

5-2012

A Linear Sampling multiple frequency method for target detection

Tyler Bowman

University of Arkansas, Fayetteville

Follow this and additional works at: <http://scholarworks.uark.edu/eleguht>

Recommended Citation

Bowman, Tyler, "A Linear Sampling multiple frequency method for target detection" (2012). *Electrical Engineering Undergraduate Honors Theses*. 21.

<http://scholarworks.uark.edu/eleguht/21>

This Thesis is brought to you for free and open access by the Electrical Engineering at ScholarWorks@UARK. It has been accepted for inclusion in Electrical Engineering Undergraduate Honors Theses by an authorized administrator of ScholarWorks@UARK. For more information, please contact scholar@uark.edu.

A LINEAR SAMPLING MULTIPLE FREQUENCY METHOD FOR TARGET DETECTION

A LINEAR SAMPLING MULTIPLE FREQUENCY METHOD FOR TARGET DETECTION

An Undergraduate Honors College Thesis

in the

Department of Electrical Engineering
College of Engineering
University of Arkansas
Fayetteville, AR

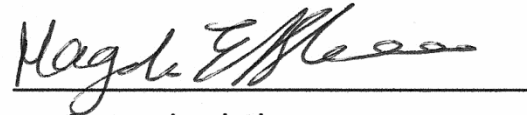
by

Tyler Clark Bowman

25 April 2012

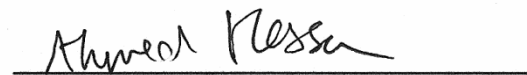
This thesis is approved.

Thesis Advisor:



Prof. Magda El-Shenawee

Thesis Committee:



Dr. Ahmed Hassan

ABSTRACT

In the field of inverse scattering problems of electromagnetic imaging, there are many techniques that can be used to detect unknown objects. Generally these methods maintain a direct relationship between the precision of the target shape and the amount of time required to obtain the solution. However, it has been shown that hybridization, or a combination of techniques, can be used to obtain the shape reconstruction that is accurate and less expensive computationally.


Previous research in the Computational Electromagnetics Group of Professor El-Shenawee at the University of Arkansas has looked into the use of hybridization by combining the Level Set algorithm, a precise but slow shape reconstruction technique, with the Linear Sampling Method (LSM), a very fast technique. It was found that taking the result from the LSM and using it as the initial guess of the Level Set algorithm can enhance the computational expenses. The goal of this work is to implement a multiple frequency model of the LSM and to test it for two-dimensional metallic targets.

The results show that a reasonably accurate reconstruction could be attained using the multiple frequency LSM technique to detect single and multiple targets. The results also show that some frequencies, not known a priori, can deteriorate the detection of the target. However, averaging the detected targets over a band of frequencies has shown a potential of more accurate results compared to the use of a single frequency. This work focused on the microwave band of frequency; however, the preliminary results will be extended to the terahertz band.

THESIS DUPLICATION RELEASE

I hereby authorize the University of Arkansas Libraries to duplicate this thesis when needed for research and/or scholarship.

Agreed


Tyler Clark Bowman

Refused

Tyler Clark Bowman

ACKNOWLEDGEMENTS

Thanks goes to Dr. Magda El-Shenawee, the Computational Electromagnetics Group leader at the University of Arkansas, for serving as the thesis advisor for this work. Special thanks also goes to Dr. Ahmed Hassan, who provided the scattering data from the objects under test and the original LSM code.

This work was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-10-2-0093. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

TABLE OF CONTENTS

I.	INTRODUCTION	1
A.	Background Research	1
B.	The Linear Sampling Method	3
1.	Basic LSM	3
2.	Multiple Frequency LSM	5
II.	APPROACH AND IMPLEMENTATION	7
A.	Simulation Procedure	7
1.	Single Object: Six-pointed Star	7
2.	Multiple Objects: Ellipse Pair	8
3.	Threshold Value Comparison	9
B.	Results	10
1.	First Simulation Run	10
2.	Single Object Best-Fit Reconstructions	11
3.	Multiple Objects Best-Fit Reconstruction	14
4.	Same Threshold Value Comparison	16
III.	CONCLUSIONS AND FUTURE RESEARCH	22
IV.	REFERENCES	23
	APPENDIX A: MATLAB CODE	24
	APPENDIX B: THRESHOLD SWEEP CONTOUR SELECTIONS	30

LIST OF FIGURES

Fig. 1.1: The results of previous hybridization research.	2
Fig. 1.2: Example of the contour field resulting from LSM.	5
Fig. 2.1: Single object case: a six-pointed star	7
Fig. 2.2: Multiple objects case: a pair of ellipses	9
Fig. 2.3: First set of field reconstructions.	10
Fig. 2.4: Single object reconstruction at each frequency with α solved.	11
Fig. 2.5: Single object best-fit contours selected at each frequency with α solved.	12
Fig. 2.6: Single object reconstruction at each frequency with α constant.	12
Fig. 2.7: Single object best-fit contours selected at each frequency with α constant.	13
Fig. 2.8: Percent error graph for single object reconstruction.	13
Fig. 2.9: Second scenario reconstruction at each frequency with α constant.	15
Fig. 2.10: Multiple object best-fit contours selected at each frequency with α constant.	15
Fig. 2.11: Percent error graph for the multiple object reconstruction.	16
Fig. 2.12: Single object comparison of multiple frequency reconstructions at each threshold value when solving for α_z .	18
Fig. 2.13: Error comparison of single object at each threshold value when solving for α_z .	18
Fig. 2.14: Single object comparison of multiple frequency reconstructions at each threshold value when using constant α_z .	19
Fig. 2.15: Error comparison of single object at each threshold value when using constant α_z .	19
Fig. 2.16: Multiple object comparison of multiple frequency reconstructions at each threshold value when using constant α_z .	20

Fig. 2.17: Error comparison of multiple objects at each threshold value when using constant α_z .	20
Fig. B.1: Single object contours when solving for α_z with threshold of -0.1.	30
Fig. B.2: Single object contours when solving for α_z with threshold of -0.3.	30
Fig. B.3: Single object contours when solving for α_z with threshold of -0.5.	31
Fig. B.4: Single object contours when solving for α_z with threshold of -0.7.	31
Fig. B.5: Single object contours when using constant α_z with threshold of -0.1.	32
Fig. B.6: Single object contours when using constant α_z with threshold of -0.3.	32
Fig. B.7: Single object contours when using constant α_z with threshold of -0.5.	33
Fig. B.8: Single object contours when using constant α_z with threshold of -0.7.	33
Fig. B.9: Multiple object contours when using constant α_z with threshold of -0.1.	34
Fig. B.10: Multiple object contours when using constant α_z with threshold of -0.3.	34
Fig. B.11: Second scenario contours when using constant α_z with threshold of -0.5.	35
Fig. B.12: Second scenario contours when using constant α_z with threshold of -0.7.	35

I. INTRODUCTION

The shape reconstruction using microwave imaging is one of the most current research topics in the field of electromagnetics. In particular, in reverse scattering problems, where an unknown object is reconstructed from the fields scattered when it is illuminated with microwave frequency signals, have been shown to be very useful for a wide variety of applications. Such applications may include advanced medical imaging, concealed or buried object detection and reconstruction, and security scanning. In all of these cases, the ability to retrieve an accurate reconstruction is absolutely vital. However, it is also important that the computational expense be kept low. This work focused on the microwave band of frequency; however, the preliminary results will be extended to the terahertz band.

A. BACKGROUND RESEARCH

There has been a significant amount of prior research in inverse scattering solution techniques at the University of Arkansas, specifically concerning a method using the Level Set algorithm [1]-[8]. Level Set is a very precise method that starts with an initial guess, or a large area in which the object being observed is estimated to exist. It then takes the field data scattered by the object across a range of frequencies and refines the shape and location of the object at each discrete frequency before moving to a higher one. The resulting image tends to be a very precise reconstruction of the object being illuminated.

The significant drawback of the Level Set method is that the highly precise reconstructions come at the price of solution CPU time. This particular problem makes the Level Set method impractical for use in some applications in which accurate results are needed in real time. Thus, other methods were researched to determine a process that could be used for the reconstruction that is both precise and requires less time to obtain the image.

To this end, it was suggested to use a method of hybridization between two techniques [9], [10]. A faster but less precise technique could be used to approximate the size and location of the object in the calculation space. Then the Level Set algorithm could use the output of the faster algorithm as the initial guess for its refinement over the frequency range. The faster method chosen for this hybrid algorithm is the Linear Sampling Method (LSM), which has been shown to work well in hybridization with other techniques [11]. The LSM will be explained in greater detail in the following section. Using the LSM, it was found that an accurate reconstruction could be obtained for the case of a six-pointed star at a far lower CPU time than the Level Set algorithm alone. The results of the two stages of the hybridization can be seen in Figure 1.1.

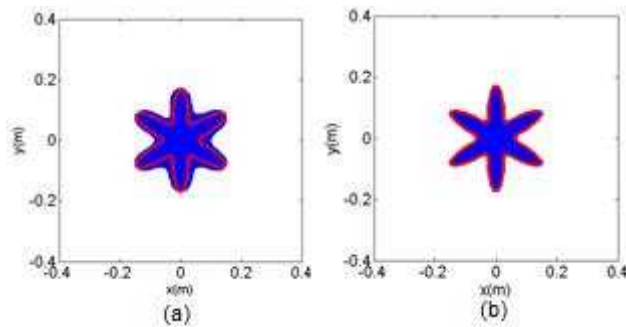


Fig. 1.1: The results of the previous hybridization research. (a) Results from LSM. (b) Results of Level Set using hybridization [9].

These reconstruction results can then be compared to the actual contour to find the percentage of error. The process used to describe the error divides the entire calculation space into a grid of 250 pixels width in both the x- and y-dimensions. The pixels of the image received by the image reconstruction algorithm are then compared to the pixels contained within the true object. The number of mismatched pixels between the two is then divided by the pixels of the true object and multiplied by 100 to get the percentage. The error from the hybridization images

shown in Figure 1, as well as the solution time and a comparison to the results of each method taken individually, are listed in Table 1.

Table 1: Comparative CPU time and error of Level Set, LSM, and hybrid algorithms [9].

Algorithm	CPU Time	Error
Level Set	9.5 hours	3%
LSM	4 minutes	40%
LSM/Level Set	36 minutes	4%

These results indicate that there is indeed a very good improvement on the Level Set by first employing the LSM. Results comparable to the Level Set alone were obtained in a drastically reduced solution time. While these results are far preferable to the Level Set reconstruction, the LSM image taken was the most optimal contour taken at a single frequency.

However, it is difficult to determine what single frequency would be best for the reconstruction, and it is highly dependent on the true shape and size of the object. Therefore, being able to obtain a reasonable image using a frequency sweep would be desirable, as it removes the uncertainty of selecting a single frequency. The goal of this research is to determine a proper multiple frequency procedure for the LSM for the use in optimum hybridization with Level Set in the future.

B. THE LINEAR SAMPLING METHOD

Basic LSM

The LSM is an algorithm that seeks to solve the following far-field equation for the entire solution space [12]:

$$\int_{\Omega} u_{\infty}(\hat{x}, d) g_z(d) ds(d) = \Phi_{\infty}(\hat{x}, z) \quad (1)$$

The $u_{\infty}(\hat{x}, d)$ term is the far field scattering of the object that is causing the scattering, $g_z(d)$ is the indicator function that describes the shape of the object in the calculation space, and

$\Phi_{\infty}(\hat{x}, z)$ is the field created by a point source in vacuum and not dependent on the scattering of the object itself. The variable d is a vector defining the direction of propagation of the wave. LSM gathers the field data and attempts to solve for the indicator function $g_z(d)$ for every point in the solution space using the following summation [12]:

$$\|\vec{g}_z\|^2 = \sum_{p=1}^N \left(\frac{\sigma_p}{\alpha_z + \sigma_p^2} \right)^2 |(U^* b_z)_p|^2 \quad (2)$$

In this equation, $\|\vec{g}_z\|^2$ is the indicator function that is being solved. N is the number of points at which the field is measured. The σ_p terms are singular values achieved by the singular value decomposition of the field data as shown later in equations (5) and (6). The b_z term is a simple multiple of the point source far field data $\Phi_{\infty}(\hat{x}, z)$ as shown in equation (7). U^* is the inverse of one of the unitary matrices obtained from decomposing the measured field data in equation (5). For LSM, this number is part of the following limiting equation for the algorithm [12]:

$$N > 2ka \quad (3)$$

For this equation, a describes the radius of a circle that would surround the object being imaged. The k value is the wave number. Generally, a lower number of points is desired for reconstruction in order to be viable, so only the wave number is easily changed by adjusting the frequency at which the data is taken. For all of the simulations run for this research, the number of points is set constantly at 20. Thus, the upper limit of the frequency sweep is determined by equation (4).

$$f < \frac{Nc}{4\pi a} = \frac{4.77 \times 10^8 \text{ m/s}}{a} \quad (4)$$

The remaining terms of equation (2) are calculated by decomposing the measured field data, denoted in the following equation by A^{δ} .

$$A^\delta = USV^* \quad (5)$$

$$S_{p,p} = \sigma_p \quad (6)$$

The b_z value is a factor of the point source scattering dependent on the number of points.

$$(b_z)_l = \frac{N}{2\pi} \Phi_\infty(d_l, z) \quad (7)$$

Finally, the α_z value is a regularization parameter that seeks to fulfill equation (7). This is achieved by solving for the zero values of the summation in equation (8) [12].

$$\|A^\delta \vec{g}_z^\alpha - b_z\|^2 = \delta^2 \|\vec{g}_z^\alpha\|^2 \quad (8)$$

$$f(\alpha) = \sum_{p=1}^N \frac{\alpha^2 - \delta^2 \sigma_p^2}{(\sigma_p^2 + \alpha)^2} |(U^* b_z)|^2 \quad (9)$$

The variable δ is the noise of the system. In a practical case it can be found by using the system with no object present, but for the sake of the simulations it was obtained by adding a small amount of randomly generated values to the field data. However, this regularization does not drastically increase the capabilities of the LSM. A proof of concept for this matter will be shown in the results. Instead, the following assumption can be made for all cases:

$$\alpha_z = 0.01 \times \frac{2\pi}{k} \quad (10)$$

The final output of equation (2) results in a value of $\|\vec{g}_z\|^2$ for every point in the field. The logarithmic value of this array can be plotted as a contour field from which a value can be selected to represent the boundary of the object being imaged.

