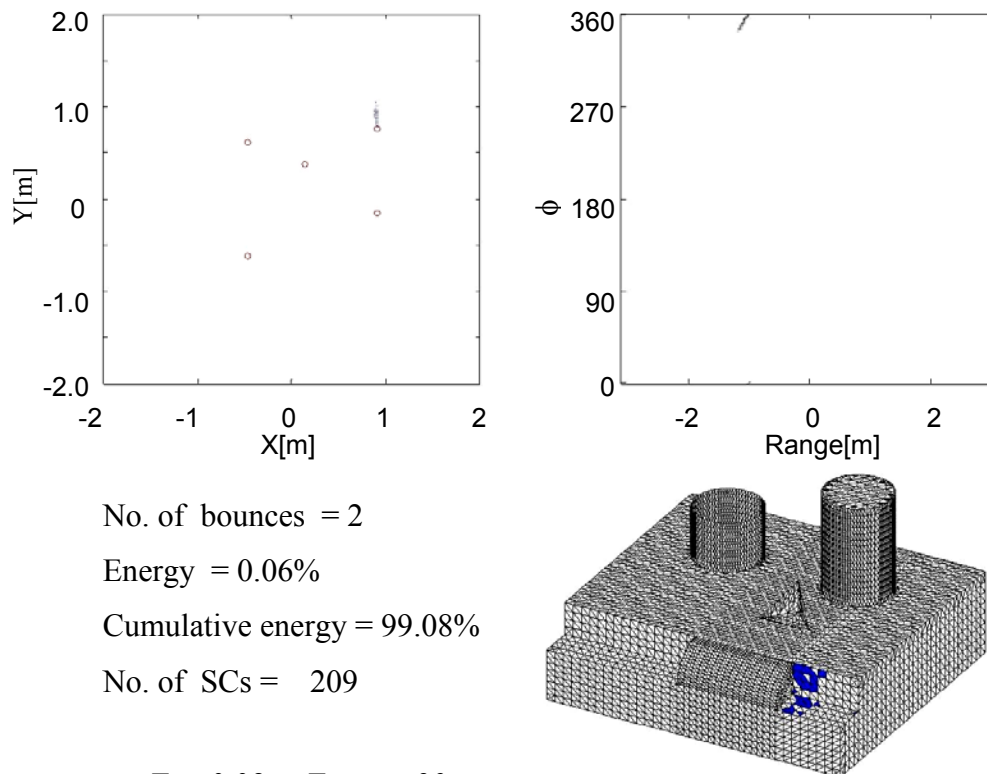
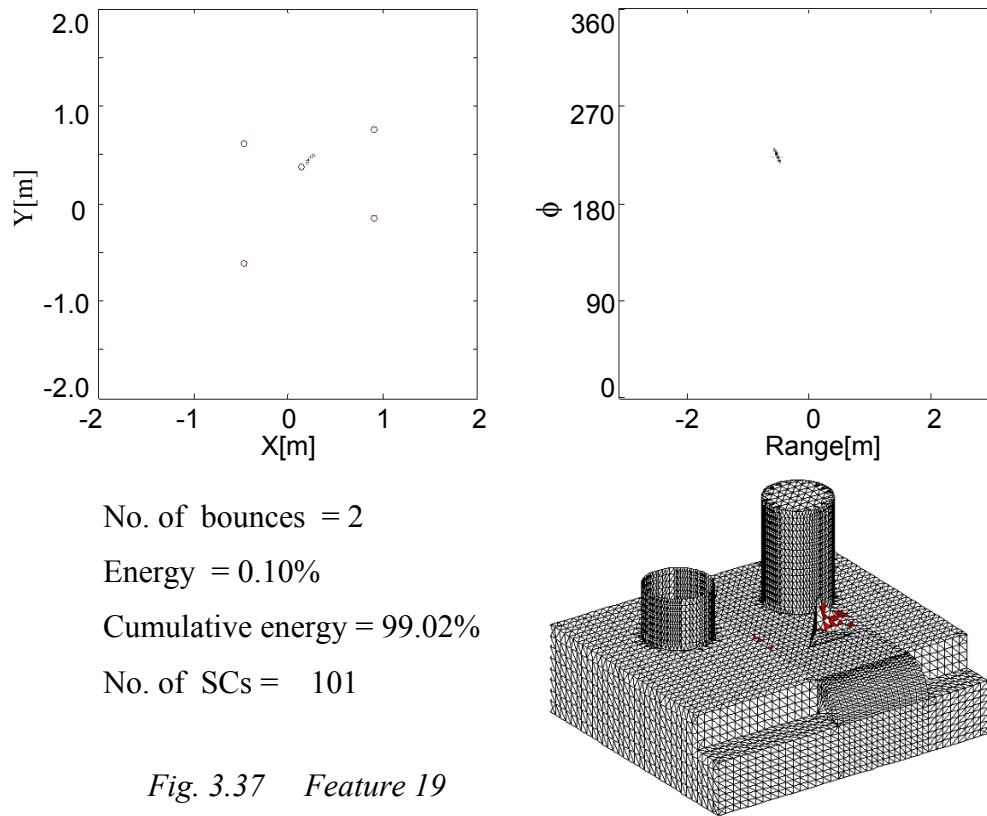


Fig. 3.36 Feature 18



3.4 Summary

Two problems for feature extraction from electromagnetic scattering data have been addressed in this chapter. Investigation found that parameterization for a model combining both the scattering centers and resonances is much more difficult and challenging. A CGA with adaptive feeding for parameterization based on the more complex model is first presented. The algorithm attempts to extract all the model parameters simultaneously. The effectiveness of the proposed method is demonstrated using the VFY 218 measurement data. It is shown that a much sparser representation can be obtained than the non-global parameterization approaches.

Second, current scattering center extraction algorithms do not account for the visibility of the scattering centers. Such visibility information could be very useful for feature identification. Two scattering feature extraction algorithms, the sinogram-based algorithm and the multi-dimensional segmentation algorithm, are presented to account for visibility. The first algorithm takes advantage of known angular behaviors of different scattering features in the sinogram to extract and parameterize the most important scattering features. The limit of this parameterization model is that it requires known behaviors of the scattering features in the data. The multi-dimensional segmentation algorithm is next researched to relax this constraint. Besides the 3-D location of the scattering centers, it also utilizes the ray history information and visibility as additional dimensions to parse the different scattering features in a higher dimensional space. Tests on the Slicy model show the effectiveness of the algorithm. The algorithm is useful in signature analysis and target feature extraction.

CHAPTER FOUR

ELECTROMAGNETIC INVERSE SCATTERING

In Chapter Two, it is shown that for a given frequency and aspect angle, the scattered field data for a known target can be numerically simulated by a forward EM solver. Based on the scattered field data, Chapter Three discusses how scattering features corresponding to different physical mechanisms and locations can be identified, separated and categorized. There exist other scenarios in practice, however, where the target is not known in advance. In these cases, scattering feature analysis has to be carried out in a blind mode.

One standard way to approach this type of problem is the inverse synthetic aperture radar (ISAR) imaging algorithm [24]. The ISAR algorithm sets out to create a microwave image of the target from collected radar scattering data by means of the inverse discrete Fourier transform (IDFT). Scattering centers can then be located by extracting the peaks in the obtained image. Although the technique is fast and robust in obtaining an approximate shape of a target, it suffers from the Fourier resolution limit and image artifacts due to multiple scattering phenomena [31]. More general inverse scattering algorithms can overcome the problems in ISAR imaging at the expense of much higher computational cost. Given the development of faster forward EM algorithms and improved computational power, these inverse scattering algorithms are worth pursuing because of their potentially superior performance.

In this chapter, the Fourier-based ISAR imaging algorithm and its difficulties in processing data with multiple scattering effects are first addressed. Next, an inverse scattering algorithm combining a forward electromagnetic solver with a genetic algorithm (GA) is developed to carry out the shape inversion of two-dimensional metallic objects with strong multiple scattering effects. An approach combining a hybrid genetic algorithm (HGA) with the tabu list concept is then proposed to increase the efficiency of the GA. Results of the shape inversion using the developed algorithm are presented for several Ipswich objects [38] based on both simulated and measured data.

4.1 Radar Imaging with Multiple Scattering Effects

A radar image is the spatial distribution of the radar reflectivity of a target [24]. The quality of a radar image should be judged by how faithfully it reproduces the spatial distribution of the target reflectivity. The problem of radar imaging is to form a target image by processing the received radar data reflected from the target.

Radar imaging can be considered as a linearized form of electromagnetic inverse scattering. In general, the scattering mechanisms on a real target are quite complex. They can be found rigorously by solving a field integral equation based on fundamental electromagnetic theory. The use of rigorous solutions, however, is computationally intensive. The simpler point-scatterer model is usually used in place of the rigorous solution in real radar imaging applications. In the scattering center model, the scattering effect of each point scatterer is assumed to be a two-way time delay and an amplitude

modulation. The scattered field from the target can then be written as the coherent summation of the scattered field from each point scatterer:

$$E(k, \theta) = \sum_i \sigma(x_i, y_i) \cdot \exp(-j2kR_i) \quad (\text{Eq. 4.1})$$

where $\sigma(x_i, y_i)$ is the reflectivity of the i^{th} point scatterer located at (x_i, y_i) , k is the wave number of the transmitted wave, θ is the aspect angle, and R_i is the distance between the radar and the i^{th} point scatterer.

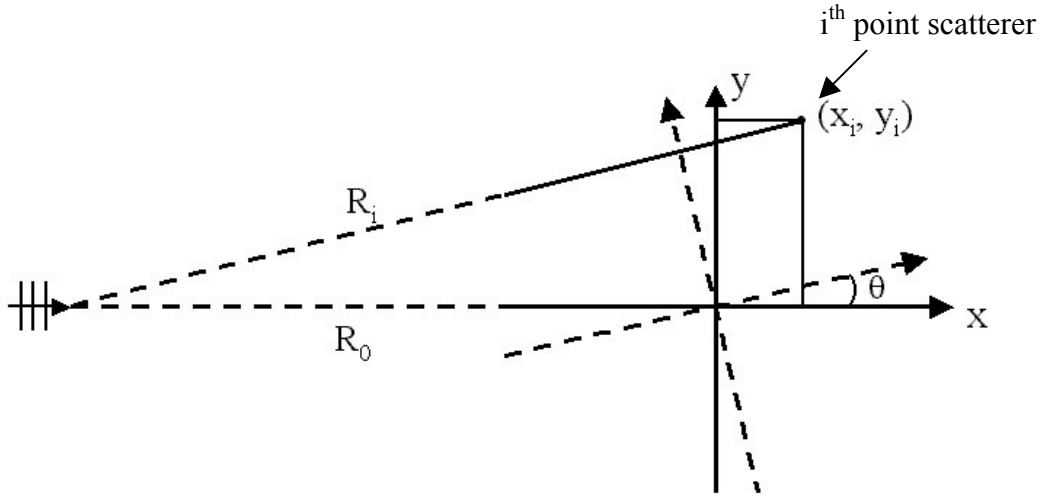


Fig. 4.1 Configuration of an ISAR imaging process

If the distance between the radar and the target satisfies the far-field criterion, and relative bandwidth and the angular extent θ about the nominal view are sufficiently small, the phase term in Eq. 4.1 can be approximated as (see Fig. 4.1):

$$\begin{aligned}\exp[-j2kR_i] &\cong \exp[-j2k(R_0 + x_i \cos \theta + y_i \sin \theta)] \\ &\cong \exp(-j2kR_0) \cdot \exp[-j2(kx_i + k_c y_i \theta)]\end{aligned}\quad (\text{Eq. 4.2})$$

where k_c represents the center frequency. Ignoring the R_0 term, Eq. 4.1 can be rewritten as

$$E(k, \theta) = \sum_{i=1}^{N_s} \sigma(x_i, y_i) \cdot \exp[-j2(kx_i + k_c y_i \theta)] \quad (\text{Eq. 4.3})$$

Eq. 4.3 shows that the relationship between $\sigma(x_i, y_i)$ and the received radar data $E(k, \theta)$ becomes a simple 2-D Fourier transform pair. Therefore, a radar image can be generated by inverse Fourier transforming the collected radar data in the frequency and angle domains.

