NASA-CR-196282

-- Final Report --

July 1, 1992 to April 15, 1994

Advanced Studies of Electromagnetic Scattering

Grant No. NCC3-273

Submitted to

Dr. Afroz Zaman
National Aeronautics and Space Administration
Lewis Research Center
Cleveland, OH 44135

Prepared by

Hao Ling
Department of Electrical and Computer Engineering
The University of Texas at Austin
Austin, TX 78712-1084

July 1994

-- Final Report --

July 1, 1992 to April 15, 1994

Advanced Studies of Electromagnetic Scattering

Grant No. NCC3-273

Submitted to

Dr. Afroz Zaman
National Aeronautics and Space Administration
Lewis Research Center
Cleveland, OH 44135

Prepared by

Hao Ling
Department of Electrical and Computer Engineering
The University of Texas at Austin
Austin, TX 78712-1084

July 1994

PROJECT SUMMARY

PROJECT OVERVIEW:

The contract "Advanced Studies of Electromagnetic Scattering" (under Cooperative Agreement NCC3-273) was awarded to the University of Texas at Austin in July 1992. The objective of the project is to develop and extend the ISAR (inverse synthetic aperture radar) image simulation capabilities of the ray-shooting code *Xpatch*. The total amount awarded, $20,000 + $40,000 = $60,000, covers the period from July 1, 1992 to April 15, 1994.

Technical Monitor: Dr. Afroz Zaman
M.S. 54-8
NASA Lewis Research Center
21000 Brookpark Road
Cleveland, OH 44135
Tel: (216)-433-3415
FAX: (216)-433-6371

Principal Investigator: Dr. Hao Ling
Department of Electrical and Computer Engineering
The University of Texas at Austin
Austin, TX 78712-1084
Tel: (512)-471-1710
FAX: (512)-471-5445
e-mail: ling@ece.utexas.edu

Other Investigator: Mr. Rajan Bhalla
Graduate Student

SUMMARY OF ACCOMPLISHMENTS

In radar signature applications it is often desirable to generate the range profiles and inverse synthetic aperture radar (ISAR) images of a target. They can be used either as identification tools to distinguish and classify the target from a collection of possible targets, or as diagnostic/design tools to pinpoint the key scattering centers on the target. The simulation of synthetic range profiles and ISAR images is usually a time intensive task and computation time is of prime importance. Our research has been focused on the development of fast simulation algorithms for range profiles and ISAR images using the shooting and bouncing ray (SBR) method.

The shooting and bouncing ray method is a high frequency electromagnetic simulation technique for predicting the radar returns from realistic aerospace vehicles and the scattering by complex media. The basic idea behind the SBR method is very simple. Given the geometrical description of a target, a large set of geometrical optics rays is shot towards the target. Rays are traced according to the laws of geometrical optics as they bounce around the target. At the exit point of each ray, a ray-tube integration is performed to sum up its contribution to the total scattered field. While the basic idea behind the SBR methodology is simple, when combined with CAD tools for geometrical modeling and fast ray-tracing algorithms developed in computer graphics, this technique becomes a very powerful tool for characterizing the scattering from large, complex targets. One such development is the general-purpose SBR code, *Xpatch*, which is currently used by the Air Force and the aerospace industry in programs related to target identification and low-observable vehicle design.

During the duration of this project, we have developed a series of fast schemes for *Xpatch* to speed up range profile computation and ISAR image simulation. First, we have developed the theory and implementation of using bistatic data for ISAR image formation. The major computation time for *Xpatch* can be attributed to two parts: geometrical ray

tracing and electromagnetic computation. The latter includes the computation of the geometrical optics field associated with each ray and the ray tube integration algorithm. As the complexity and size of the target increases the ray tracing time can become a dominant portion of the total computation time. In the usual practice where monostatic data are utilized to generate ISAR images, the ray tracing portion of the code is executed for every look angle. Bistatic scattered field data, on the other hand, is cheaper to acquire, since ray tracing is performed once for the incident direction and only the electromagnetics calculation is needed for every look angle. We have derived the principle of the bistatic imaging algorithm. The ISAR image of the scattering target can be reconstructed via Fourier inversion of the bistatic far field data under the physical optics approximation. The slice of Fourier data collected under the bistatic condition is a circle shifted from the center by the incident wave vector. To generate the ISAR image using the FFT algorithm, the bistatic data need to be interpolated onto a rectangular grid. ISAR images generated using the *Xpatch* data for various targets have been studied. We find that when multiple scattering contributions are small, the ISAR images generated via the bistatic arrangement bear excellent fidelity to their monostatic counterpart.

In a separate development we have derived a closed form time-domain ray-tube integration formula for the computation of the time-domain response (or range profile) of a conducting target. This formula gives the explicit contribution of each exit ray in the time domain. Therefore, by summing the contribution from each ray in the time domain directly, the overall time-domain response of the target can be obtained without resorting to any multi-frequency calculations. Furthermore, by utilizing the bistatic imaging scheme derived earlier, we have also extended the above idea to two-dimensional ISAR image formation. A closed form image-domain ray-tube integration formula has been derived that gives the contribution of each ray to the overall ISAR image directly, without resorting to any multi-frequency, multi-aspect calculations.