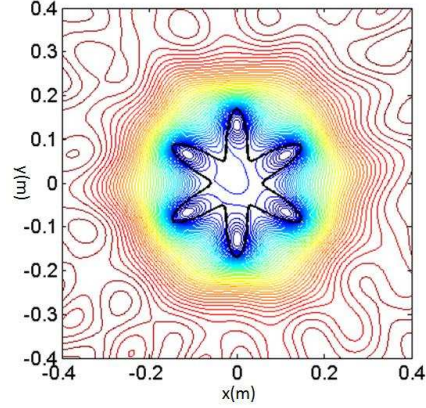


Fig. 1.2: Example of the contour field resulting from LSM.

Multiple Frequency LSM

Currently, there are several multiple frequency models of the LSM used for various implementations. One way of obtaining the multiple field value for the object is to normalize the $\|\vec{g}_z\|^2$ solved at each frequency and add them together [13].

$$g_{z,MF} = \sum_{f=1}^{N_F} \frac{\|g_{z,f}\|^2}{\max(\|g_{z,f}\|^2)} \quad (11)$$

Equation (9) outputs a grid of field values in the same manner as the normal LSM, and should provide a relatively reliable manner of getting an approximation of the image to input to the Level Set method. There have been other methods for multiple frequency reconstructions suggested [14],[15], but the one given here is the easiest to implement and has the fastest solution. The research being shown in this work seeks to determine the validity of such a method for the multiple frequency range.

II. APPROACH AND IMPLEMENTATION

A. SIMULATION PROCEDURE

Single Object: Six-pointed Star

The multiple frequency LSM is used for two different scenarios. The first scenario is a single six-pointed star as shown in figure 2.1. The computational domain as a whole stretches from -0.4 to 0.4 on both the x- and y-axes and is divided into a grid of 250x250 solution points at which equation (2) solves for the indicator function.

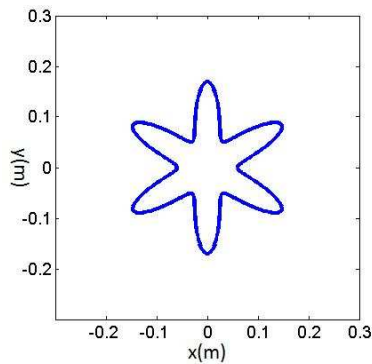


Fig. 2.1: Single object case: a six-pointed star

The 2D image was placed in a Method of Moments forward solver to obtain the field data at each of the measurement points. As stated before, the current version of the algorithm utilizes 20 equally spaced measurement angles. The measurements are then placed an LSM algorithm developed in MATLAB to solve for equation (2). The computer code can be found in Appendix A.

The first set of simulations was performed at five frequencies ranging from 0.5 to 2.5 GHz in steps of 0.5 GHz. However, as can be seen in the Results Section, the higher frequencies of this sweep caused deterioration in the LSM that prevented good results due to the restriction given by equation (3).

In order to keep the LSM within the proper frequency range for reliable reconstruction results, the sweep was changed to eleven frequencies from 0.5 to 1.5 GHz. The upper range for this sweep was selected using equation (4). If a circle of radius 0.25 is assumed to cover the entire object and 20 solution points are used, then the maximum frequency that can be utilized is 1.91 GHz.

This new sweep was used to obtain image reconstructions at each individual frequency as well as for the multiple frequency summation in order to compare the individual frequencies to the multiple frequency average. A separate program was then used to find the best fit contour from each of these fields as well as the percent error of that contour. The equation used to calculate the error is the same as was used in the single-frequency hybridization and is described in the following equation:

$$\%_{error} = \frac{\text{mismatched pixels between imaged and actual}}{\text{pixels in actual object}} \times 100\% \quad (12)$$

This equation was implemented with every single contour of the field and compared to each other. The contour with the least error was saved as the best-fit reconstruction. This process was performed both using the α_z parameterization and by using a constant α_z of 0.01. The results of both of these reconstructions were shown to give roughly the same level of accuracy, as will be shown in the Results Section, but using a constant α_z greatly reduced the simulation time.

Multiple Objects: Ellipse Pair

The second scenario researched using LSM was that of a pair of identical ellipses spaced an equal distance from the origin, as can be seen in figure 2.2. The solution space for this scenario was decreased to -0.12 to 0.12 on both the x and y axes to refine the solution around the objects.

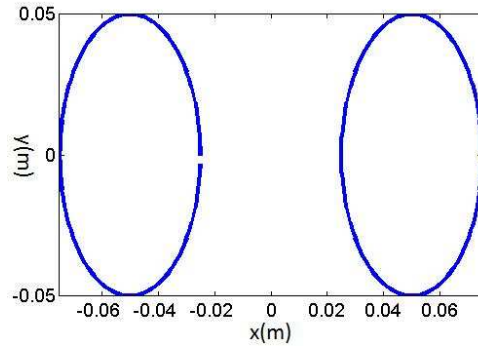


Fig. 2.2: Multiple objects case: a pair of ellipses

In this case the area containing the scattering objects can be placed within a circle of an approximate radius of 0.09 meters. Thus the maximum frequency available for the sweep would be 5.31 GHz. Therefore, the sweep chosen for this object's reconstruction was eleven points from 3.0 to 5.0 GHz. The same method for analyzing the first scenario was used for the second: the best fit contour was found for each frequency as well as the multiple frequency summation and then compared. Since the first scenario showed that α_z parameterization was not necessary for an accurate reconstruction, only a constant α_z of 0.01 was used for the second scenario.

Threshold Value Comparison

After comparing the best-fit contours of the two scenarios observed in this research, it was determined that simply looking at the best contour is not completely practical for a realistic scenario, since knowing the best contour is not possible without knowing the true shape of the object in the first place. Therefore, a more fair comparison between the values at each distinct frequency and the multiple frequency reconstruction should be drawn from selecting a single contour value to use for all frequencies. Thus for the three sets of results (first scenario with α_z solved, first scenario with α_z constant, and second scenario with α_z constant) the value of the

best fit contours was determined in order to establish a relative range of values that could be used for the comparisons.

From this comparison, several threshold values were chosen for the fields of each set of results. The error calculation code was adjusted to select the contour closest to that value for each frequency and the multiple frequency model, and calculate only the error of those contours. This would create a more fair comparison and should also show a reasonable threshold to use for future unknown images.

B. RESULTS

First Simulation Example

Figure 2.3 shows the results of the very first simulation set performed on the single object field data. The locator function field was calculated for five different frequencies ranging from 0.5 GHz to 2.5 GHz.

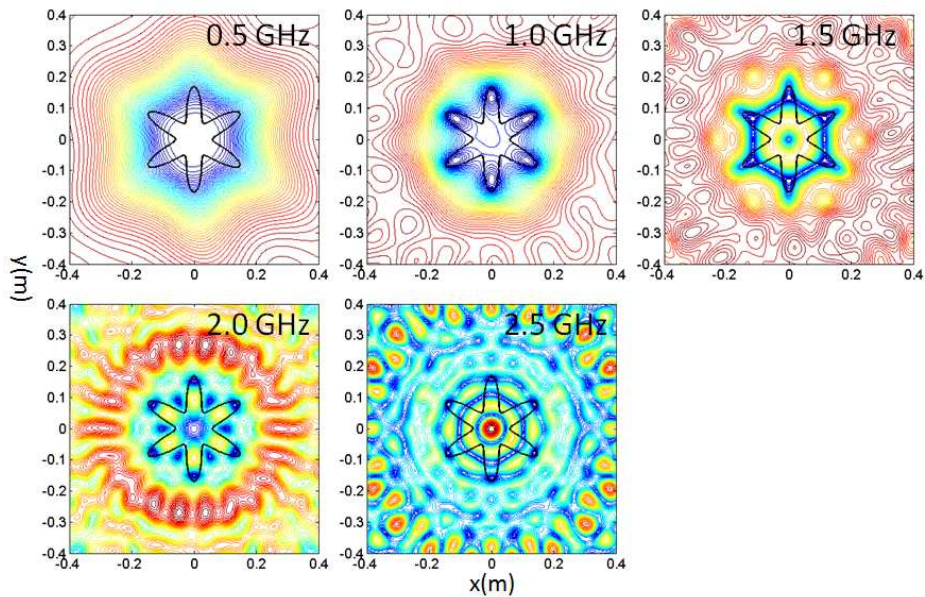


Fig. 2.3: First set of field reconstructions.

When observing the field reconstructions, there should be a trend from high values (red) in empty space to low values (blue) where the object is located. It can be seen here that for 0.5, 1.0, and 1.5 GHz the reconstruction follows the shape reasonably well, but for higher frequencies LSM fails to give good results because of the restriction given in equation (3) as described in the Procedures Section. Thus, a new sweep of eleven points from 0.5 to 1.5 GHz was performed for the same scenario.

Single Object Best-Fit Reconstructions

Figure 2.4 shows the fields reconstructed at each frequency and the multiple frequency model when α_z is solved, while figure 2.5 shows the best fit contours for each of these reconstructions. Figures 2.6 and 2.7 similarly show the field reconstructions and best fit contours when α_z is kept constant at $0.02\pi/k$ according to equation (10), respectively. In the field reconstruction images, the true object is shown by the black outline.

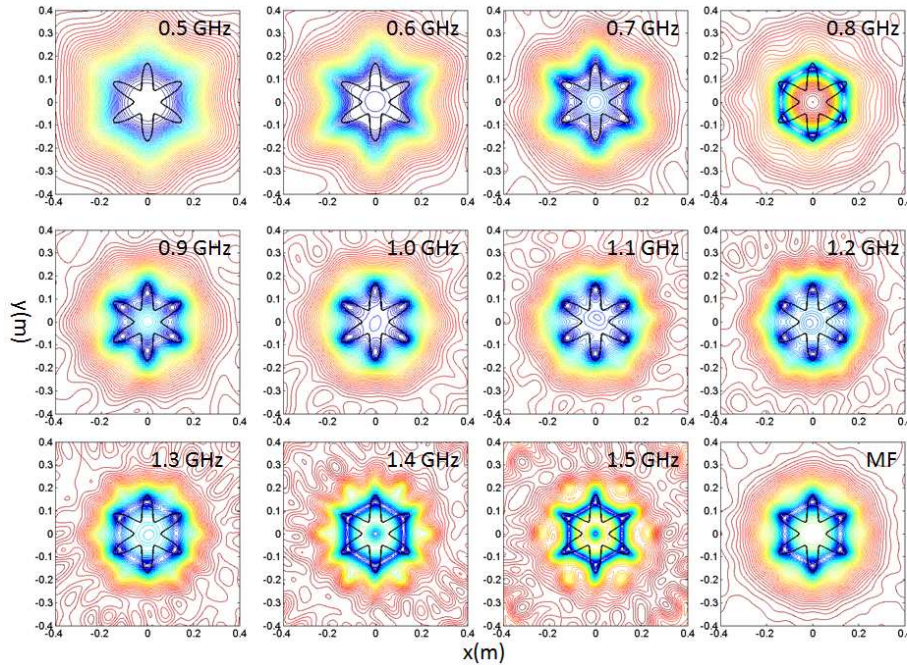


Fig. 2.4: Single object reconstruction at each frequency with α solved.