The point-scatterer model of a target is an approximation of the real scattering from a target. The model assumes that only first-order scattering exists for each point scatterer. A real target usually contains other important scattering mechanisms, such as resonances and multiple scattering effects. Due to the mismatch between the real scattering data and the simplified point-scatterer model, unfocused artifacts can occur in the radar image at locations not corresponding to physical features of the target [25].

As an example, a set of measured radar data is processed by using the Fourier-based ISAR imaging technique combined with a motion compensation algorithm reported in [53]. The ISAR data were collected from the MERIC radar (a ground-based radar located in France) of two airplanes in flight with different flight paths (Fig. 4.2). The resulting ISAR images are compared to the predicted images generated by electromagnetic simulation.

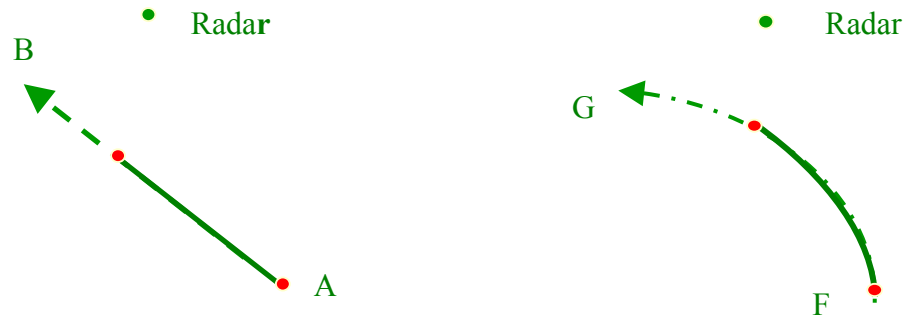


Fig. 4.2 Flight paths of two airplanes

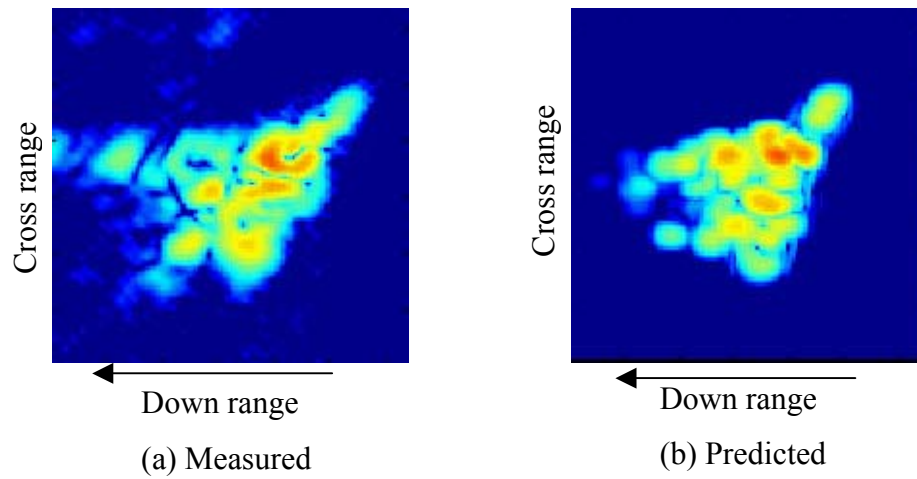


Fig. 4.3 Images from measured and predicted data for Target 1

In the actual data collection, the aircraft attitude data were also recorded by other motion sensors onboard the aircraft. Based on the attitude data and the geometry model of the target, an electromagnetic simulation is carried out by using the shooting-and-bouncing-ray solver to create a set of predicted radar images. Sample images from the

measured and predicted data for the two different targets are shown in Figs. 4.3 and 4.4, respectively.

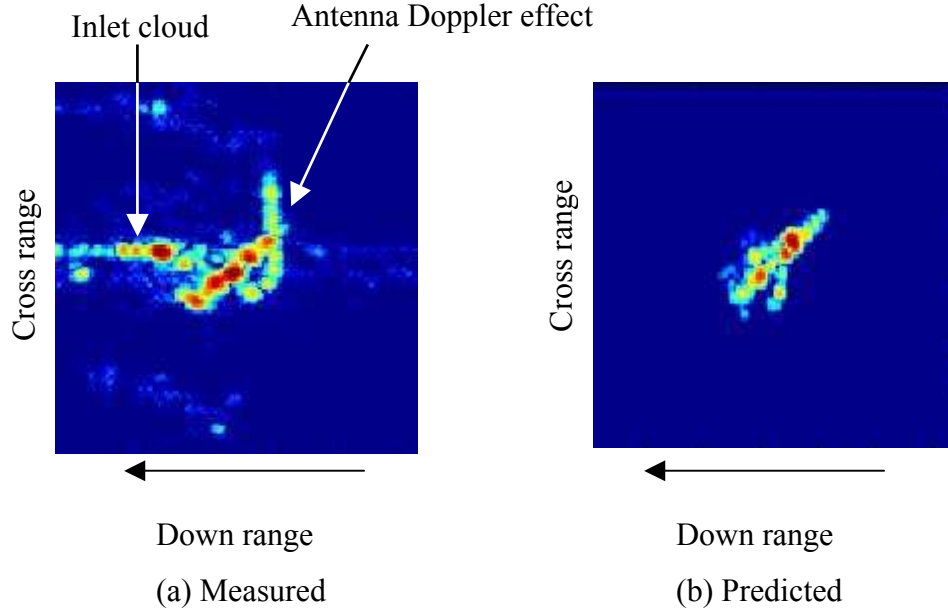


Fig. 4.4 Images from measured and predicted data for Target 2

As Fig. 4.3 shows, the image generated from the measured data matches well with the predicted image based on the computer model of the airplane. The key difference comes from the trailing cloud on the left side of the measured image. This range-delayed artifact comes from the multiple scattering phenomenon in the inlet duct of the actual target. It is absent in the simulated image since the computer model has a sealed inlet. This phenomenon is also observed in Fig. 4.4(a), as a trailing cloud exists along the down-range direction. This artifact makes the target image look a lot longer than the actual target. This artifact is absent in Fig. 4.4(b) since the number of ray bounces in the simulation is limited to 4. Fig. 4.5 shows the simulation results when the number of

bounces is increased to 50 to include all the multiple bounce rays. The simulated image shows an inlet cloud in the same vicinity as that in the measured image. However, the simulated inlet return is much stronger than that in the measured image. This difference possibly indicates that the actual inlet duct has been treated by a radar absorbing material, while the computer model assumes perfectly conducting duct walls.

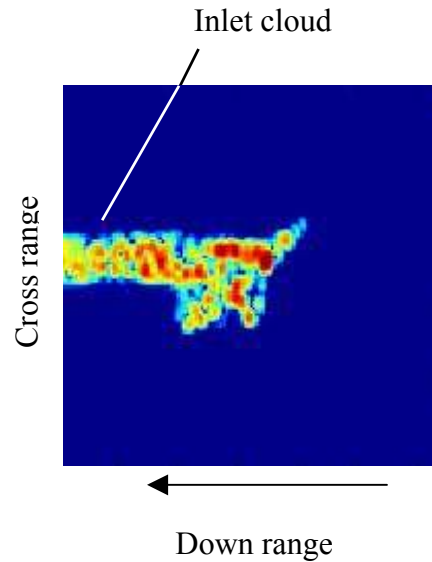


Fig. 4.5 Image from predicted data including higher-order scattering for Target 2

Another artifact in Fig. 4.4(a) is the cloud spreading across the aircraft nose along the cross-range direction. This artifact is due to the scanning antenna in the nose cone of the airplane. It violates the rigid-body assumption in the imaging model and thus results in an unfocused image feature. Detailed discussion of this type of Doppler artifacts can be found in [54].

As indicated above, the ISAR image formed from the actual measured data of a target is not necessarily a faithful geometrical image of the target. It may contain other features due to high-order scattering physics or target dynamics. Therefore, it is desirable to develop better inverse scattering algorithms to overcome these limitations.

4.2 Electromagnetic Inversion of Metallic Objects

4.2.1 Inverse scattering algorithms

Electromagnetic inverse scattering entails the reconstruction of the shape or material of an object from its scattered field data. As a step toward overcoming the image artifacts in radar imaging, research described in this and the following sections attempts to solve the electromagnetic inverse scattering problem more rigorously. Naturally, rigorously solving the electromagnetic inverse scattering problem is much more challenging.

The inverse problem can usually be cast into an optimization problem whereby the difference between the measured fields and the computed fields from a forward electromagnetic solver is minimized. Fig. 4.6 demonstrates the flowchart of the typical inverse scattering algorithm, where $E^s(f, \theta)$ represents the scattered electric field from the target with two independent parameters, the radar frequency f and the aspect angle θ . Typically, the optimization problem for inverse scattering is an extremely challenging one since the cost surface contains many local minima, signifying the non-uniqueness of the solution. Local optimization algorithms usually lead to undesirable solutions.

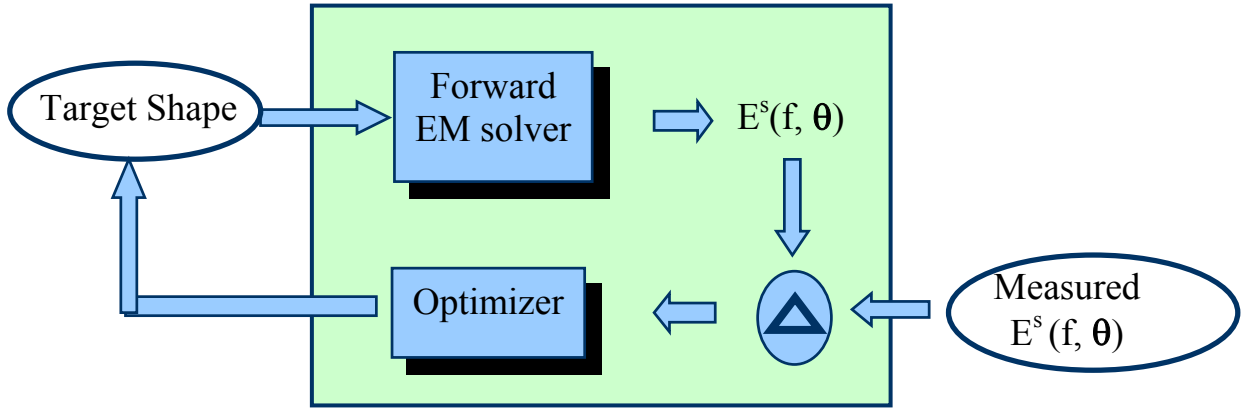


Fig. 4.6 Flowchart of the typical inverse scattering algorithm

As already discussed in Chapter Three, genetic algorithms are a good candidate for global optimization problems. Some researchers have explored the use of GA together with computational electromagnetic solvers to attack the inverse scattering problem [33-37, 39-42]. The algorithm starts with a set of random initial guesses for the unknown target shape. The forward EM solver calculates the simulated scattered field from these shapes and compares them with the collected data $E^s(f, \theta)$ from the real target. By defining a cost function, the optimizer picks out the members with better (lower) cost function values and produces members for the next iteration. The process is iterated until the cost function value for the optimized shape reaches the threshold set in advance. The eventual optimized shape is considered to be a good approximation to the real target.

4. 2. 2 Solving a 2-D inverse scattering problem for metallic objects with GA

Inverse scattering for metallic targets is a difficult problem because of relatively strong scattering and complex cost surface with many local minima, especially for

targets with multiple scattering effects [31]. As an initial investigation, the GA in conjunction with a method of moments (MoM) solver is implemented to invert metallic objects from the Ipswich measurement data set [38], which were released for the test of inverse scattering algorithms. In particular, attention is focused on concave metallic objects with strong multiple scattering effects.

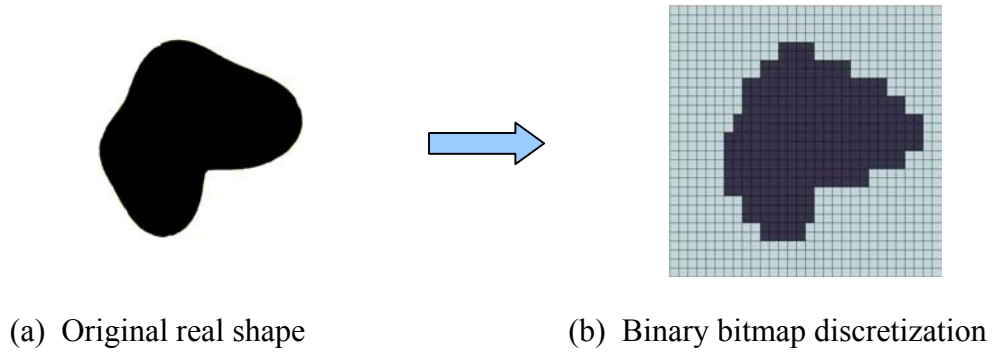


Fig. 4.7 Binary bitmap discretization for arbitrary 2-D shapes

For the inversion of two-dimensional (2-D) objects, two types of geometrical descriptions have been used: the Fourier series scheme [33,35] and the binary bitmap discretization [34,36]. The Fourier series scheme is efficient in representing smooth convex shapes; however, it does not work well for objects with highly concave shapes or disconnected parts. The binary bitmap discretization is a more general way to represent arbitrary 2-D shapes (Fig. 4.7). Its main drawback is the larger degrees of freedom required to accurately model simple shapes. More recently, cubic B-splines were also

investigated as a way to accurately represent complex shapes [37]. In this study, the binary bitmap approach is used to discretize the search space.

For the inverse problem for metallic objects, the measured scattering field E^{mea} from the object is known, while the shape and size of the object are unknown. The MoM solution based on the electric field integral equation is applied to obtain the rigorously computed scattered field E^{cal} from each assumed shape. To evaluate the performance of each shape, the cost function is defined as the root-mean-square (RMS) error between E^{mea} and E^{cal} (normalized with respect to E^{mea}):

$$Cost\ Func = \left\{ \frac{1}{M} \sum_{m=1}^M \frac{|E_m^{cal} - E_m^{mea}|^2}{|E_m^{mea}|^2} \right\}^{\frac{1}{2}} \quad (\text{Eq. 4.4})$$

A genetic algorithm is applied as the searching tool to minimize the cost function. In the GA implementation, the initial generation is produced randomly and each object shape is encoded into an $N \times N$ binary array with ones representing metal and zeros representing free space. A 2-D median filter is used as a low-pass filter to eliminate unrealistic shapes consisting of isolated cells. With a fixed window size of $M \times M$, the median filter slides through every cell of the binary array and sets the cell to one if the sum of the cell values within the window is greater than or equal to $M^2/2$ and zero otherwise. Fig. 4.8 shows a sample shape before and after the use of the described median filter, where the white cell represents the metal and the black is free space.

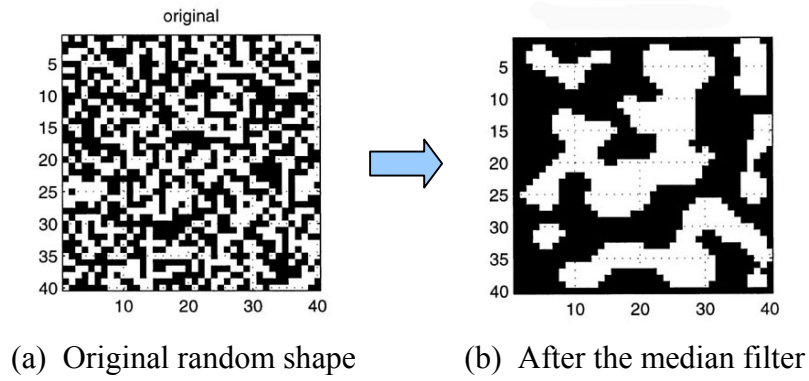


Fig. 4.8 Geometrical median filter for more realizable shapes

The median filter is applied once every several generations so that the isolated cells are cleared from time to time, while the new features created by the mutation and crossover operations have a chance to survive in the population.

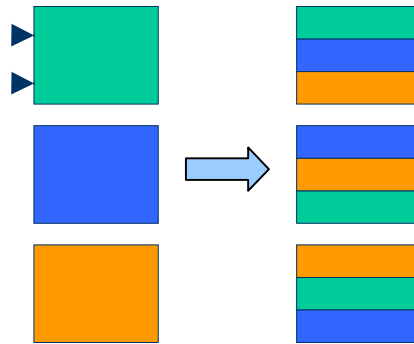


Fig. 4.9 Two-point crossover scheme

The cost function for each member is then calculated and the shapes with low cost values are selected as parents to produce the next generation. A 2-point crossover scheme involving three selected parents is used (Fig. 4.9). The process selects three parents and divides each parent into three parts. The three parent shapes are then intermingled to

produce three children shapes. This crossover scheme exhibits a more disruptive characteristic for regeneration than the conventional one-point or two-point crossover. It serves to counteract against the median filtering effect. The mutation operation, which is applied to the individual array cells, inverts the cell according to a preset mutation rate. The selection, crossover and mutation process is iterated until the lowest cost function in the population reaches a sufficiently small threshold or when the cost function does not decrease any further. The procedure of the MoM-GA inverse algorithm is given in Fig. 4.10.

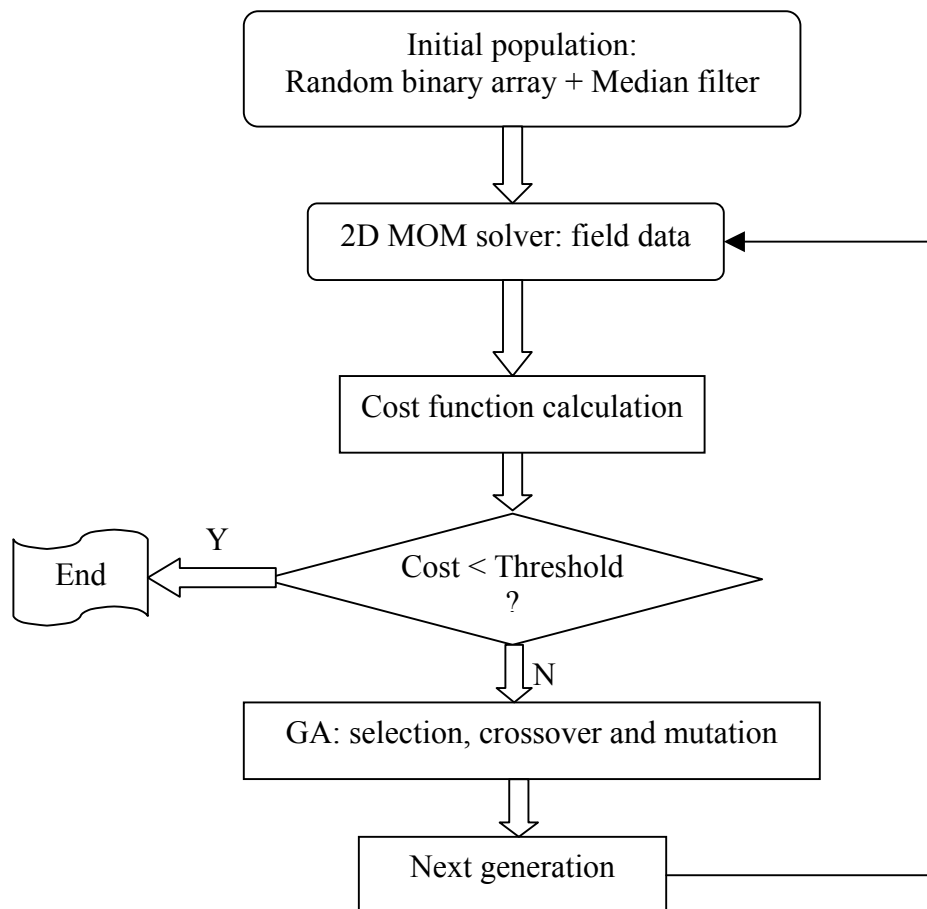


Fig. 4.10 Flow chart of MoM-GA inverse scattering algorithm

4. 2. 3 MoM-GA algorithm test

As a simple test, a convex metallic object shown in Fig. 4.7 is first inverted by the implemented algorithm. The electric field data is numerically simulated by the MoM and collected in the single frequency and monostatic case, where the transmitter and receiver are at the same location. A total of 32 views are set around the target. The size of the target is limited to 1.1λ . A 16×16 binary array is used to represent the target shape. Fig. 4.11 shows the actual shape and the MoM-GA reconstructed shape after 800 iterations. Figs. 4.12 (a) and (b) show the convergence rate and field comparison, respectively. The results indicate that the implemented algorithm can easily reconstruct a simple 2-D convex metallic shape. Both the eventual inverted shape and the field agree well with the real ones.

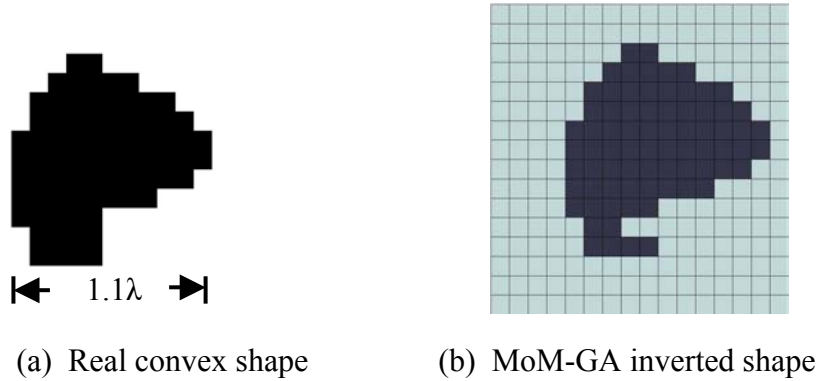


Fig. 4.11 Real and MoM-GA inverted shape for convex shape

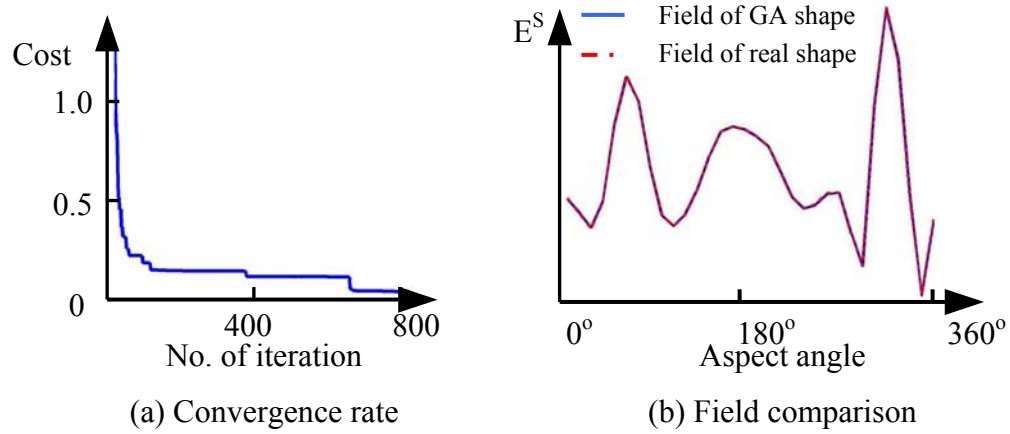


Fig. 4.12 Convergence rate and field comparison for convex shape inversion

The second test is for two circular cylinders with a mutual distance of 0.46λ , a little less than the 0.5λ Rayleigh resolution criterion (Fig. 4.13). A radar image by the Fourier-based imaging method is first formed. For this purpose, the scattered data is collected in bistatic configuration. The image result is shown in Fig. 4.14(a). As predicted, the two cylinders cannot be resolved when they are too close. For the MoM-GA shape inversion, the data is collected in the monostatic configuration along 32 uniformly-distributed aspect angles. A 32×32 binary array is used to represent the target space. Fig. 4.14(b) shows the reconstructed shape by the MoM-GA for two circular cylinders. The MoM-GA can resolve the two close cylinders by using less sampled data and shows a better resolution than that of Fourier-based algorithm.