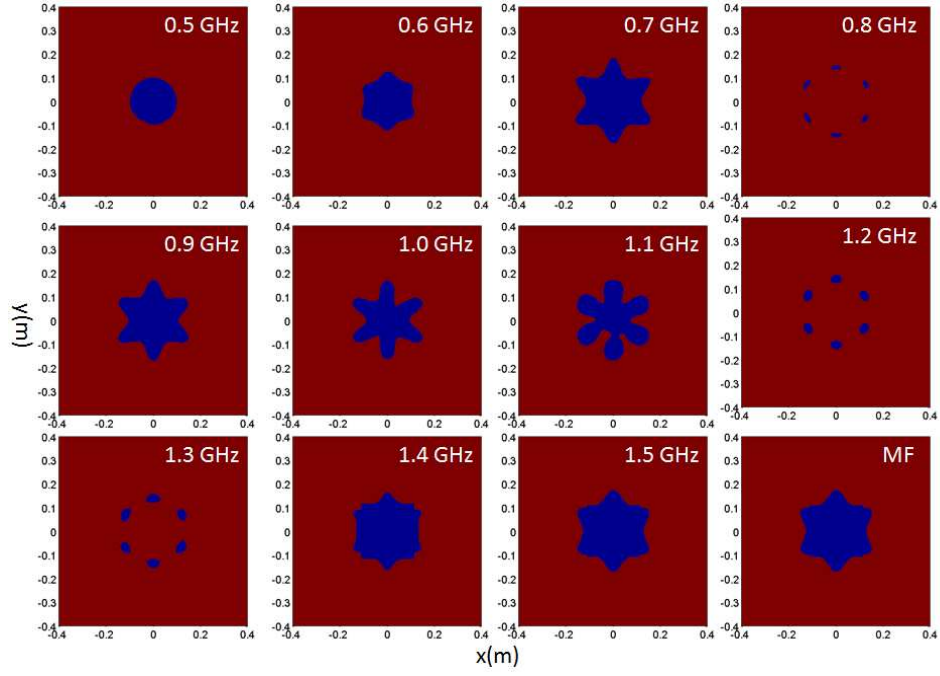


Fig. 2.5: Single object best-fit contours selected at each frequency with α solved.

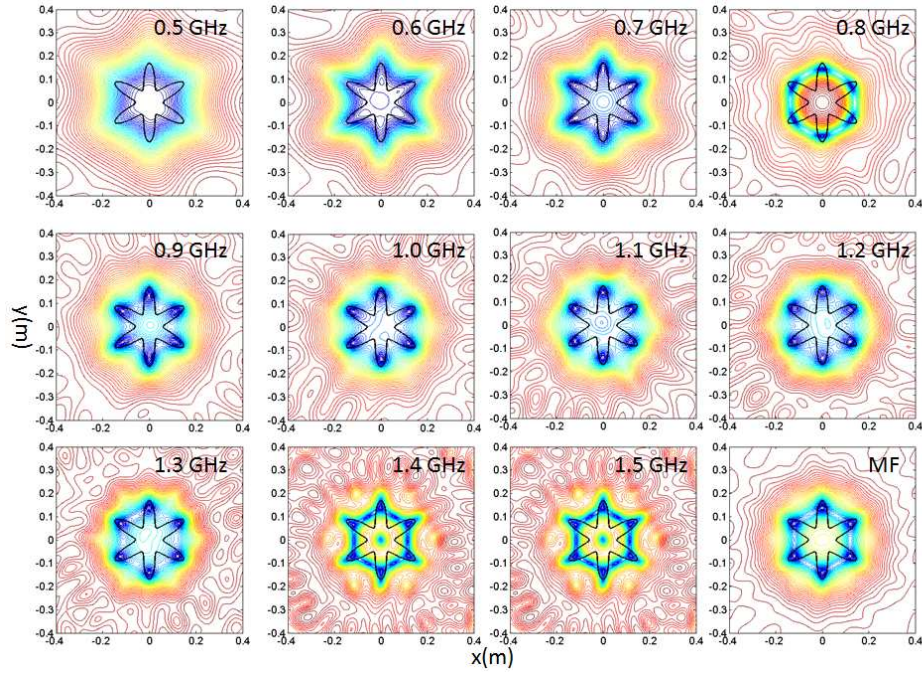


Fig. 2.6: Single object reconstruction at each frequency with α constant.

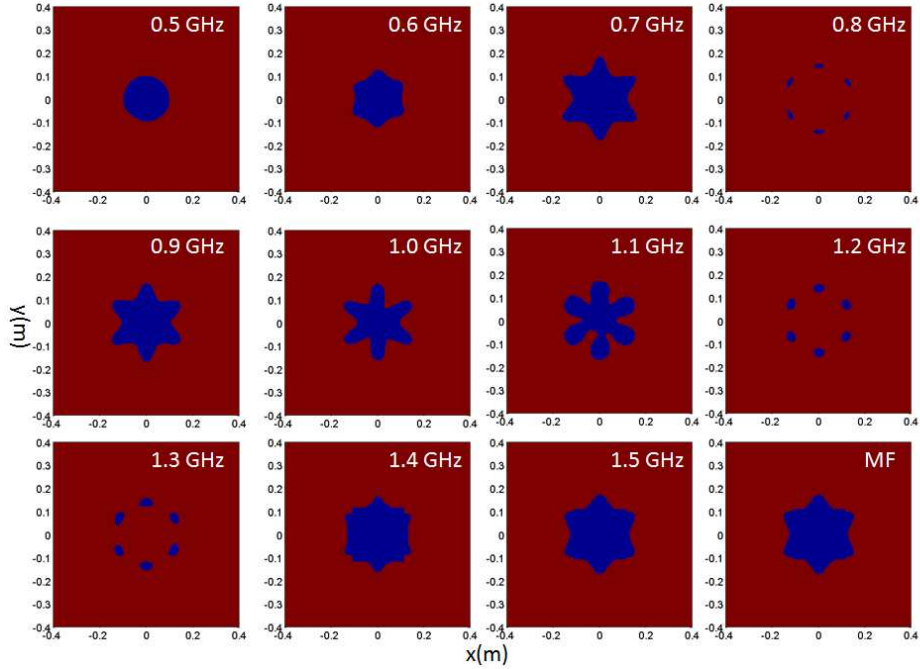


Fig. 2.7: Single object best-fit contours selected at each frequency with α constant.

Using equation (12), it is possible to obtain a percent error for each of the best fit contours for both cases of the first scenario, as shown in figure 2.8.

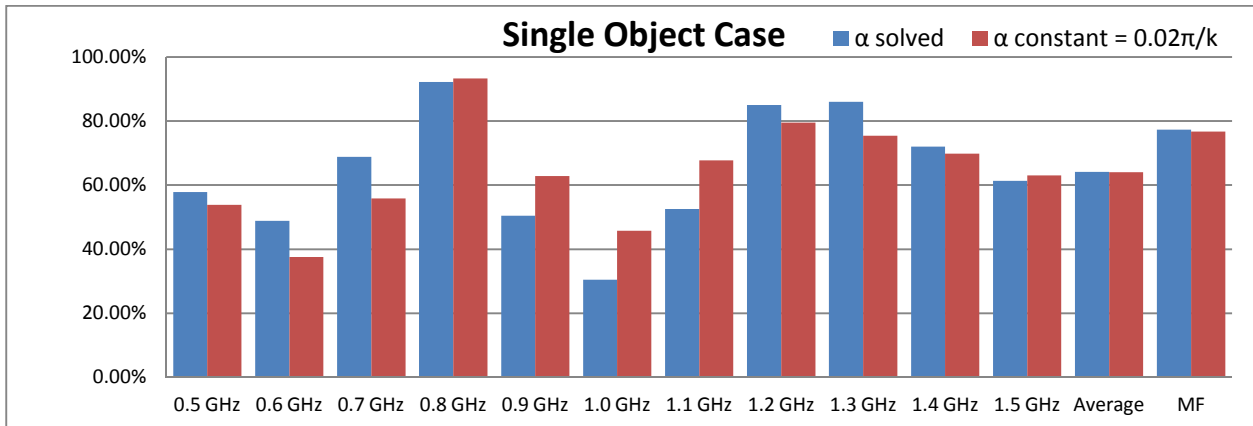


Fig. 2.8: Percent error graph for single object reconstruction.

In both cases the multiple frequency model, denoted by the text “MF” in the associated images, gives a reasonable reconstruction for the best fit model. One thing to note about the reconstruction is that at certain frequencies such as 0.8, 1.2, and 1.3 GHz, the best-fit contour

selected is not a viable reconstruction. This is likely due to the fact that the LSM fails when the frequency is at one of the object's eigenvalues, which is heavily dependent on the shape and size of the object itself. Other frequencies such as 1.0 and 1.1 GHz gave very reliable reconstructions. However the goal of the LSM in hybridization is not to get a perfect reconstruction, but to obtain a reasonable first guess for the more rigorous Level Set algorithm. For such a purpose, these multiple frequency models give viable results.

It should also be noted that the differences between solving for α_z or leaving it constant are fairly negligible, especially when compared across a range of frequencies. The average error across the frequency sweep is essentially the same in both cases, with the difference in the multiple frequency reconstruction being similarly negligible. However, the solution time when assuming α_z to be constant is far less than solving for α_z . Thus, for the multiple object reconstruction only the constant α_z case will be considered.

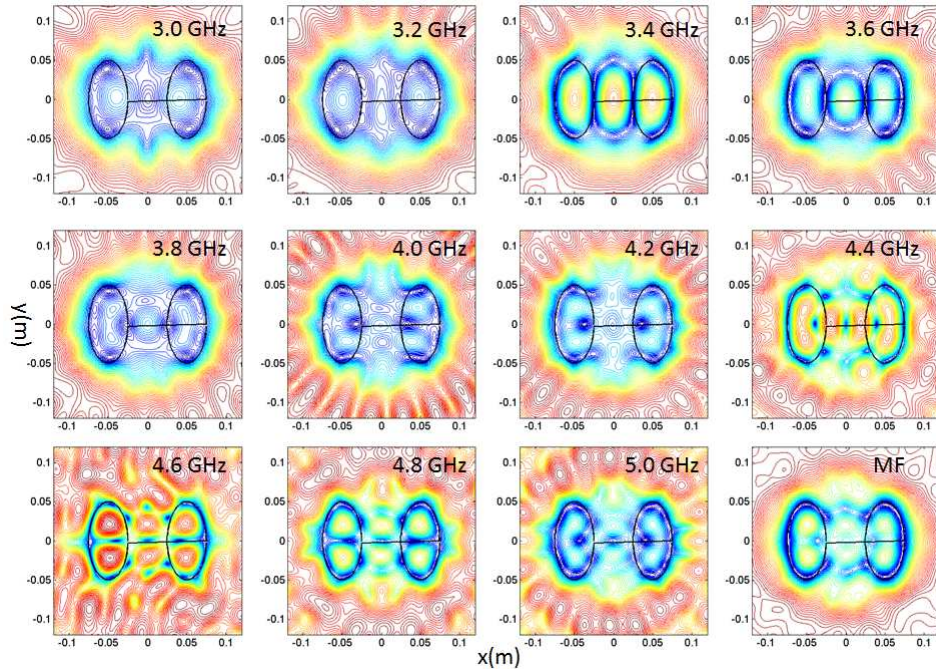


Fig. 2.9: Second scenario reconstruction at each frequency with α constant.

Multiple Object Best-Fit Reconstruction

Figures 2.9 and 2.10 show the field reconstruction and best-fit contours of the scenario involving the two ellipses. Figure 2.11 shows the percent error for each of the best fit contours.