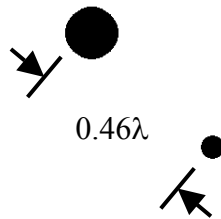
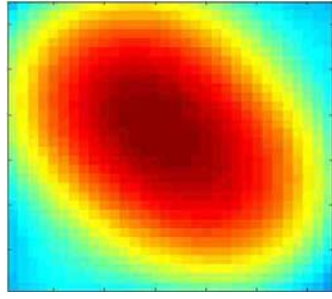
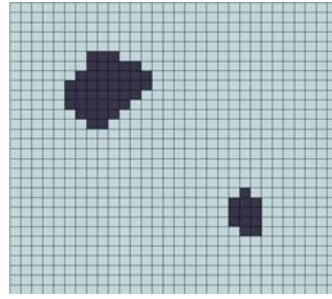


Fig. 4.13 Geometry of two circular cylinders

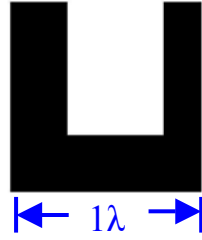


(a) Result of Fourier-based imaging

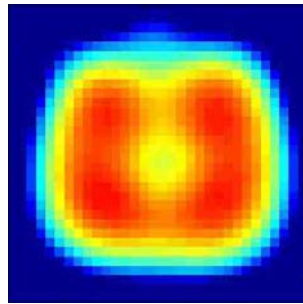


(b) MoM-GA inverted shape

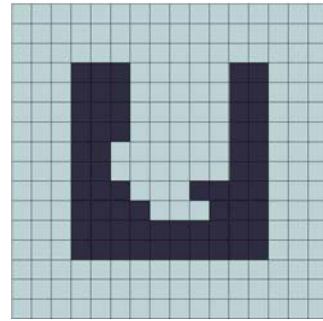
Fig. 4.14 Result comparison between the Fourier-based imaging algorithm and the MoM-GA for two close cylinders



(a) Geometry of the small cavity



(b) Result of Fourier-based imaging



(c) MoM-GA inverted shape

Fig. 4.15 Geometry and result comparison for the small cavity

The third test addresses a small cavity with the size of 1.0λ , as shown in Fig. 4.15(a). As in the previous test, a radar image is first formed. The image result is shown

in Fig. 4.15(b). The open end is hard to see because the multiple scattering effects smear the image. For the MoM-GA inversion, the data are collected in monostatic configuration along 32 uniformly distributed aspect angles. A 16×16 binary array is used to represent the target space. Fig. 4.15(c) shows the reconstructed shape by the MoM-GA for the small cavity structure. The result indicates the ability of the MoM-GA to reconstruct the target with strong multiple scattering effects.

4. 2. 4 Shape inversion of Ipswich objects

Three objects from the benchmark Ipswich measurements [38] are next selected for testing. They include the triangular cylinder, the dihedral, and the circular cavity. The data are inverted by using the MoM-GA. Results based on both the MoM-simulated field and the measured data are obtained. The Ipswich objects were measured at a single frequency of 10GHz in the bistatic configuration. There was a total of 36 transmitter positions around the object and 18 receiver locations for each transmitter position (Fig. 4.16). Fig. 4.17 shows the shapes and sizes of three metallic Ipswich objects selected for inversion. They are labeled as Ips009, Ips004, and Ips011, respectively. For Ips009 and Ips011, the electric field is parallel to the axis of the target. For Ips004, the electric field is perpendicular to the axis of the target.

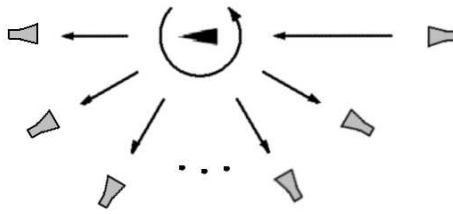
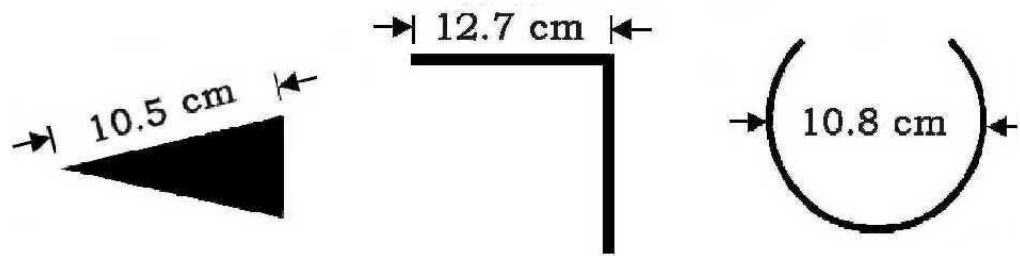


Fig. 4.16 Ipswich measurement setup



(a) Triangular cylinder (Ips009) (b) Dihedral (Ips004) (c) Circular cavity (Ips011)

Fig. 4.17 Real shapes of three Ipswich objects

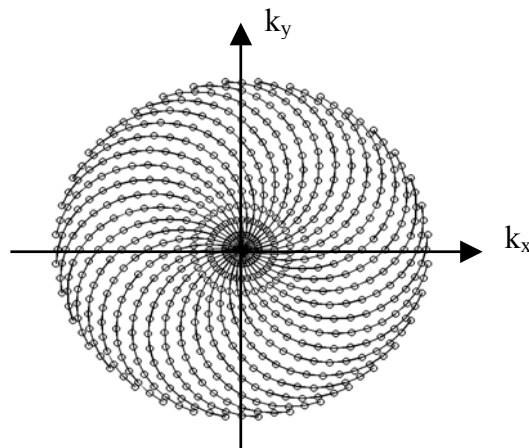


Fig. 4.18 Measured data in k -space

Imaging results for these targets are first generated using the traditional Fourier imaging method. The data are placed into the k-space, as shown in Fig.4.18, and are then interpolated into a rectangular grid. A 2-D fast Fourier transform is used to generate the images. The resulting images are shown in Fig. 4.19. In the results corresponding to the dihedral and circular cavity, image artifacts due to multiple scattering can be observed. For the dihedral, the multiple scattering causes strong artifacts outside the body. For the circular cavity, the multiple scattering effect obscures the opening of the cavity.

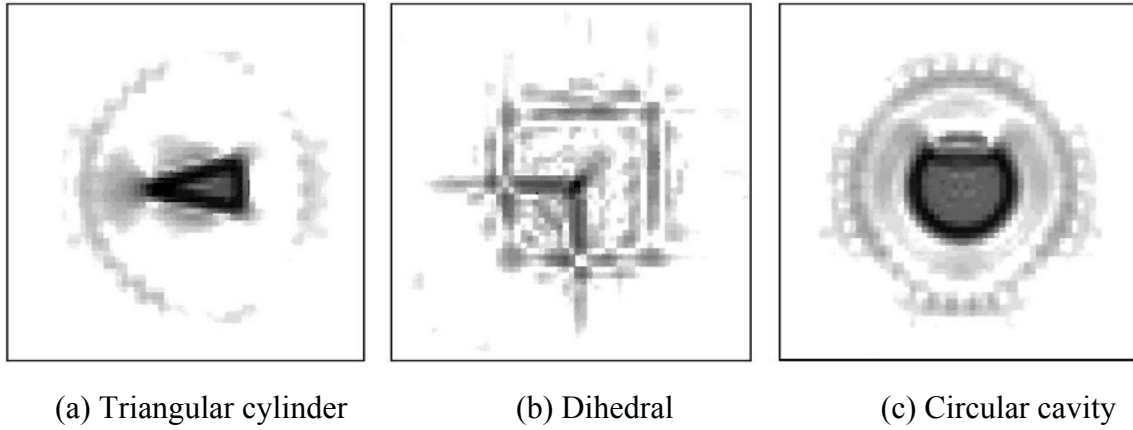
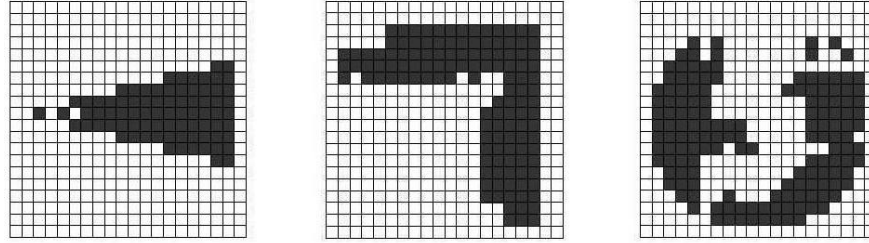


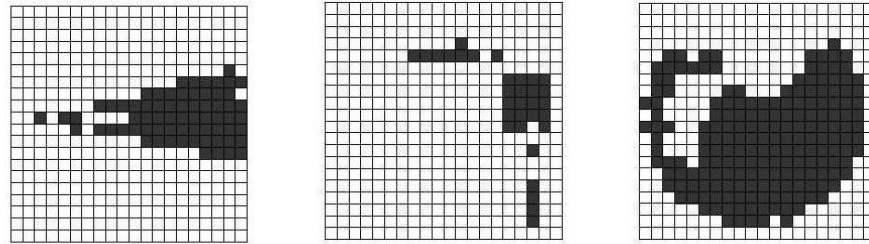
Fig. 4.19 Imaging results for three Ipswich objects

The MoM-simulated field data are first used as the input of the GA inversion algorithm. In all reconstructions, the number of chromosomes in the population is set to 100, and the crossover and mutation rates are set to 0.8 and 0.2, respectively. The search area is chosen to be 15cm×15cm for the dihedral, and 12cm×12cm for the triangular cylinder and circular cavity. The number of cells within this area is set to 20×20. The reconstructed results in Fig. 4.20(a) show the final inverted shapes of the three objects, which are in fairly good agreement with the real shapes. The final RMS costs for the

three objects were found to be 3%, 38%, and 18%, respectively. The dihedral and the circular cavity contain strong multiple scattering, yet their inverted shapes closely resemble the correct objects.



(a) Inversion based on MoM-simulated field



(b) Inversion based on measured complex field

Fig. 4.20 Inversion of three Ipswich objects

Next, the inversion algorithm is applied to the actual measured data. Fig. 4.20(b) shows the final reconstructed shapes. As shown, the inverted shape is good for the triangular cylinder, which has no multiple scattering effects. For the dihedral, the reconstructed shape is not continuous, but is quite similar to the real object. The circular cavity shows the most discrepancy with the real shape. The exterior and the opening of the cavity are correctly inverted, but the interior part of the cavity shape is not as satisfactory.

Interestingly, in all three inversions, the cost value of the final shape by GA is lower than that of the exact shape (38% vs. 58% RMS error for the triangular cylinder; 82% vs. 92% for the dihedral, and 55% vs. 73% for the circular cavity). These lower cost values indicate a mismatch between the measured field and the MoM-simulated field. Most likely, it is this difference that drove the GA to zoom onto a shape that is similar to the exact shape but has a lower cost value.

The RMS error listed above is particularly high for the dihedral. The agreement between the measured data and the MoM-computed data for this shape is good in field amplitude; however, a relatively large phase difference exists (even after adjusting for the rotation center) between the two results. The MoM results are also checked against other targets in the Ipswich data set, and the phase agreement is found to be good. Therefore, the phase data for the dihedral (Ips004) is suspect. As an alternative, the inversion is also performed based on only the amplitude of the fields, yielding the reconstructed shape shown in Fig. 4.21. The quality of this reconstruction is close to that of the MoM-simulated data shown in Fig. 4.20(a).

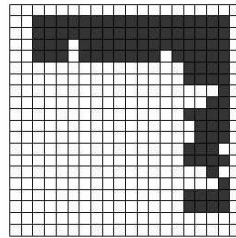


Fig. 4.21 Inversion of the dihedral based on only the amplitude of the measured data

4. 2. 5 Shape inversion by SBR-GA scheme for large cavity structures

Many inverse scattering problems in the real world deal with electrically large targets. For electrically large targets, faster forward electromagnetic solvers have to be utilized in place of the MoM to compute the scattered field, such as the ray-tracing method discussed in Chapter Two. Not only is the ray-tracing based forward solver efficient for the computation of electrically large targets [14,15], it is also capable of obtaining multiple frequency information with little additional cost.

As a study, a 2-D straight cavity structure with a size of 4m×4m is generated to evaluate the effect of frequency diversity. Simulated data created by the shooting and bouncing ray (SBR) solver is used as input. The data are collected in the monostatic case with 36 views uniformly distributed around the target. Two different frequency configurations are considered. In the first case, only single frequency data are created at 300 MHz. In the second case, the data are created through a frequency band ranging from 175 MHz to 425 MHz. The GA combined with SBR (SBR-GA) is then applied to invert the shape of the cavity. Fig. 4.22 shows the results for both the single-frequency and the eight-frequency cases. While the shape is successfully reconstructed in the multi-frequency case, it fails in the single frequency case. The finding indicates that frequency diversity is useful in the inversion of targets with strong multiple scattering effects. This result is consistent with [55], which claims that possible advantages can be achieved by frequency diversity.

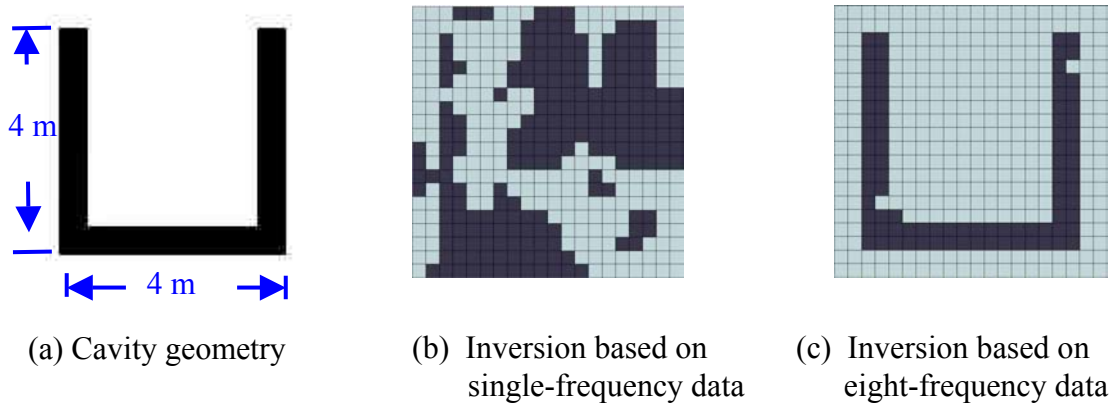


Fig. 4.22 Effect of frequency diversity

Four 2-D cavities with more complex inside structures and strong multiple scattering effects are then set up and inverted by using the SBR-GA (Fig. 4.23). The sizes of the cavities are over 10 wavelengths. The target space is discretized into a 20×20 binary array. Because the goal here is to reconstruct the inside structure of each cavity, the three outside walls (left, right and bottom) are fixed for each cavity, only leaving the upper opening and the inside part (a 16×15 binary array) for reconstruction. The inversion uses the simulated data from 64 frequencies swept from 267 MHz to 533 MHz.

The inverted shapes are shown on the right column in Fig. 4.23. For the first two straight cavities, the inside structures are correctly reconstructed. For the other two cavities, which have more complex inside structures, the inverted shapes are not perfect but they indicate that the algorithm is trying to approach the inside structure. The results indicate that the SBR-GA is a promising method for inversion of large metallic targets with strong multiple scattering effects.

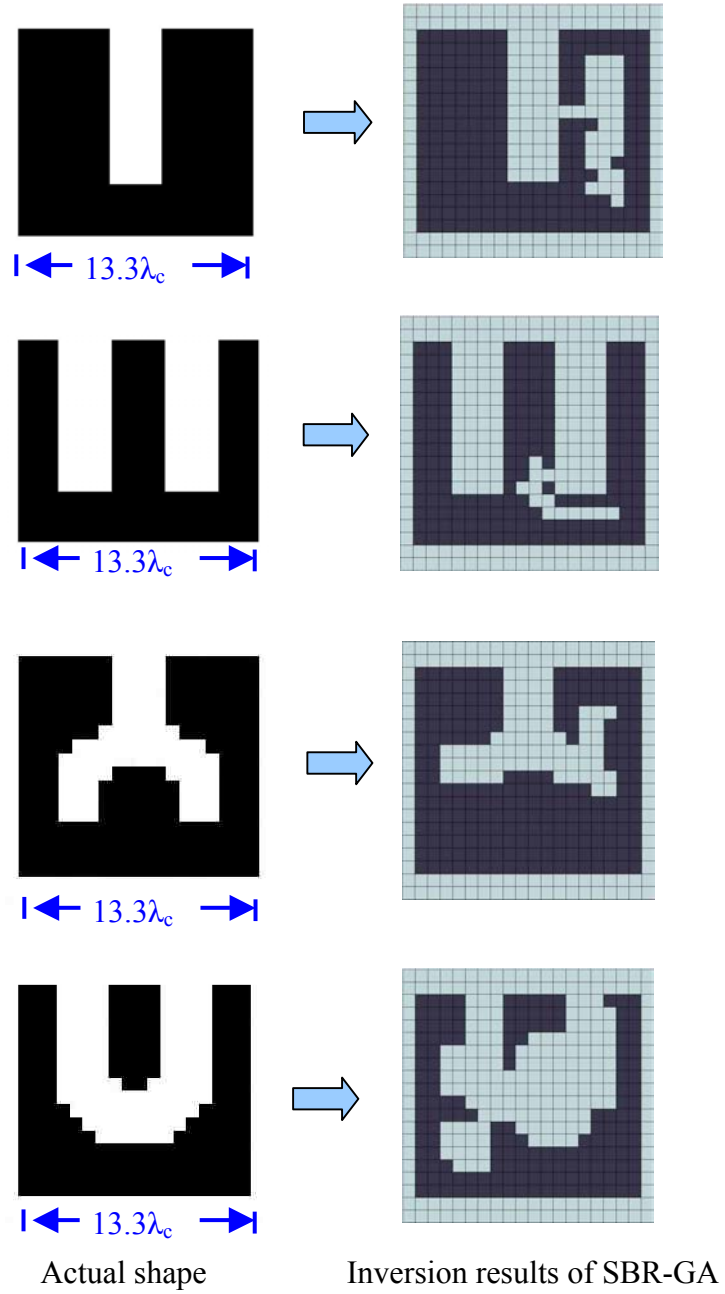


Fig. 4.23 Inversion of electrically large cavities with fixed outsides

4.3 Shape Inversion by HGA-Tabu

The main obstacle to extending the inverse scattering algorithms presented in the last section to complex targets in 3-D is the long computation time. The reason is that the

convergence rate of the GA scheme is usually slow for more complex targets. In this section, an approach combining the hybrid genetic algorithm (HGA) with the tabu list concept is proposed to increase the search efficiency of the GA scheme. In addition, a more efficient point-interpolation scheme is adopted to describe the 2-D geometry.

4. 3. 1 Obstacle of the GA-based inverse scattering schemes

While GAs are well suited in searching for the global optimum, the scheme presented in the previous section suffers from slow convergence and inefficient target geometry description. A simple improvement to the geometry description problem is to use a multi-point spline interpolation scheme, as shown in Fig. 4.24. For 2-D problems, the number of variables is just $2N$ (the x and y coordinates for each point) if N points are used. The obtained spline curve can then be discretized into a number of points to meet the discretization requirement for the EM solver.

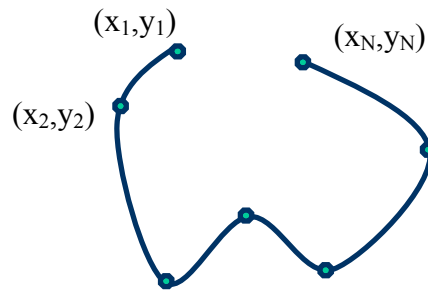


Fig. 4.24 Multi-point spline interpolation scheme

The GA convergence problem is due to the complexity of the multi-dimensional cost surface. This problem is especially acute for convex structures such as cavities. A sample 2-D cost surface is demonstrated in Fig. 4.25(a). Here five points are used to describe a circular cavity. If four points are fixed and only the remaining single point is disturbed within the square dashed-line area, a sample cost surface with many local minima can be observed as shown in Fig. 4.25(b). The plot indicates the complexity of the cost surface for even a single-point problem. The cost surface for an N-point problem will be much more complex, and the process of finding the real solution (the global minimum) would be very time-consuming.

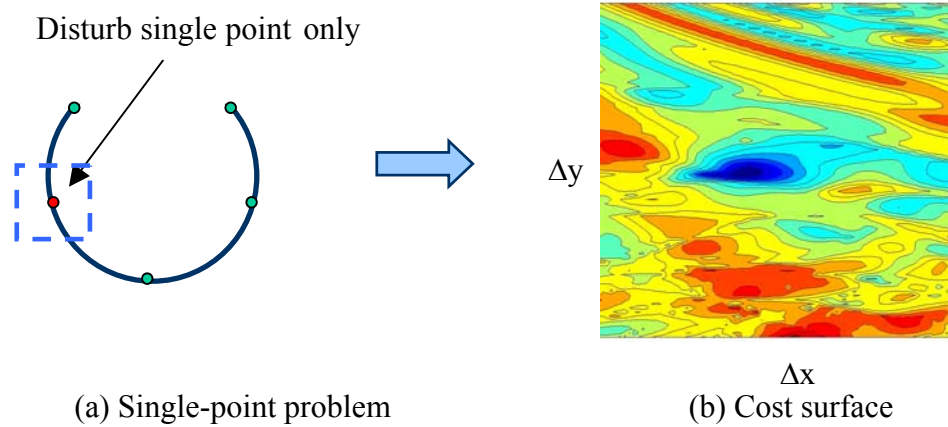


Fig. 4.25 Sample cost surface for single -point problem

Because the evolutionary process for the standard GA to reach a cost minimum is generally slow in comparison to that of a local search algorithm, a natural way to speed up the simple GA would be to hybridize it with a local search algorithm. In fact, researchers in different disciplines have explored the use of this so-called hybrid GA (HGA) [41, 42]. While improving performance, the HGA also causes some

inefficiencies. Specifically, the parent selection scheme of the GA usually gives priority to the best members with lower cost values, and this priority usually leads to a population highly clustered around the local minima (Fig. 4.26). This clustering is necessary for the simple GA to evolve closer to the exact minimum. For the HGA, however, since the local minima have been completely explored by the local search, such clustering will lead to the re-exploration of those regions, which is quite wasteful.