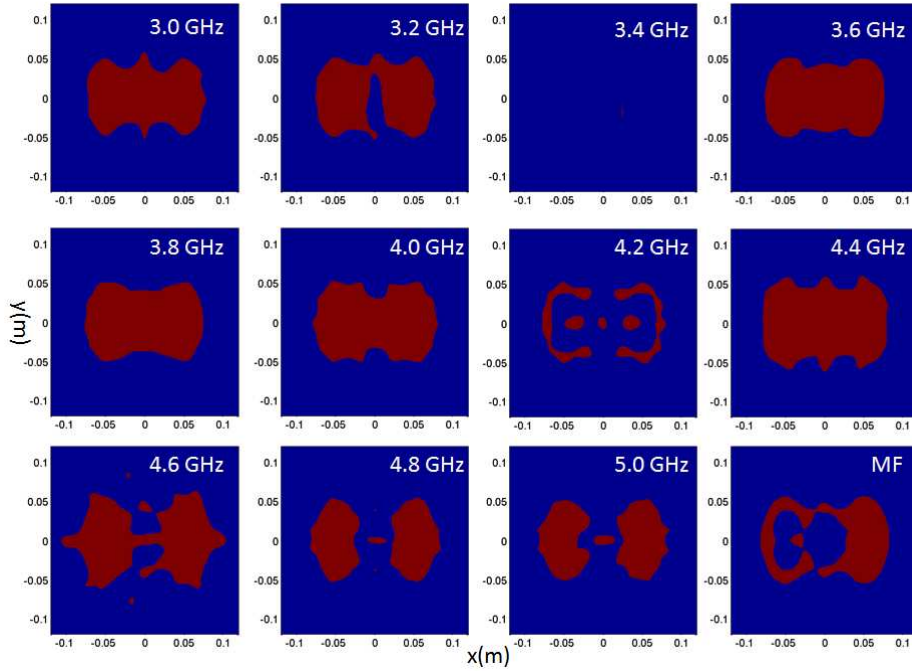


Fig. 2.10: Multiple object best-fit contours selected at each frequency with α constant.

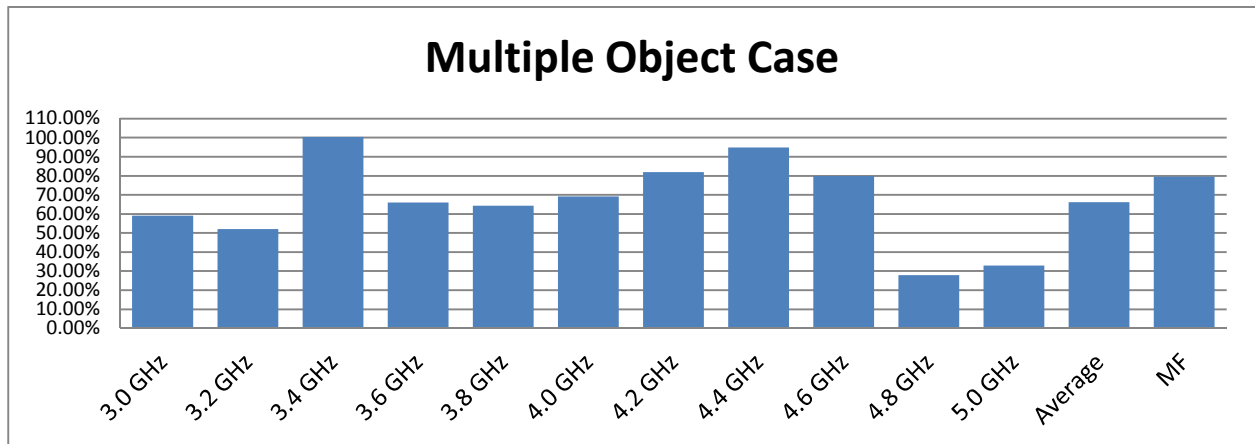


Fig. 2.11: Percent error graph for the multiple object reconstruction.

As in the scenario with the six-pointed star, the multiple frequency LSM produced a fairly reasonable contour of the two ellipses that. While the best fit contour was not as good as an average contour taken from the best fit reconstructions, it does prevent the possibility of accidentally selecting a poor reconstruction such as 3.4 or 4.2 GHz. This makes the multiple frequency technique a viable method for obtaining an initial guess for hybridization.

The Same Threshold Value Comparison

However, in a realistic case it is not possible to compare the error of one best-fit contour to another, since the true shape of the object is required to determine the best-fit contours, and the purpose of the research is to find the unknown object in the first place. Thus, a more fair comparison must be drawn by comparing all of the fields at the same values. Table 2 shows the different thresholds that were originally selected by the error-calculation code to outline the best fit contours.

Table 2: Best-fit contours and their thresholds

<u>Frequency</u>	<u>star (solved alpha)</u>		<u>star (constant alpha)</u>		<u>ellipses</u>	
	<u>contour</u>	<u>value</u>	<u>contour</u>	<u>value</u>	<u>contour</u>	<u>value</u>
0.5 GHz	2	-1.236	5	-1.007	9	-0.699
0.6 GHz	1	-1.139	4	-0.900	9	-0.667
0.7 GHz	7	-0.862	10	-0.559	1	-0.967
0.8 GHz	5	-1.040	10	-0.633	11	-0.610
0.9 GHz	10	-0.919	10	-0.724	10	-0.682
1.0 GHz	8	-0.992	13	-0.558	12	-0.591
1.1 GHz	11	-0.882	15	-0.512	11	-0.645
1.2 GHz	5	-1.182	9	-0.781	23	-0.079
1.3 GHz	6	-1.138	15	-0.482	30	0.157
1.4 GHz	9	-1.031	13	-0.607	16	-0.443
1.5 GHz	11	-0.935	15	-0.448	14	-0.587
Average	6.82	-1.032	10.82	-0.656	13.27	-0.528
MF	12	-0.618	15	-0.644	17	-0.389

The contour number is simply the point on the logarithmic scale at which the selected threshold value occurs. From this table, a threshold range was selected to observe how well the MF reconstruction behaves at a certain value in comparison to the individual frequencies at that value. The thresholds selected were -0.1, -0.3, -0.5, and -0.7. These thresholds were applied to all three sets of field values to obtain a set of contours. Only the numerical error results and the multiple frequency comparison will be shown here. In each of the following images, “Th” indicates the threshold value for the data. The full sets of contours can be seen in Appendix B.

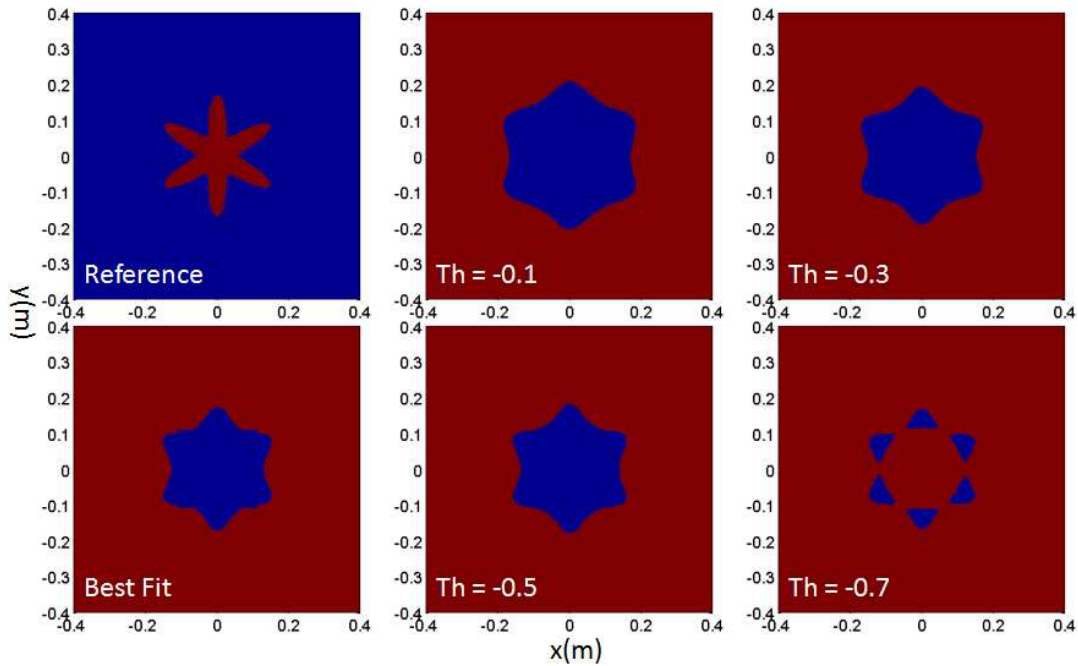


Fig. 2.12: Single object comparison of multiple frequency reconstructions at each threshold value when solving for α_z . “Th” is the threshold value.

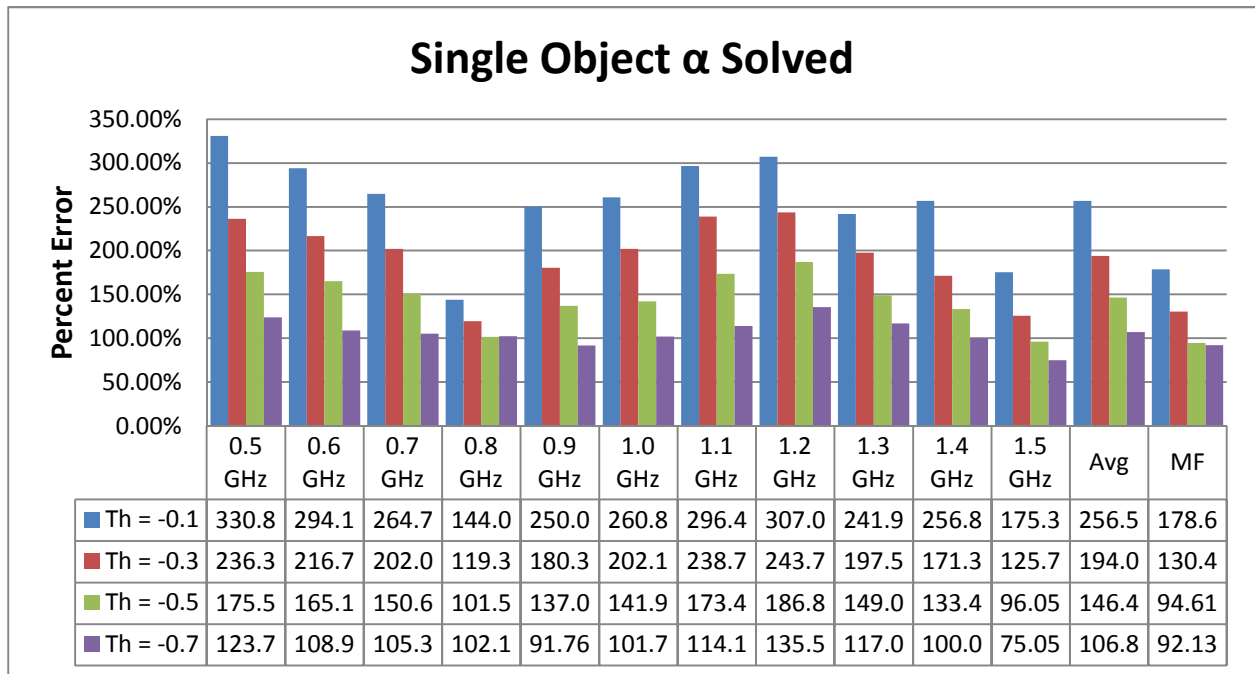


Fig. 2.13: Error comparison of single object at each threshold value when solving for α_z .

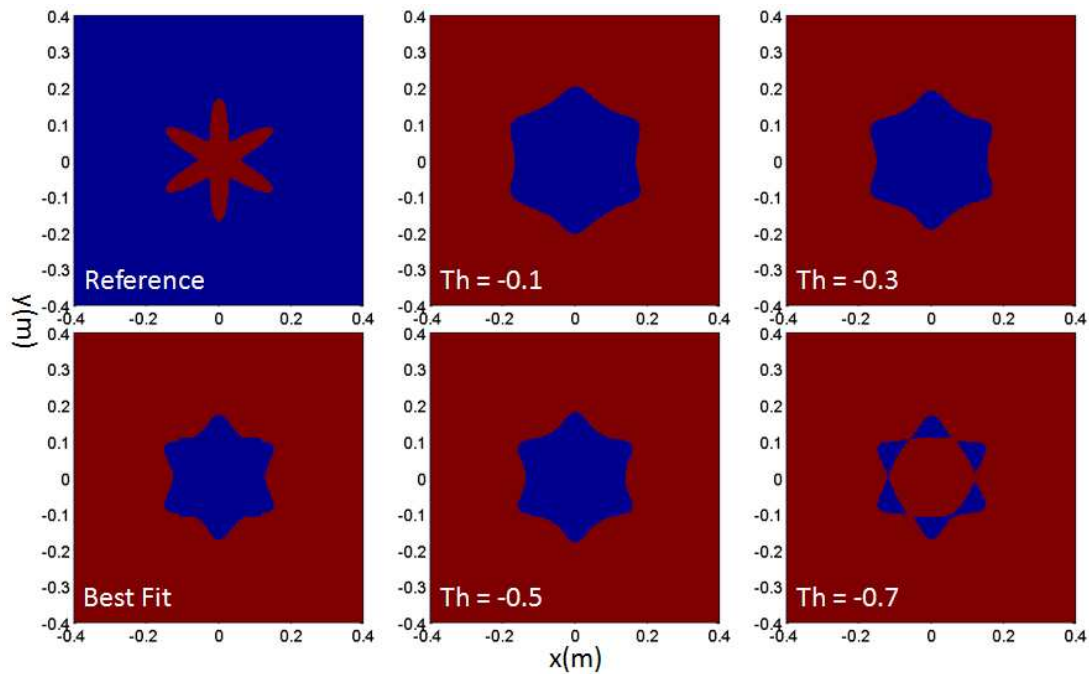


Fig. 2.14: Single object comparison of multiple frequency reconstructions at each threshold value when using constant α_z .