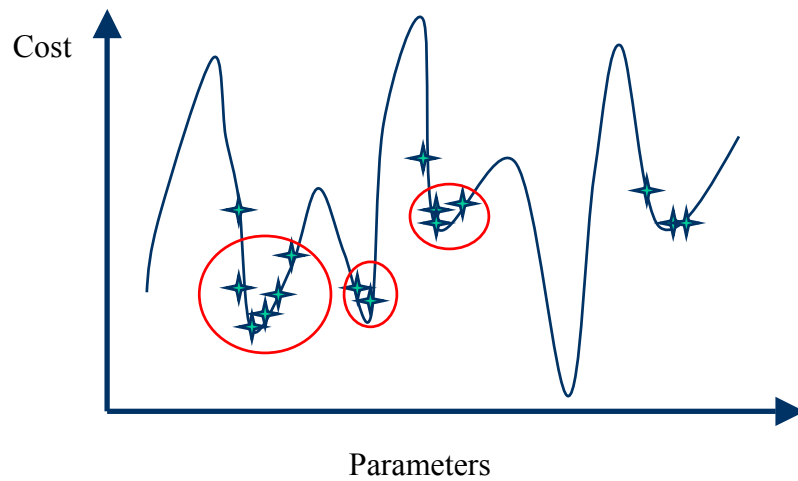


Fig. 4.26 Clustering around the local minimum

The tabu search (TS) is another global search strategy that has been developed for combinatorial problems [56, 57]. The TS is a local search algorithm with memory. The most important feature of the TS is that it utilizes a tabu list to prevent the revisiting of local minima. The study here proposes a technique combining HGA with the tabu list concept to increase the efficiency of the HGA. The tabu list is adopted to exclude those regions in the parameter space that have already been explored by the local search. In

this manner, there will be no revisiting of the explored regions, and the GA population can be spread out to explore new regions, thus improving search efficiency.

4.3.2 HGA-Tabu approach

In the HGA-Tabu approach, the initial generation is produced randomly. The new population is then produced through the selection, crossover and mutation operators. After these standard GA processes, the best member \bar{P} is selected as the initial guess to carry out a local search. In this study, the gradient search reported in [31] is adopted as the local search algorithm. The resulting local minimum in the parameter space, denoted as \bar{P}_{\min} (see Fig. 4.27), is then placed into the new GA population. In addition, a gradient search is also carried out to obtain the local maximum \bar{P}_{\max} from the same initial guess in order to estimate the extent of the local minimum. After both local searches are completed, the region that is centered at the minimum and limited by the radius is defined as $|\bar{P}_{\max} - \bar{P}_{\min}|$ the "*tabu region*" and is recorded into a tabu list. Symmetry around the local minimum is assumed in this construct. In subsequent GA reproductions, all new members are checked against this tabu list to ensure that none is in the tabu regions of the sample space. Thus, the population is forced to spread out to unexplored regions, resulting in higher HGA search efficiency. Further, a new tabu region is appended to the tabu list every time a new local minimum is explored by the local search.

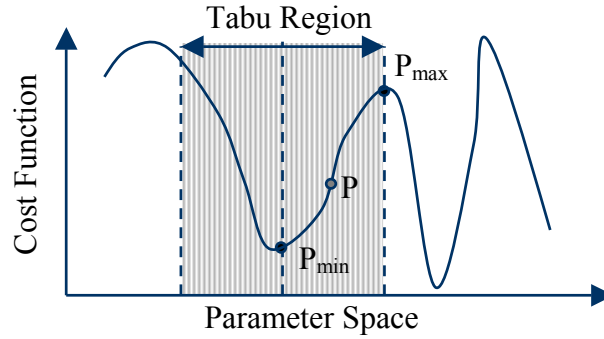


Fig. 4.27 Establishment of the “tabu region”

The flow chart of HGA-Tabu algorithm is shown in Fig. 4.28. In using the inverse algorithm to reconstruct the shape of a cavity from its scattered field data, a set of randomly created shapes are first generated. These shapes are described by N ordered points in a two-dimensional space. The profile of the object is then obtained by using a spline interpolation. Next, the MoM solution to the electric field integral equation is used as the forward electromagnetic solver to generate the computed scattered field E^{cal} from each assumed shape. The cost function is defined by Eq. 4.4 as the root-mean-squared (RMS) difference between E^{cal} and the measured scattered field E^{mea} . The HGA-Tabu algorithm is then applied as the optimizer to minimize the cost function. A binary-encoded GA is used in this approach.

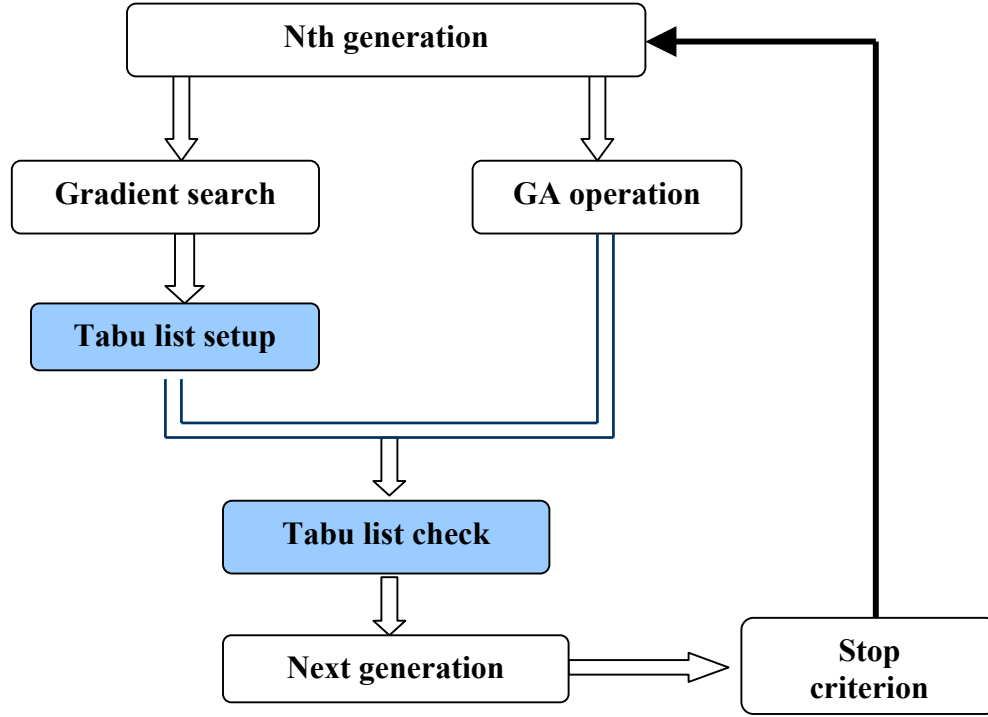


Fig. 4.28 Flow chart of HGA-Tabu algorithm

4.3.3 HGA-Tabu evaluation

The proposed HGA-Tabu algorithm is first evaluated on a purely mathematical cost function expressed as:

$$f(X) = 10n + \sum_{i=1}^n [(x_i - 2.5)^2 - 10 \cos 2\pi(x_i - 2.5)] \quad (\text{Eq. 4.5})$$

where n is the dimension of the function and argument x_i is within $[0, 10]$. This function has 10^n minima, and the global minimum is zero at $x_i = 2.5$. This model is ideal for testing the stability and efficiency of optimization algorithms. Fig. 4.29 shows the contour plot of the function when $n=2$. For the HGA-Tabu evaluation, n is set to 3 and the GA population is set to 100. The hybrid GA and the HGA-Tabu algorithms are then

run 20 times independently to obtain the averaged convergence curves plotted in Fig. 4.30. While the HGA never has a chance to reach the global minimum zero within 300 generations, the HGA-Tabu algorithm shows good stability in reaching the global minimum. Moreover, the convergence rate of the HGA-Tabu algorithm is much faster than that of the HGA.

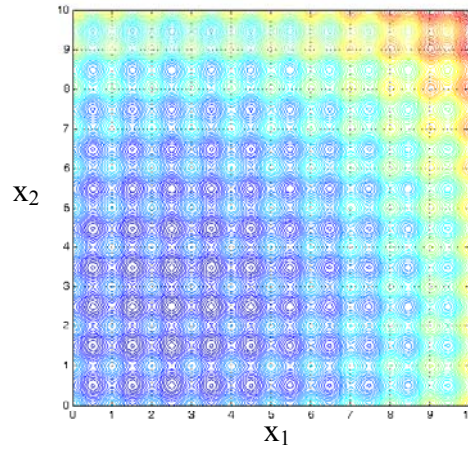


Fig. 4.29 Contour plot of Eq. 4.5

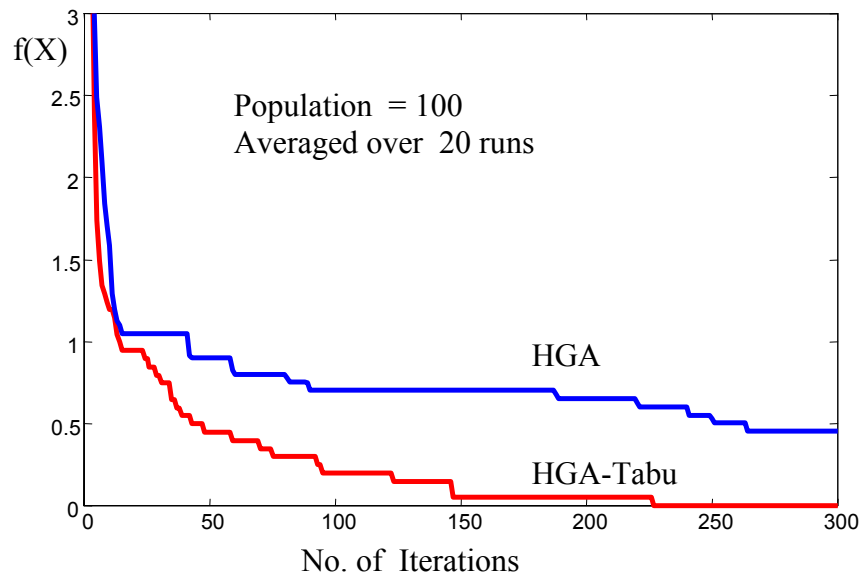


Fig. 4.30 Evaluation of efficiency: HGA vs HGA-Tabu

The HGA-Tabu algorithm is used to reconstruct the shape of a metallic, partially open, circular cylindrical cavity with a diameter of 10.8 cm (Ips011 in the Ipswich data set) [38]. The measurement was taken at a single frequency of 10 GHz in a bistatic configuration, as indicated in Fig. 4.16. The electric field is parallel to the axis of the cylinder. The population number for the GA is set to 200, the geometry is described by $N=5$ points, and the crossover and mutation rates are set to 0.8 and 0.4, respectively. The search area is chosen to be $16.2 \text{ cm} \times 16.2 \text{ cm}$. The inversion algorithm is first tested by using MoM-simulated data as the input. The results (see Fig. 4.31) show that the HGA-Tabu algorithm is able to converge to the correct shape after an average of 30 generations, and the final shape is in excellent agreement with the actual shape. In comparison to the HGA, the HGA-Tabu showed an improvement of about 100 generations for convergence.

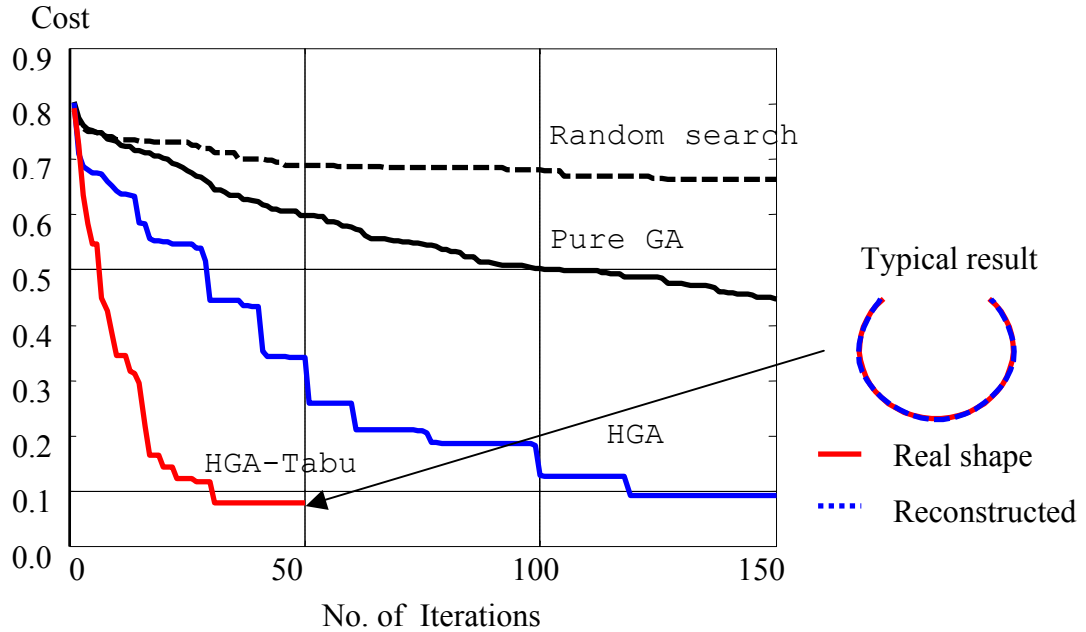


Fig. 4.31 Evaluation of efficiency: HGA vs HGA-Tabu

Next, the inversion algorithm is applied to the actual measured data for Ips011. Fig. 4.32 shows the convergence comparison between random search, simple GA, HGA, and HGA-Tabu. All the results are averaged over 10 independent runs with different initial populations. As expected, the simple GA shows an improvement over the random search. The HGA further improves the convergence rate of the simple GA. The best results are consistently obtained by the HGA-Tabu algorithm. To achieve an RMS of 0.55, the HGA requires an average of 220 generations while the HGA-Tabu algorithm requires only an average of 75 generations. It should be noted that the high RMS error for the best result obtained by the HGA-Tabu algorithm comes from a mismatch between the measured field and the MoM-simulated field, as discussed earlier in Section 4.2.4.

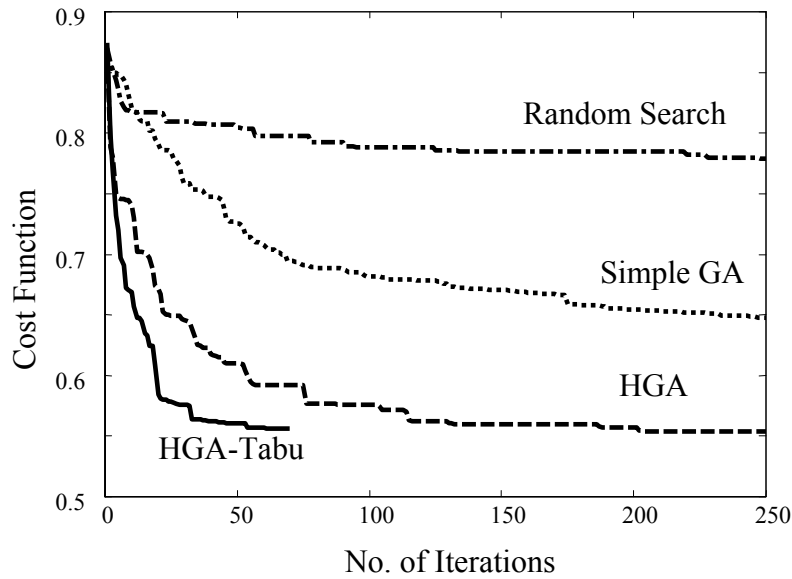


Fig. 4.32 Convergence comparison for inversion of Ips011 for random search, SGA, HGA, and HGA-Tabu

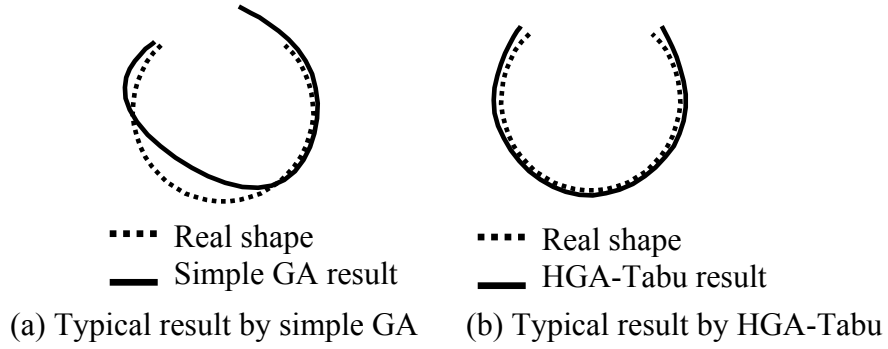


Fig. 4.33 *Ips011 inversion results from measured data*

Fig. 4.33(a) shows the typical shape from the simple GA after 250 generations plotted against the real profile of the cavity. The result indicates that more iterations are needed for convergence. Fig. 4.33(b) shows the typical reconstructed shape from the HGA-Tabu after 75 generations. As shown, the inverted shape is very close to the real profile. The overhead of implementing the gradient search in each generation is about 10% of the total computation cost. The time for the tabu list check is negligible, as there is no cost function evaluation.

4.4 Summary

High-frequency radar imaging takes advantage of the simplicity and efficiency of the Fourier inversion method, but results in image artifacts from higher-order scattering effects. In contrast, the inverse scattering algorithm has the ability to reconstruct targets containing strong higher-order scattering effects. In this chapter, GA-based schemes are developed to address the shape inversion of convex structures containing strong multiple scattering effects.

Investigation indicates that the slow convergence rate of the simple GA scheme limits its application. This problem can be partially overcome by an approach combining the hybrid genetic algorithm with the tabu list concept. The tabu idea is adopted to increase the search efficiency by forbidding revisits of local minima already explored by the local search. The algorithm is tested by reconstructing the shape of a metallic cavity based on both simulated and measured data from the Ipswich measurement set. Inversion results from the HGA-Tabu show faster convergence and a higher success rate than those of the simple GA and the hybrid GA. The computation overhead per generation for the new algorithm is small. The algorithm has potential for speeding up other optimization problems.

CHAPTER FIVE

CONCLUSIONS

In this dissertation, three related electromagnetic scattering problems, namely, high frequency electromagnetic ray tracing, scattering feature extraction, and inverse scattering are studied. New approaches are presented to advance the state of the art in each of the areas. In Chapter Two, the presented study leads to an alternative ray tracing algorithm which can outperform the traditional algorithms for complex targets. The performance of the proposed techniques demonstrates their potential application to the study of high-frequency EM scattering prediction. In Chapter Three, a GA-based algorithm with an adaptive-feeding technique is first developed to simultaneously extract both scattering centers and resonances. Scattering feature extraction algorithms are then developed with the consideration of the visibility of scattering centers. In Chapter Four, inverse scattering problems with strong multiple scattering effects are studied. A GA-based method is presented to invert the shapes with multiple scattering effects. An approach combining hybrid GA with the tabu list idea are then developed to further improve the performance of the GA-based inversion algorithms.

A multiplaten Z-buffer (MPZ) ray tracing algorithm is first implemented and evaluated in Chapter Two. Investigation indicates that the performance of the traditional binary space partition (BSP) tree-based ray tracing algorithm increases monotonically

with the number of facets. The performance of the proposed MPZ algorithm, on the other hand, is independent from the number of facets describing the targets, and only slightly increases with the number of MPZ layers. Further study reveals that its performance is in fact depend on the number of pixels a ray travels across. Based on this fact, a multi-aspect approach (MAMPZ) is proposed to further improve the performance of the MPZ algorithm by setting up multiple MPZs in diverse directions to reduce the number of pixels a ray has to travel across. Tests on the numerous targets indicate that the developed ray tracer can outperform the traditional standard algorithm for targets with as few as several thousand facets if the MPZ aspect resolution is set fine enough. The algorithm is also used for ray tracing on the IGES parametric surfaces. An investigation indicates that the MPZ ray tracer can quickly find the approximate intersection point and the corresponding surface parameters. Note that the exact intersection between a ray and the parametric surface has to be located by a numerical iterative search. The approximate intersection point detected by the MPZ ray tracer is very close to the exact intersection point (within a pixel), which makes the MPZ IGES ray tracer potentially attractive because the numerical search for the intersection can be seeded with a very good initial guess.

As an alternative to the traditional BSP tree-based algorithm, the developed MAMPZ algorithm has shown a potential for speeding up the high-frequency electromagnetic scattering computations.

In Chapter Three, a GA method is first developed to simultaneously extract all the scattering centers and resonances from scattering data. The proposed algorithm achieves

a better convergence by combining with an adaptive feeding idea. Tests on the measured data indicate that the proposed algorithm can achieve sparser results than other non-global based methods. In addition, methods are developed to utilize the specific visibility of scattering centers to separate and categorize the different scattering mechanisms. Based on a simple parametric model, the developed sinogram-based algorithm can detect and separate the features such as stable scattering centers and features with circular angle behavior. To make the method more general to include scattering mechanisms with all kinds of angular behaviors, a multi-dimensional segmentation algorithm is proposed. The successful implementation on the test targets demonstrates its advantage. It is shown that the spatial locations of the scattering centers alone do not offer sufficient separability to allow for a successful extraction of all the scattering mechanisms. When the feature space is increased to higher dimensions by utilizing ray bounce numbers and angular persistence, the scattering mechanisms become naturally separated and can thus be extracted more easily.

Radar imaging takes advantage of model simplicity and the efficiency of FFT technique, but it is unable to deal with high-order scattering effects. As an alternative, a GA-based inverse scattering algorithm is developed in Chapter Four to reconstruct the target shapes containing strong high-order scattering effects. Two GA-based schemes, MoM-GA and SBR-GA, are developed and evaluated by taking the 2-D shape inversion for Ipswich metallic objects and large metallic cavity structures, respectively. It is found that unlike the radar imaging technique, the algorithms have the potential to deal with objects having strong multiple scattering effects. More investigation indicates that

convergence of the inverse scattering algorithms is usually slow. Meanwhile, the MoM-like full-wave forward solver, while accurate, becomes basically impractical for use in the inversion for electrically large targets, and the high-frequency approximate solver has to replace it.

The slow convergence rate of the GA inversion scheme is improved by the HGA-Tabu algorithm. This approach combines a hybrid GA with the tabu list concept, which increases the search efficiency by forbidding revisits of local minima. The test results from the HGA-Tabu algorithm show faster convergence and a higher success rate than those of the simple GA and hybrid GA. The computation overhead per generation for the new algorithm is small. The algorithm has potentially valuable use in other optimization problems.