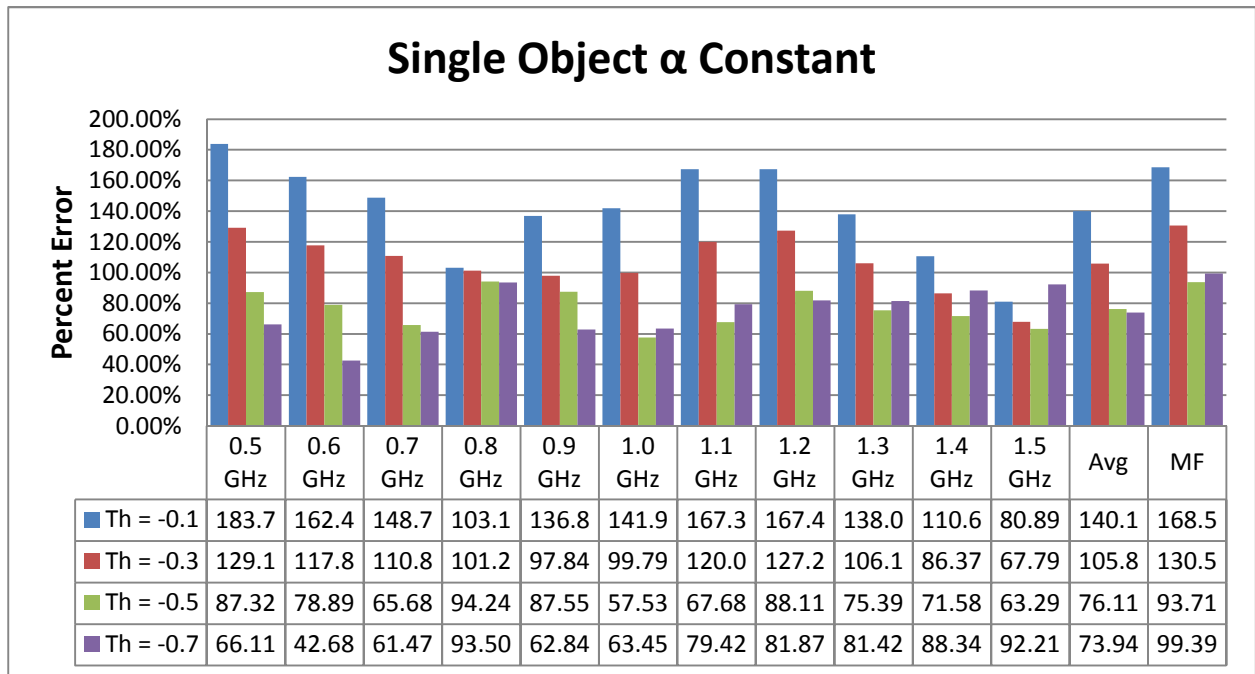


Fig. 2.15: Error comparison of single object at each threshold value when using constant α_z .

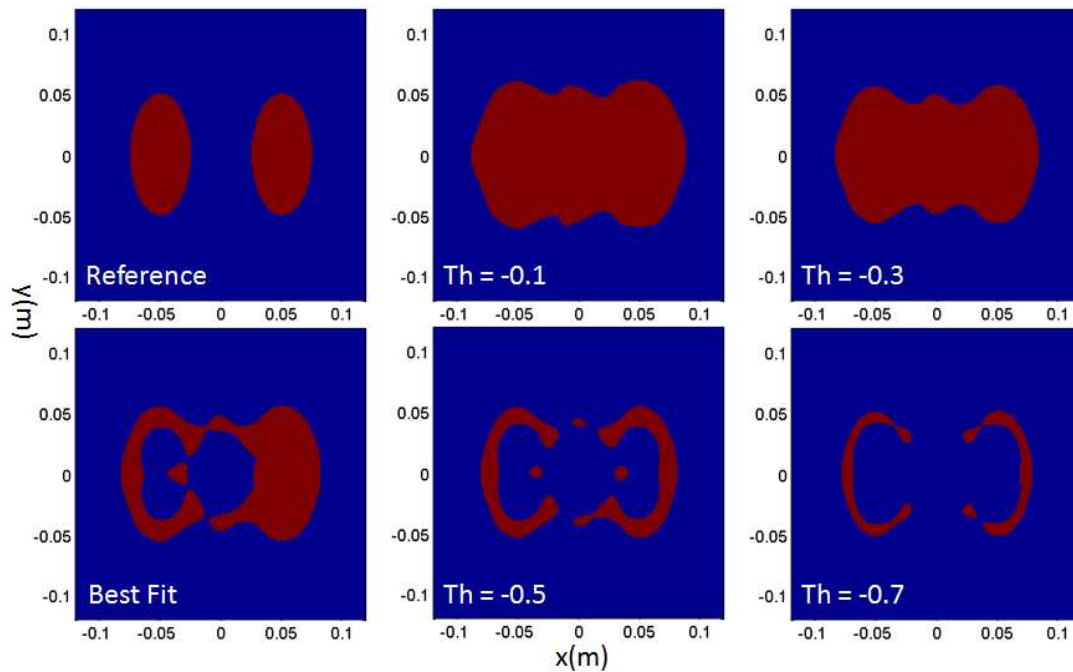


Fig. 2.16: Multiple object comparison of multiple frequency reconstructions at each threshold value when using constant α_z .

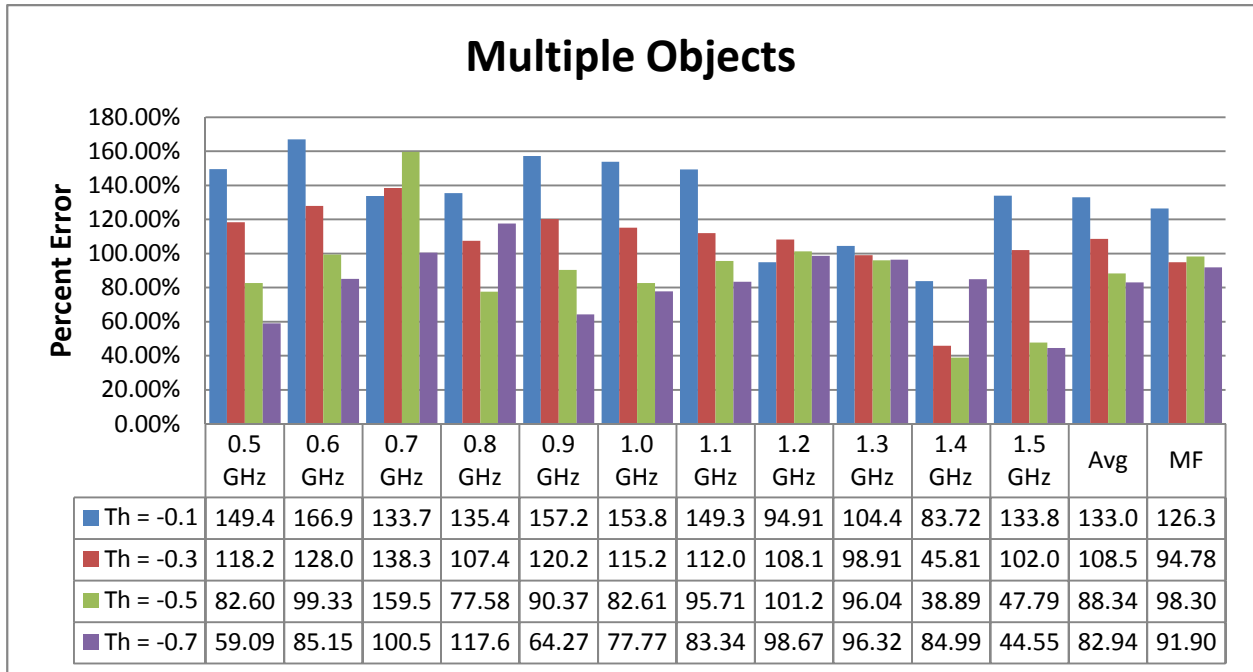


Fig. 2.17: Error comparison of multiple objects at each threshold value when using constant α_z

In all three of the cases observed, the multiple frequency model continued to have a reasonable reconstruction compared to the individual frequencies at the same threshold value. In particular, it seems that threshold values between -0.3 and -0.5 would give reliable reconstructions for all cases. However, these values cannot be assumed for cases not mentioned here until more research has been conducted.

Note that in these cases there are error values greater than 100%. This is because the number of mismatched pixels is compared to the number of pixels in the actual object rather than the solution space as a whole. Thus, it is possible to have more mismatched pixels than exist in the actual object. This could be avoided by using the total pixels in the solution space instead of the pixels in the object, but the percentage comparisons are still fair since all percentages are drawn from the same method.

III. CONCLUSIONS AND FUTURE RESEARCH

This research has shown that there is a simple yet reliable method for taking a multiple frequency reconstruction of the LSM. While this method does not necessarily give a better reconstruction than some specific frequencies, it is a reliable method of avoiding the frequencies in a sweep that provide very poor reconstructions. This multiple frequency LSM works equally well for a single complicated object or multiple objects. The true test of the validity of the method will have to come from testing the hybridization method with the multiple frequency output of the algorithm.

Other future research will involve attempting to expand the multiple frequency method to a wider number of cases. First of all, it would be useful to determine that the LSM still works for asymmetrical cases. Thus, research will be conducted into reconstruction of objects with less regular distribution or objects not centered at the origin of the xy-plane.

Additionally, the current version of the LSM is only used for perfect electric conductor objects, and LSM is not dependent on the permittivity of the object, so it is important to see if a reconstruction can be gathered from objects of lower permittivity. Thus, in order to perform any reconstruction, a method of solving for the scattered fields of 2D dielectric objects must first be determined. Then the LSM will be utilized to test if reconstruction results can be obtained in short CPU time. Finally, for practical applications it will be necessary to upgrade the LSM algorithm to handle 3D reconstructions so that multiple frequency and hybridization with the Level Set algorithm can be tested.

IV. REFERENCES

- [1] A. M. Hassan, M. R. Hajihashemi and M. El-Shenawee, "Inverse Scattering Shape Reconstruction of 3D Bacteria Using the Level Set Algorithm," Progress In Electromagnetics Research B, vol. 39, pages 39-53, 2012.
- [2] M. R. Hajihashemi and M. El-Shenawee, "Level Set Algorithm for Shape Reconstruction of Non-Overlapping Three-Dimensional Penetrable Targets," IEEE Trans. Geosci. & Rem. Sens., vol.50, no.1, pp.75-86, Jan. 2012.
- [3] M. R. Hajihashemi and M. El-Shenawee, "Inverse scattering of three-dimensional PEC objects using the Level Set method," Progress In Electromagnetics Research, vol. 116, 23-47, 2011.
- [4] M. R. Hajihashemi and M. El-Shenawee, "The Level Set Shape Reconstruction Algorithm Applied to 2D PEC Targets Hidden Behind a Wall," Progress In Electromagnetics Research B, vol. 25, pp. 131–154, 2010.
- [5] M. R. Hajihashemi and M. El-Shenawee, "High Performance Computing of the Level Set Reconstruction Algorithm," Journal of Parallel and Distributed Computing JPDC, vol. 70, pp. 671-679, June 2010.
- [6] M. R. Hajihashemi and M. El-Shenawee, "TE versus TM for the Shape Reconstruction of 2-D PEC Targets using the Level Set Algorithm," IEEE Trans. Geosci. & Rem. Sens., vol. 48, no. 3, pp. 1159-1168, March 2010.
- [7] D. A. Woten, M. R. Hajihashemi, A. M. Hassan and M. El-Shenawee, "Experimental Microwave Validation of the Level Set Reconstruction Algorithm," IEEE Trans. Antennas and Propag., vol. 58, no. 1, pp. 230-233, Jan 2010.
- [8] A. Hassan, M. Hajihashemi, M. El-Shenawee, A. Al-Zoubi, A. Kishk, "Drift De-noising of Experimental TE Measurements for Imaging of 2D PEC Cylinder using the Level Set Algorithm," IEEE Antennas and Wireless Propagation Letters, vol. 8, pp. 1218-1222, 2009.
- [9] A. M. Hassan, T. Bowman, and M. El-Shenawee, "The Linear Sampling Method for the Acceleration of the Level Set Algorithm," Proceedings of the IEEE International Symposium on Antennas and Propagation and USNC/URSI National Radio Science Meeting, Chicago, USA, July 8-13, 2012.
- [10] M. Pastorino. "Hybrid Reconstruction Techniques for Microwave Imaging Systems." 2010 IEEE International Conference on Imaging Systems and Techniques (IST). Pages 198-203. 1-2 July 2010.
- [11] M. Brignone *et al.* "A Hybrid Approach to 3D Microwave Imaging by Using Linear Sampling and ACO." IEEE Transactions on Antennas and Propagation, vol. 56, no. 12, pages 3224-3232, October 2008.

[12] F. Cakoni, D. Colton, and P. Monk, *The Linear Sampling Method in Inverse Electromagnetic Scattering*, Society for Industrial and Applied Mathematics, 2011.

[13] I. Catapano, C. Crocco, and T. Isernia, "An improved Linear Sampling Method for location and shape reconstruction of 3D buried targets," *Proceedings of Geoscience and Remote Sensing Symposium, 2007. IGARSS, 2007. IEEE International*. Pages 4810-4813. 23-28 July 2007.

[14] H.F. Alqadah, J. Parker, M. Ferrara, and H. Fan. "Space-frequency sparse regularization for the linear sampling method." 2011 International Conference on Electromagnetics in Advanced Applications (ICEAA). Pages 421-424. 12-16 September 2011.

[15] B. Guzina, F. Cakoni, and C. Bellis. "On the multi-frequency obstacle reconstruction via the linear sampling method." *Inverse Problems* volume 26, issue 12, 125005. December 2010.

APPENDIX A: MATLAB CODE

Linear Sampling Method code

```
close all
clc
clear

% Code for the LSM method for 2D PEC targets. The formulation is based on
%Chapter 1 in "F. Cakoni, D. Colton, P. Monk, The Linear Sampling Method in
%Inverse Electromagnetic Scattering, Society for Industrial and Applied
%Mathematics, 2011"

% Set number of frequencies
fs=11;

% Set frequency values
f=[0.5e9 0.6e9 0.7e9 0.8e9 0.9e9 1e9 1.1e9 1.2e9 1.3e9 1.4e9 1.5e9];

% Set scattered field file names
fieldfile=['m_lsm_f_low_1.dat '; 'm_lsm_f_low_2.dat '; 'm_lsm_f_low_3.dat ';
'm_lsm_f_low_4.dat '; 'm_lsm_f_low_5.dat '; 'm_lsm_f_low_6.dat '; 'm_lsm_f_low_7.dat
'; 'm_lsm_f_low_8.dat '; 'm_lsm_f_low_9.dat '; 'm_lsm_f_low_10.dat';
'm_lsm_f_low_11.dat'];

% Set scattered field with noise file names
fieldsfile=['mn_lsm_f_low_1.dat '; 'mn_lsm_f_low_2.dat '; 'mn_lsm_f_low_3.dat ';
'mn_lsm_f_low_4.dat '; 'mn_lsm_f_low_5.dat '; 'mn_lsm_f_low_6.dat ';
'mn_lsm_f_low_7.dat '; 'mn_lsm_f_low_8.dat '; 'mn_lsm_f_low_9.dat ';
'mn_lsm_f_low_10.dat'; 'mn_lsm_f_low_11.dat'];

% k: is the propagation factor
k= (2*pi/3e8)*f

% nxy: is the number of pixels in the x and y direction. The imaging domain
%will be composed of nxy*nxy pixels
nxy=250;

% N: is the number of incident and receiver directions
N=20;

% Initiate field arrays
% GMF: Multiple frequency average values of the GG function
% amf: Multiple frequency average values of solved alpha
GMF=zeros(nxy,nxy);
amf=zeros(nxy,nxy);

tic
for i=1:fs

    %m: is the file with the scattered field calculated using the MOM code of
    %Reza. The file is formatted such that the real part is in the 1st column
    %and the imaginary part is in the 2nd column. The scattered fields are
    %arranged into a matrix "u" where the rows represent the different receiver
    %angles and the column represent the different incident angles

    m=load(fieldfile(i,:));
    mm=m(:,1)-j*m(:,2);
    mm=mm*j;
    u=reshape(mm,N,N);
```

```

% m: is the file with the scattered field+NOISE calculated using the MOM code of
%Reza. The file is formatted such that the real part is in the 1st column
%and the imaginary part is in the 2nd column. The scattered fields are
%arranged into a matrix "As" where the rows represent the different receiver
%angles and the column represent the different incident angles

m=load(fieldsfile(i,:));
mm=m(:,1)-j*m(:,2);
mm=mm*j;
As=reshape(mm,N,N);

% s is a factor used in the regularization. It is calculated by taking the
%norm of the difference between the scattered field+Noise and the exact
%scattered field. In real measurements s can be estimated by measuring the
%level of noise

s=norm(As-u);

% x&y: limits of the imaging domain
x=linspace(-0.4,0.4,nxy);
y=linspace(-0.4,0.4,nxy);

% xa: receiver angles
% h: step in receiver angles
xa(1:N,1)=0:2*pi/N:2*pi-2*pi/N;
h=2*pi/N;

options = optimset('TolX',1e-14);
% GG: is the LSM unknown that we solve for at each pixel in the domain
% a: is the regularization parameter calculated at each pixel in the domain
% a: can also indicate to the shape of the object.
GG=zeros(nxy,nxy);
a=zeros(nxy,nxy);

% The following is the main for loop that scans each pixel in the domain
%calculating "GG" and "a" at each pixel.

frequency=i
for ii=1:nxy
    ii
    for jj=1:nxy
        % is the far field pattern due to a point source located at the
        %auxiliary point x(jj),y(ii).
        pinf=exp(j*pi/4)/sqrt(8*pi*k(i))*exp(-
j*k(i)*(cos(xa)*x(jj)+sin(xa)*y(ii)));
        bz=pinf/h;

        [U,S,V]=svd(As);
        % "fzero" is used to get the optimum regularization parameter by
        %solving the equation in "fun_a". A constant can be used to equal effect.

        %% a(ii,jj)=fzero(@(a) fun_a(a,S,s,N,U'*bz),[0.001 1000],options);
        a(ii,jj)=.01*2*pi/k(i);

        ub=U'*bz;
        for ig=1:N

GG(ii,jj)=GG(ii,jj)+((S(ig,ig)/(a(ii,jj)+S(ig,ig)^2))^2)*abs(ub(ig))*abs(ub(ig));
        end
    end
end
end

```

```

% The multiple frequency array takes in the normalized value of each field
%and sums them together.
GGS(i, :, :) = GG;
GMF = GMF + GG / max(max(abs(GG)));
amf = amf + a / max(max(a));

% A plot of the contours of GG indicating the shape of the object at each
frequency

figure
CC = contour(x, y, log10(GG), 50);
axis image
hold on
kk = load('contours.txt');
plot(kk(:, 1), kk(:, 2), 'k-', 'linewidth', 2)
axis([-0.4 0.4 -0.4 0.4])
set(gca, 'fontsize', 14)
set(gcf, 'color', [1 1 1])

end
toc

% A plot of the contours of GMF indicating the shape of the object averaged
%across the frequency sweep

figure
CC = contour(x, y, log10(GMF), 50);
axis image
hold on
kk = load('contours.txt');
plot(kk(:, 1), kk(:, 2), 'k-', 'linewidth', 2)
axis([-0.4 0.4 -0.4 0.4])
set(gca, 'fontsize', 14)
set(gcf, 'color', [1 1 1])

save ig_star_f_2

```

Alpha Parameterization

```

% The equation from which the regularization parameter "a" is calculated.
% "a" is calculated by finding the POSITIVE zero of "y"
function y = fun_a(a, S, s, N, ub)

y = 0;
for i = 1:N
    y = y + (a^2 - s^2 * S(i, i)^2) / ((S(i, i)^2 + a)^2 * abs(ub(i)) * abs(ub(i)));
end

```

Percent Error Calculator

```
% function CC_opt=ig_opt_CC(CC,x,y,N)

close all
clc
clear;

load ig_star_f;
G2=zeros(nxy,nxy);
G2=GMF;
CC=contour(x,y,log10(G2),50);

I=find(CC(2,:)>1);

DD=CC;
similar=zeros(50,length(CC));
for i=1:50
    levels=DD(1,I);
    in=DD(1,2);
    if(i==1)
        I1=find(DD(1,)==DD(1,1));
        similar(i,1:length(I1))=I1;
        lens(i)=length(find(DD(1,)==DD(1,1)));
        DD(2,I1)=0;
        ls(i)=DD(1,1);
    else
        I2=find(DD(2,:)>1);
        I22=find(DD(1,)==DD(1,I2(1)));
        similar(i,1:length(I22))=I22;
        lens(i)=length(find(DD(1,)==DD(1,I2(1))));
        DD(2,I22)=0;
        ls(i)=DD(1,I2(1));
    end
end

end

inb=zeros(nxy*nxy,1);
inc=zeros(length(x),length(x));
kk=load('contours.txt');

for i=1:l
    % x=linspace(-0.3,0.3,nxy);
    % y=linspace(-0.3,0.3,nxy);

    [X,Y]=meshgrid(x,y);

    IN = inpolygon(X,Y,kk(1+(i-1)*500:i*500,1),kk(1+(i-1)*500:i*500,2));

    ina=reshape(IN,length(x)*length(x),1);

    I=find(ina);

    inb=zeros(nxy*nxy,1);
    inb(I)=1;
    inc=inc+reshape(inb,length(x),length(x));
    figure(1)
    plot(kk(1+(i-1)*500:i*500,1),kk(1+(i-1)*500:i*500,2),'linewidth',3)
    hold on
end
```

```

axis image
axis([-0.3 0.3 -0.3 0.3])
set(gcf,'color',[1 1 1])
set(gca,'fontsize',14)

[III,JJJ]=find(inc>1);
if(length(III)>0)
    inc(III,JJJ)=1;
end
figure
pcolor(x,y,inc)
axis image
shading flat
set(gcf,'color',[1 1 1])
set(gca,'fontsize',14)

incc=zeros(length(x),length(x),100);
[A,B]=size(inc);

loc2=[0 0];
for i=1:50
    i
    for j=1:lens(i)
        similar(i,j);
        LL=CC(2,similar(i,j));
        xv=CC(1,similar(i,j)+1:similar(i,j)+LL);
        yv=CC(2,similar(i,j)+1:similar(i,j)+LL);
        %         plot(xv,yv)
        %         hold on
        %
        IN = inpolygon(X,Y,xv,yv);

        ina=reshape(IN,length(x)*length(x),1);
        inb=zeros(length(ina),1);
        I=find(ina);

        inb(I)=1;
        incc(:,:,i)=incc(:,:,i)+reshape(inb,length(x),length(x));
        clear III JJJ
        [III,JJJ]=find(incc(:,:,i)>1);
        if(length(III)>0)
            incc(III,JJJ,i)=1;

            [al,bl]=size(loc2);
            [ai,bi]=size(III);
            loc2(al+1:al+ai,1)=III;
            loc2(al+1:al+ai,1)=JJJ;
        end
    end
    min_err(i)=sum(sum(abs(incc(:,:,i)-inc)));
end

min_err=min_err/sum(sum(inc))*100;

figure
contour(x,y,log10(G2),50);
axis image
hold on
% kk=load('contours.txt');
% plot(kk(:,1),kk(:,2),'k-','linewidth',2)
axis([-0.4 0.4 -0.4 0.4])
set(gca,'fontsize',14)

```

```
set(gcf,'color',[1 1 1])

[mmm,I_min]=min(min_err);
figure
pcolor(x,y,-incc(:,:,I_min))
axis image
shading flat
set(gcf,'color',[1 1 1])
set(gca,'fontsize',14)

display(['Error of best fitting contour is=' num2str(mmm) '%'])
display(['Value of minimum threshold is ' num2str(1s(I_min)) ' contour number '
num2str(I_min)])

save igsd_star_f
```

APPENDIX B: THRESHOLD SWEEP CONTOUR SELECTIONS

First scenario: six-pointed star, α solved

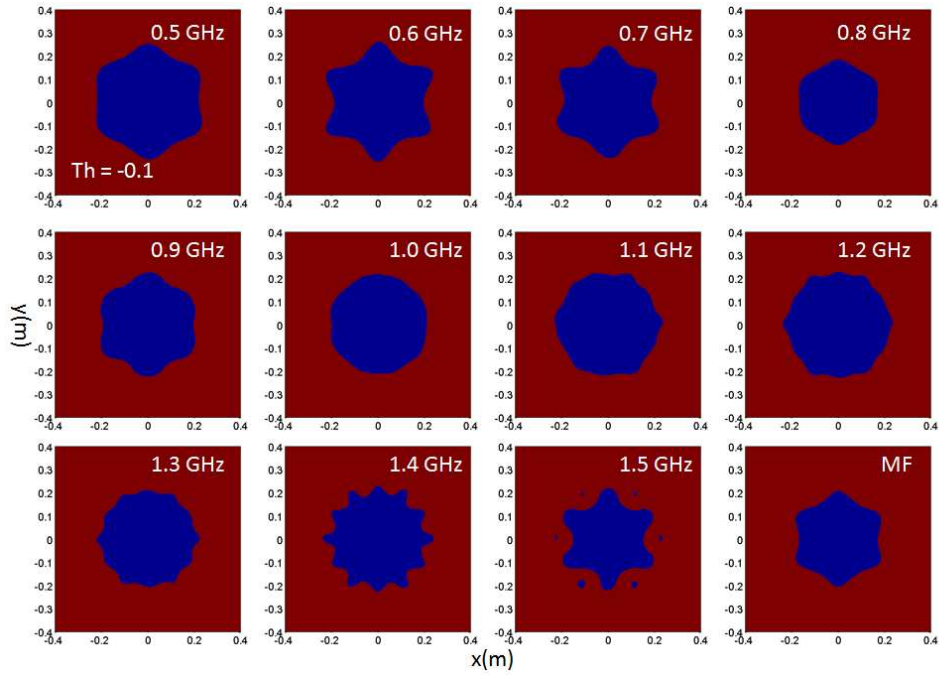


Fig. B.1: Single object contours when solving for α_z with threshold of -0.1.