BIBLIOGRAPHY

- 1 Harrington, R. F., *Field computation by moment method*, The Macmillan Co., New York, 1968
- 2 McDonald, B. H. and Wexler, A., Finite element solution of unbounded field problems, *IEEE Trans. Microwave Theory Tech.*, Vol. 20, No. 12, 1972, pp. 841-847
- 3 Heubner, K. H., *The finite-element method for engineers*, Wiley, New York, 1975
- 4 Yee, K. S., Numerical solution of initial boundary-value problems involving Maxwell's equations, *IEEE Trans. Antennas Propagat.*, Vol. 14., May. 1966, pp. 302-307
- 5 Chen, C., The discrete Fourier transform method for solving differential-integral equations in scattering, *IEEE Trans. Antennas Propagat.*, Vol. 37., Aug. 1989, pp. 1023-1041
- 6 Coifman, R., Rokhlin, V. and Wandzura, S., The fast multipole method for the wave equation: a pedestrian prescription, *IEEE Antennas Propagat. Mag.*, Vol. 35, No. 3, 1993, pp. 7-12
- 7 Steinberg, B.Z., Leviatan, Y., On the use of wavelet expansions in the method of moments, *IEEE Trans. Antennas Propagat.*, Vol. 41, No. 5, May 1993, pp. 610-619
- 8 Bowman, J. J., Senior, T. B. A. and Uslenghi, P. L. E., *Electromagnetic and acoustic scattering by simple shape*, North-Holland, Amsterdam, 1969
- 9 Rius, J. M, Vall-Llossera, M., High frequency radar cross section of complex objects in real time, 1991 *AP-S/URSI Symp. Dig.*, Vol. 2, June 1991, pp. 1062-1065
- 10 Rius, J. M.; Ferrando, M. and Jofre, L., GRECO: graphical electromagnetic computing for RCS prediction in real time, *IEEE Antennas Propagat. Mag.*, Vol. 35,

Issue 2, Apr. 1993, pp. 7 - 17

- 11 Lee, S. W., Jeng, S. K., Yu, C. L ., Liang, C. S., Physical optics impulse response from faceted targets, *IEEE Antennas Propagation Soc. Int. Symposium Digest*, 1992, pp. 1456 –1459
- 12 Keller, J. B., Geometrical theory of diffraction, *J. of Opt. Soc. Am.*, Vol. 52, No. 2, 1962, pp. 116-130
- 13 Ufimtsev, P. Y., *Method of edge waves in the physical theory of diffraction*, Moscow, Sovyetskoye Radio, 1962
- 14 Ling, H., Chou, R. and Lee, S. W., Shooting and bouncing rays: calculating the RCS of an arbitrarily shaped cavity, *IEEE Trans. Antennas Propagat.*, Vol. 37., Feb. 1989, pp. 194-205
- 15 Andersh, D. J., Hazlett, M., Lee, S. W., Reeves, D. D., Sullivan, D. P. and Chu, Y., XPATCH: a high frequency electromagnetic-scattering prediction code and environment for complex three-dimensional objects, *IEEE Antennas Propagat. Mag.*, vol. 36, 1994, pp.65-69
- 16 Sung, K. and Shirley, P., Ray tracing with the BSP tree, *Graphics GEMS IV*, edited by Kirk, D., Academic Press, 1992, pp. 271-274
- 17 Potter, L.C., Da-Ming Chiang, Carriere, R., Gerry, M.J., A GTD-based parametric model for radar scattering, *IEEE Trans. Antennas Propagat.*, Vol. 43, Oct. 1995, pp.1058 - 1067
- 18 Hurst, M. P. and Mittra, R., Scattering center analysis via Prony's method, *IEEE Trans. Antennas Propagat.*, Vol. 35, No. 8, 1987, pp. 986-988

- 19 Moghaddar, A., Ogawa, Y., and Walton, E. K., Estimating the time delay and frequency decay parameters of scattering components using a modified MUSIC algorithm, *IEEE Trans. Antennas Propagat.*, Vol. 42, No. 10, 1994, pp. 1412-1418
- 20 Bhalla, R. and Ling, H., Three-dimensional scattering center extraction using the shooting and bouncing ray technique, *IEEE Trans. Antennas Propagat.*, Vol. 44, No. 11, 1996, pp. 1445-1453
- 21 Li, Q., Rothwell, E.J., Chen, K.M., and Nyquist D.P., Scattering center analysis of radar targets using fitting scheme and genetic algorithm, *IEEE Trans. Antennas Propagat.*, Vol. 44, No. 2, 1996, pp. 198-207
- 22 Hughes, E.J. and Leyland M., Using multiple genetic algorithms to generate radar point-scatterer models, *IEEE Trans. Evol. Comput.*, Vol. 4, No. 2, 2000, pp. 147-163
- 23 Su, T. and Mittra, R., A genetic-algorithm-based extrapolation technique for high frequency scattering analysis, *Proceedings of 2002 URSI National Radio Science Meeting*, San Antonio, TX, 2002, 3, pp. 307
- 24 Mensa, D. L., *High Resolution Radar Imaging*, Artech House, Dedham, MA, 1981
- 25 Trintinalia, L.C. and Ling, H., Joint time-frequency ISAR using adaptive processing, *IEEE Trans. Antennas Propagat.*, Vol. AP-45, No. 2, 1997, pp. 221-227
- 26 Moore, J. and Ling, H., Super-resolved time-frequency analysis of wideband backscattered data, *IEEE Trans. Antennas Propagat.*, Vol. AP-43, No. 6, 1995, pp. 623-626
- 27 Trintinalia, L. C., Bhalla, R. and Ling, H., Scattering center parameterization of wide-angle backscattered data using adaptive Gaussian representation, *IEEE Trans. Antennas Propagat.*, Vol. AP-45, Nov. 1997, pp. 1664-1668

- 28 Choi, I.-S. and Kim, H.-T., One dimensional evolutionary programming-based CLEAN, *Electron. Lett.*, Vol. 37, No. 6, 2001, pp. 400-401
- 29 Bhalla, R., Moore, J. and Ling, H., A global scattering center representation of complex targets using the shooting and bouncing ray technique, *IEEE Trans. Antennas Propagat.*, Vol. AP-45, Dec. 1997, pp. 1850-1856
- 30 Zhou, Y. and Ling, H., Electromagnetic inversion of Ipswich objects with the use of the genetic algorithm, *Microwave Optical Tech. Lett.*, Vol. 33, June 2002, pp. 457-459
- 31 Otto, G. P. and Chew, W. C., Microwave inverse scattering-local shape function imaging for improved resolution of strong scatters, *IEEE Trans. Microwave Theory Tech.*, Vol. 42, Jan. 1994, pp. 137-141
- 32 Roger, A., A Newton-Kantorovitch algorithm applied to an electromagnetic inverse problem, *IEEE Trans. Antennas Propagat.*, Vol. AP-29, Mar. 1981, pp.232 - 238
- 33 Chiu, C.C. and Liu, P. T., Image reconstruction of a perfectly conducting cylinder by the genetic algorithm, *IEE Proc. Microwave, Antennas Propag.*, Vol. 143, No. 3, 1996, pp. 249-253
- 34 Takenaka, T., Meng, Z. Q., Tanaka, T. and Chew, W. C., Local shape function combined with genetic algorithm applied to inverse scattering for strips, *Microwave Optical Tech. Lett.*, Vol. 16, No. 6, 1997, pp. 337-341
- 35 Qian , Z. and Hong, W., Image reconstruction of conducting cylinder based on FD-MEI and genetic algorithms, *IEEE Antennas Propagation Soc. Int. Symposium Digest*, 1998, Vol. 2, pp. 718 –721

- 36 Pastorino , M., Massa, A. and Caorsi, S., A microwave inverse scattering technique for image reconstruction based on a genetic algorithm, *IEEE Trans. Instrum. Meas.*, Vol. 49, June 2000, pp. 573 –578
- 37 Barkeshli, K., Mokhtari, M. and Amiri, N. M., Image reconstruction of impenetrable cylinders using cubic B-splines and genetic algorithms, *IEEE Antennas Propagation Soc. Int. Symposium Digest*, Boston, July 2001, Vol. 2, pp. 686-689
- 38 McGahan, R. V., and Kleinman, R. E., Second annual special session on image reconstruction using real data, *IEEE Antennas Propagat. Mag.*, Apr. 1997, Vol. 39, No. 2, pp. 7-9
- 39 Pastorino, M., Massa, A. and Caorsi, S., A microwave inverse scattering technique for image reconstruction based on a genetic algorithm, *IEEE Trans. Instrum. Meas.*, Vol. 49, June 2000, pp. 573 –578
- 40 Michalski, K. A., Electromagnetic imaging of elliptical-cylindrical conductors and tunnels using a differential evolution algorithm, *Microwave Optical Tech. Lett.*, Vol. 28, Feb. 2001, pp. 164-169
- 41 Yen, J.; Liao, J. C.; Randolph, D.; Lee, B. and Randolph, D., A hybrid approach to modeling metabolic systems using genetic algorithm and simplex method, *IEEE Trans. Syst., Man, Cybern.*, Vol. 28, April 1998, pp. 173–283
- 42 Park, C. and Jeong, B., Reconstruction of a high contrast and large object by using the hybrid algorithm combining a Levenberg-Marquardt algorithm and a genetic algorithm, *IEEE Trans. Magnetics*, Vol. 35, May 1999, pp. 1582-1585

- 43 Hu, J. L., Ma, Y., Lin, S. M. and Wang, W. B., Multiplaten Z-buffer algorithm and its application to multiple scattering, *Electron. Lett.*, Vol. 33, No. 10, 1997, pp. 825-826
- 44 Hu, J. L., Chan, C. H. and Lin, S. M., Computation of RCS of jet engine with complex terminations based on multiplaten Z-buffer algorithm, *Microwave Optical Tech. Lett.*, Vol. 23, No. 5, Dec. 1999, pp. 281-284
- 45 Initial Graphic Exchange Specification (IGES), *U. S. Federal Information Processing Standard (FIPS) 177*, 1989
- 46 Kajiya , J. T., Ray Tracing Parametric Patches, *Computer Graphics*, Vol. 16, No. 3, July 1982, pp. 245-254
- 47 Joy, K. I. and Bhetanabhotla , M. N., Ray tracing parametric surface patches utilizing numerical techniques and ray coherence, *ACM SIGGRAPH Computer Graphics*, Vol. 20, No. 4, Aug. 1986, pp.279-285
- 48 Glassner, A. S., Space subdivision for fast ray tracing, *IEEE Comp. Graphics App.*, Vol. 4, No. 10, 1984, pp. 15-22
- 49 Kajiya, J. T., New techniques for ray tracing procedurally defined objects, *Proc. 10th Annual Conf. on Comp. Graphics Interactive Tech.*, Detroit, July 1983, pp.91-102
- 50 Weile, D. S. and Michielssen, E., E-plane microwave filters: community genetic algorithm optimization, *Electromagnetic Optimization by Genetic Algorithms*, (Eds: Rahmat-Samii Y. and Michielssen, E., John Wiley & Sons, New York, NY, 1999), pp. 324-343
- 51 Wang, H. T. G., Sanders, M. L., and Woo A., Radar cross section measurement data of VFY 218 configuration, Naval Air Warfare Center, China Lake, CA, Tech. Rep. NAWCWPNS TM07621, 1994

- 52 MacQueen, J. B., Some methods for classification and analysis of multivariate observations, *Proc. the Fifth Berkeley Symp. Math. Stat. Prob.*, 1967, pp. 281 -297.
- 53 Wang, Y., Ling, H. and Chen, V. C., ISAR motion compensation via adaptive joint time-frequency techniques, *IEEE Trans. Aerospace Electron. Syst.*, Vol. 34, No. 2, Apr. 1998, pp. 670-677
- 54 Li, J., Model-based signal processing for radar imaging of targets with complex motions, Ph.D dissertation, Univ. of Texas at Austin, 2002
- 55 Bucci, O. M., Crocco, L., Isernia, T. and Pascazio, V., Inverse scattering problems with multifrequency data: reconstruction capabilities and solution strategies, *IEEE Trans. Geosci. Remote Sensing*, Vol. 38, No. 4, July 2000, pp. 1749-1756
- 56 Glover, F., Tabu search-part I, *ORSA Journal on Computing*, Vol. 1, 1989, pp. 190-206
- 57 Glover, F. and Laguna, M., *Tabu Search: Mordern Heuristic Techniques for Combinatorial Problems*, Blackwell Scientific Publication, Oxford, 1993

Vita

Yong Zhou was born in Hubei, China in 1971, the son of Jiacheng Zhou and Guomao Wang. He received his B.S. in space physics from Peking University and M.S.E.E degree in electromagnetic and microwave technology from The Second Academy of China Aerospace Corporation, Beijing, China, in 1992 and 1995, respectively. He was a Research Engineer with the Beijing Institute of Environmental Features from 1995 to 1999. He was a Ph.D. student worked as a Graduate Research Assistant in the Department of Electrical and Computer Engineering at the University of Texas at Austin from 1999 to 2004. His research interests include computational electromagnetics, radar signal processing, scattering feature analysis, and inverse scattering.

Permanent address: 2501 Lake Austin Blvd Apt H102, Austin, TX 78703

This dissertation was typed by the author.