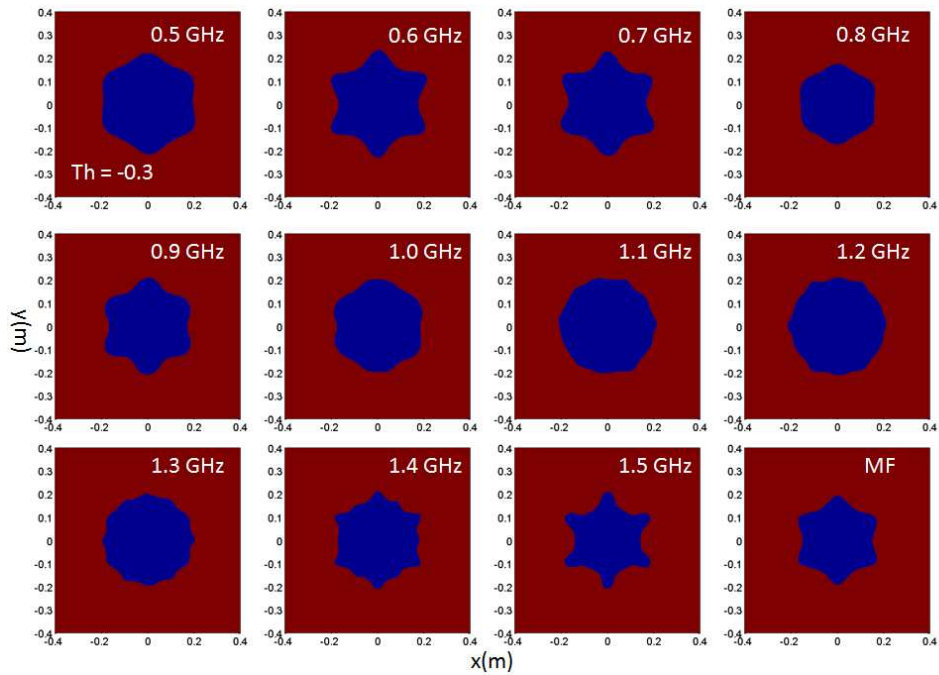


Fig. B.2: Single object contours when solving for α_z with threshold of -0.3.

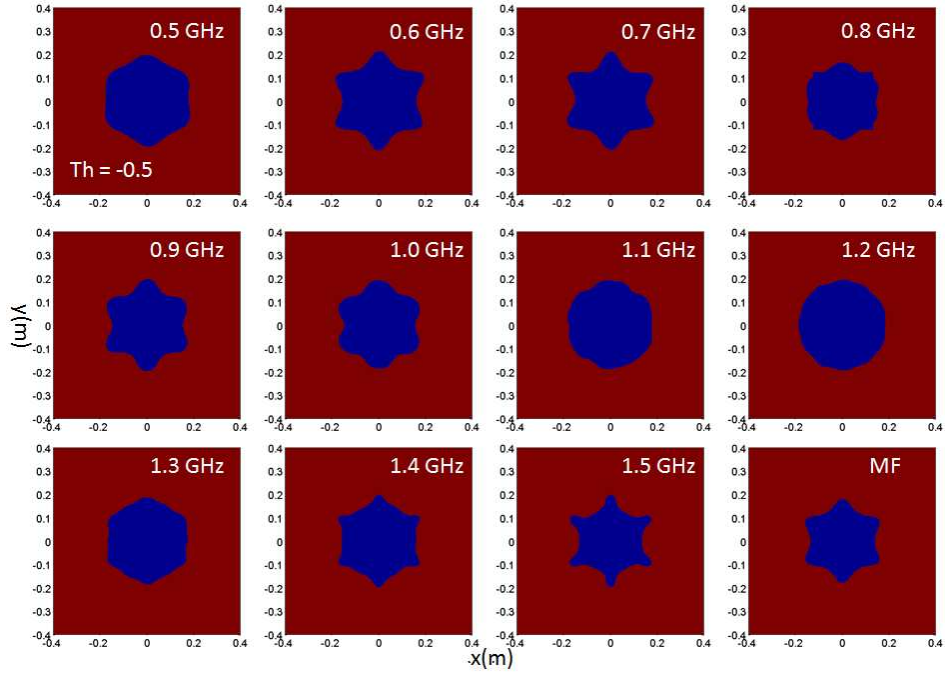


Fig. B.3: Single object contours when solving for α_z with threshold of -0.5.

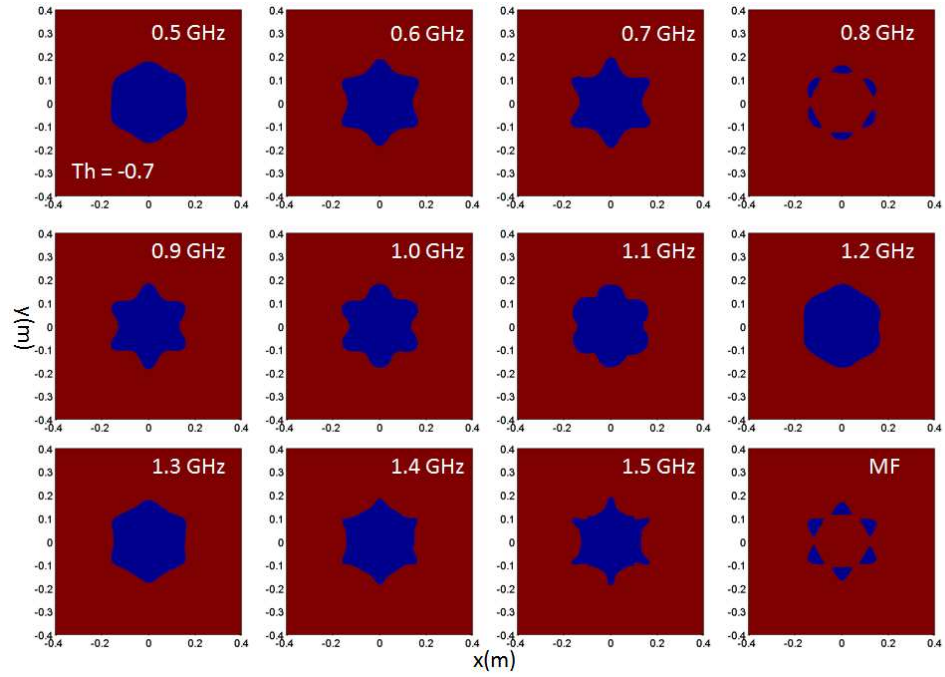


Fig. B.4: Single object contours when solving for α_z with threshold of -0.7.

First scenario: six-pointed star, α constant

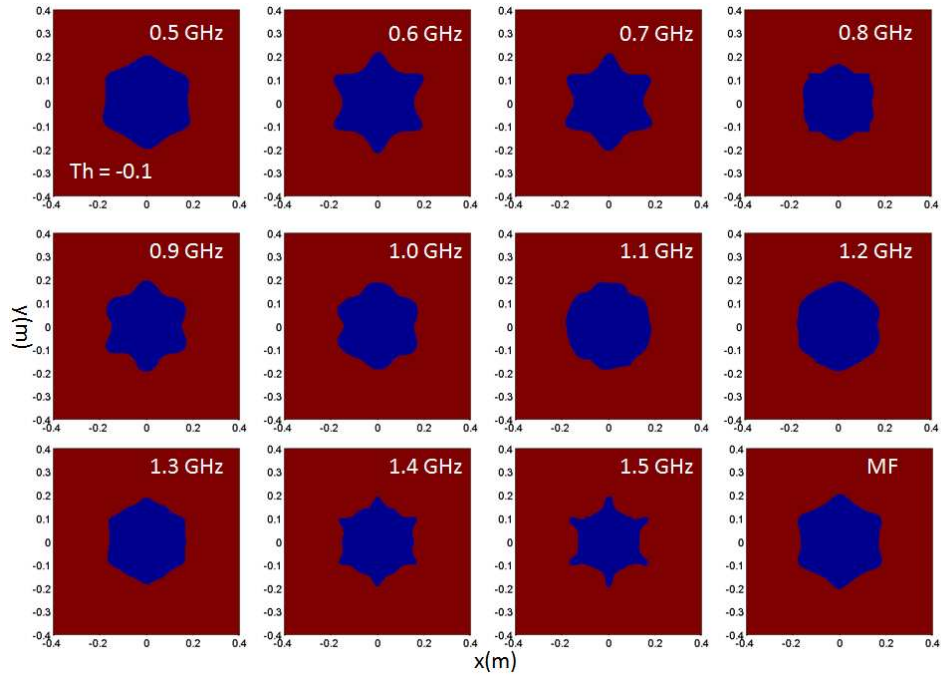


Fig. B.5: Single object contours when using constant α_z with threshold of -0.1.

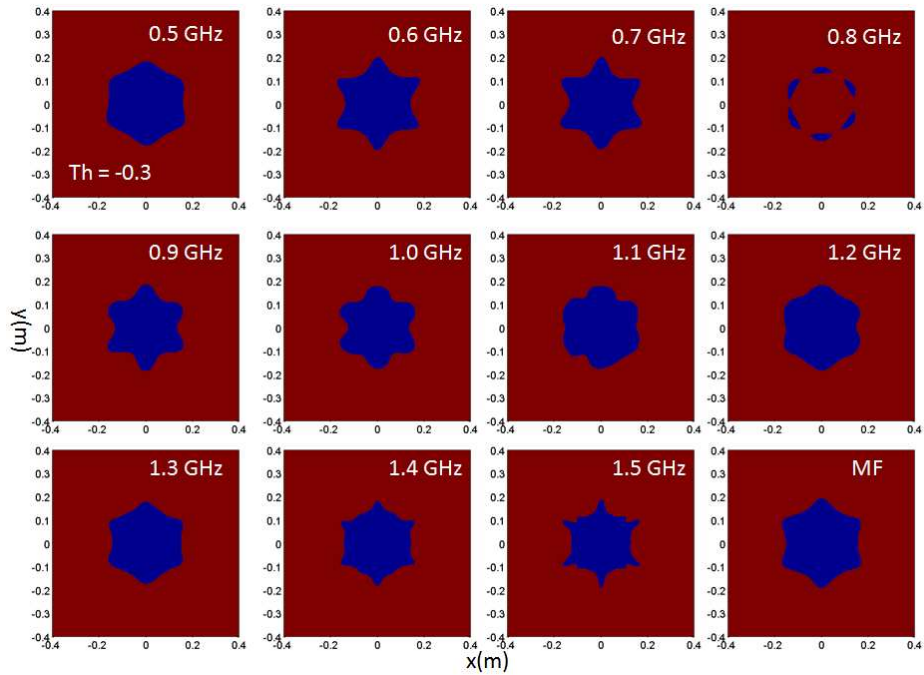


Fig. B.6: Single object contours when using constant α_z with threshold of -0.3.

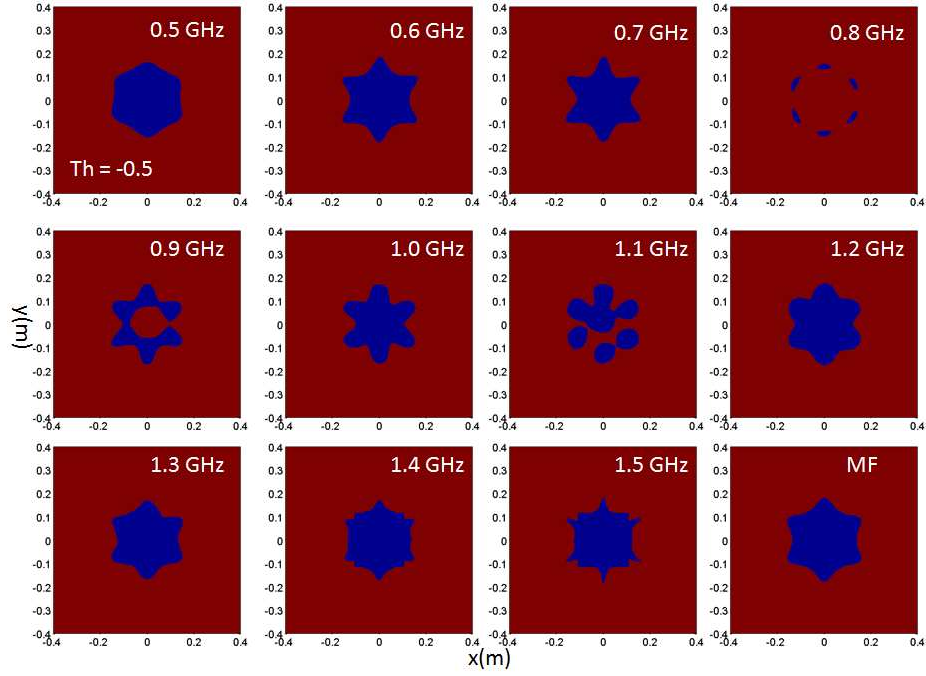


Fig. B.7: Single object contours when using constant α_z with threshold of -0.5.

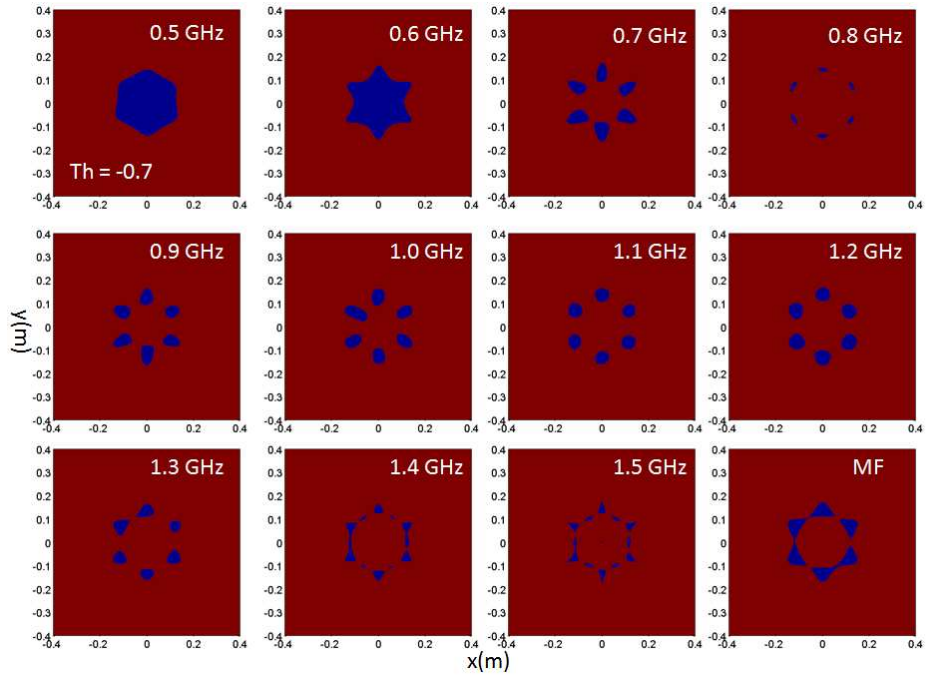


Fig. B.8: Single object contours when using constant α_z with threshold of -0.7.

Second scenario: two ellipses, α constant

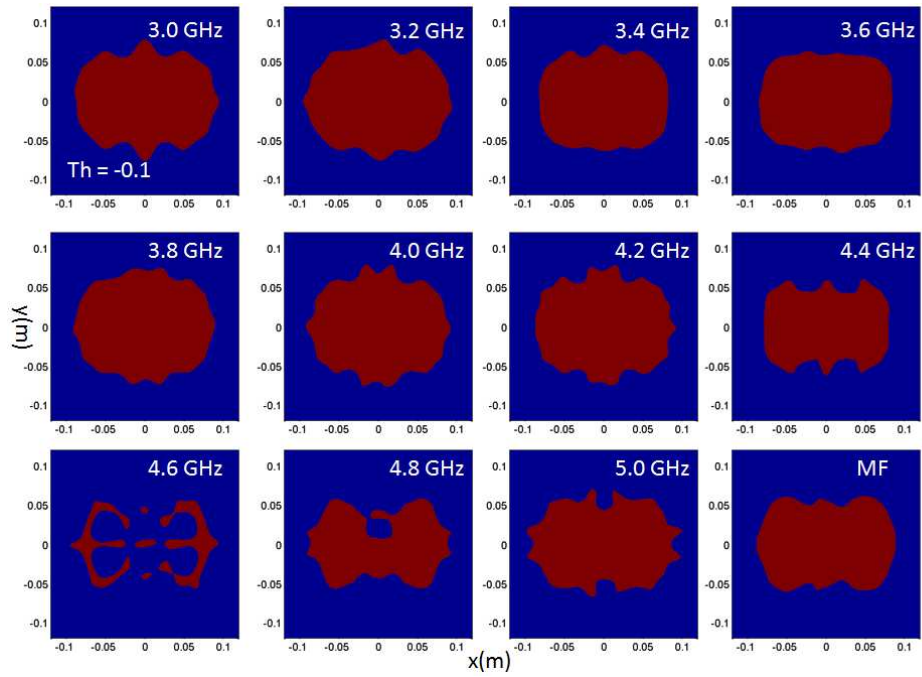


Fig. B.9: Multiple object contours when using constant α_z with threshold of -0.1.

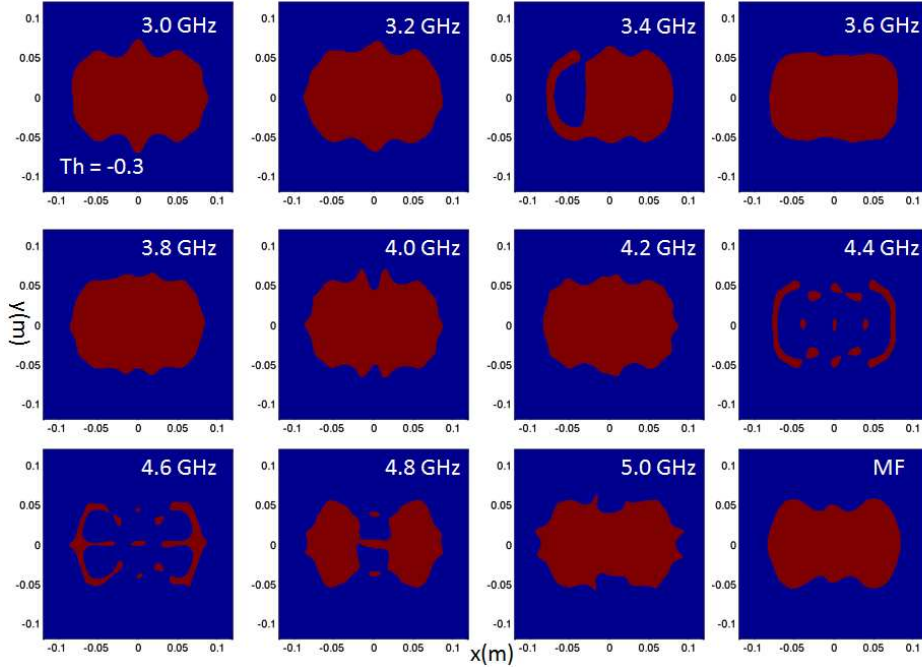


Fig. B.10: Multiple object contours when using constant α_z with threshold of -0.3.

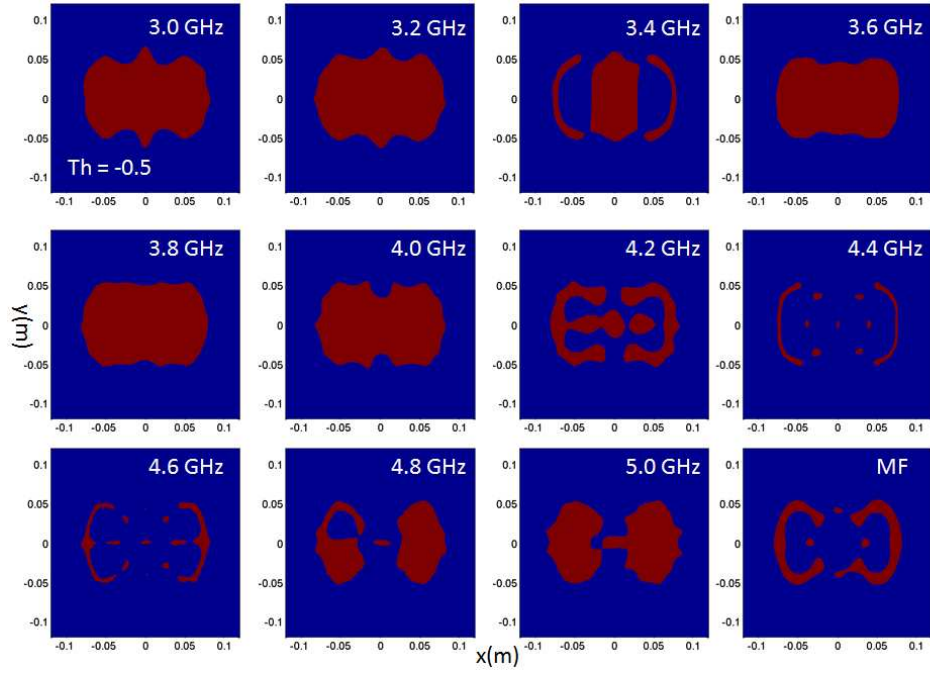


Fig. B.11: Second scenario contours when using constant α_z with threshold of -0.5.

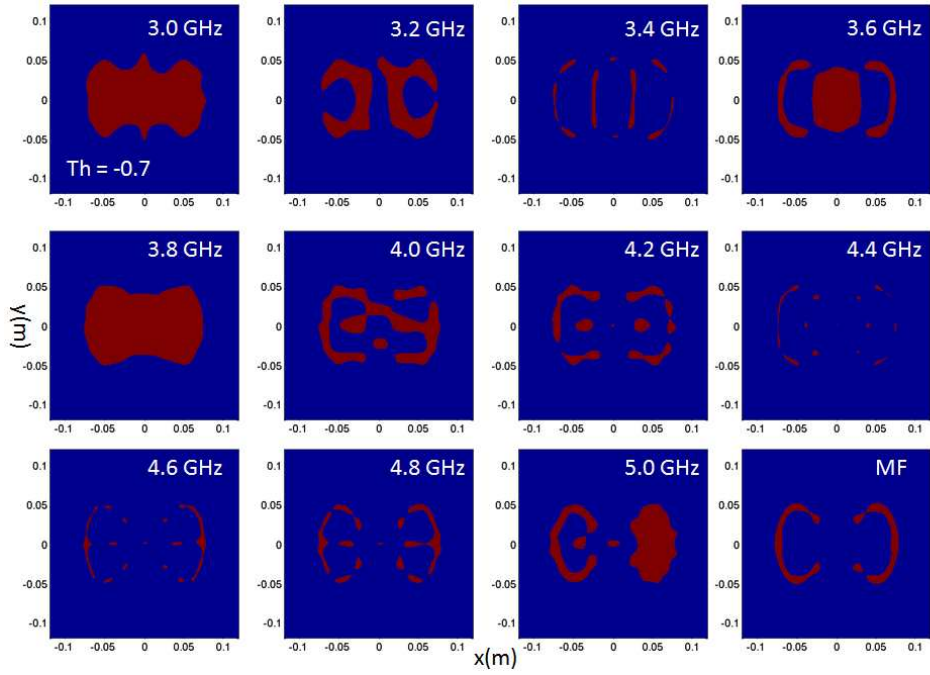


Fig. B.12: Second scenario contours when using constant α_z with threshold of -0.7.