Finally, we have found that the form of the time-domain and image-domain ray-tube integration formula can be re-written as a convolution between a non-uniformly sampled impulse train and a time-domain or image-domain ray spread function. To perform the convolution by the fast Fourier transform algorithm, the non-uniformly sampled impulse train can be first interpolated onto a uniformly sampled grid using a scheme proposed by T. D. Sullivan. Using the Sullivan scheme, a speed gain of a factor of 30 is achieved over the direct convolution in range profile computation and a factor of 180 in ISAR image formation for a typical aircraft at X-band.

In addition, we have implemented fast computation of the first-bounce (or physical optics) return using the time-domain z-buffer technique. We have also developed the adaptive ray tracing scheme proposed by M. Baden of Georgia Tech. The idea is that for those rays which can be inferred from neighboring rays, it is not necessary to trace them at all. As the ray-tracing time grows as the square of the operating frequency, the Baden scheme becomes quite useful for signature simulation in the 35 and 94 GHz ranges, as the percentage of predictable rays becomes large.

In summary, we have developed a series of fast schemes for *Xpatch* to speed up range profile computation and ISAR image formation. Our most significant achievement to date is the reduction of the range profile and image formation time, for a realistic airplane at X-band, to less than one hour on the Silicon Graphics Indigo workstation. In particular, the image formation time we have achieved is more than two orders of magnitude (i.e., over 120 hours) less than that of the standard frequency-aspect image formation process (see Fig. 1). Both the resulting range profile and the image quality have also been shown to bear excellent fidelity to the standard formation methods. All of the new features we have developed to date have been incorporated into the latest version of *Xpatch3*, which is due to be released in the fall of 1994.

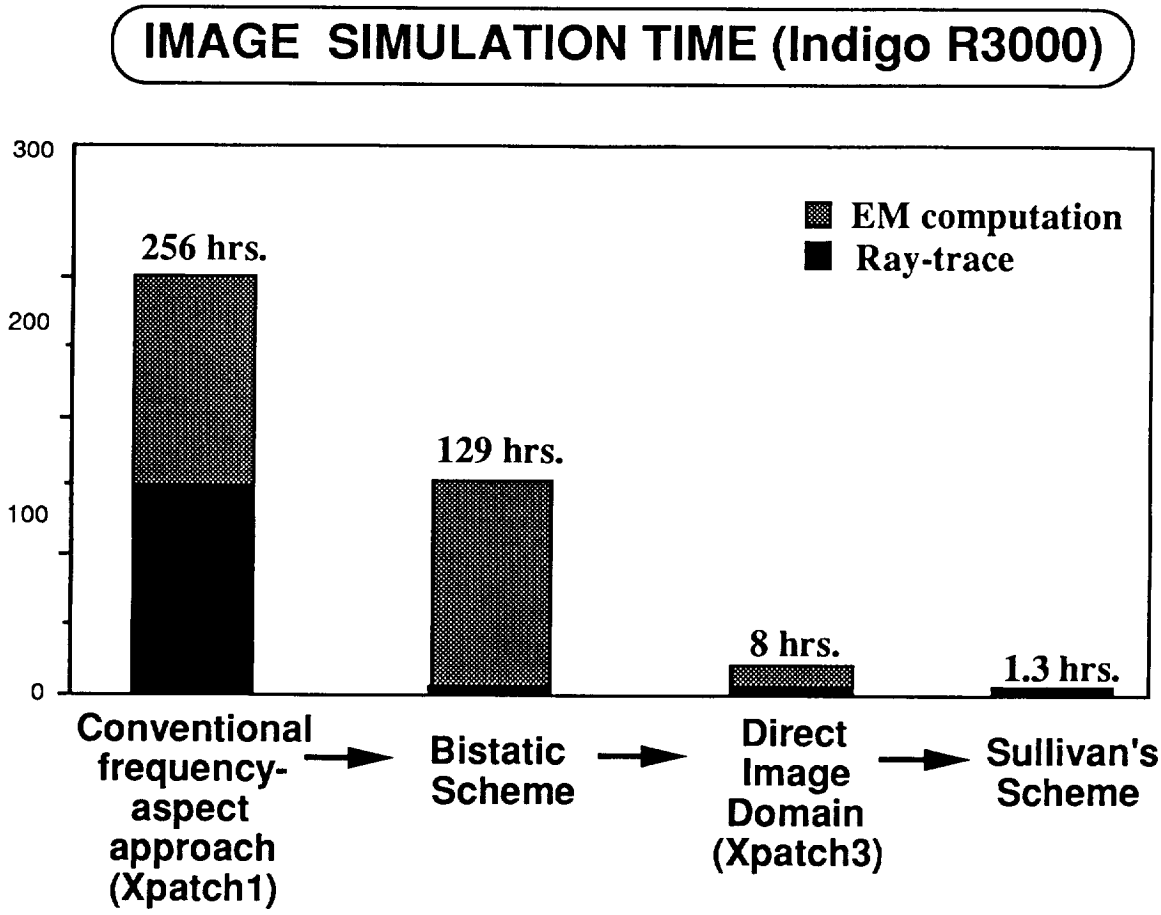


Fig. 1. Time performance of Xpatch3 for ISAR image formation. The time shown is for a Silicon Graphics Indigo workstation with an R3000 CPU. The R4000 CPU is approximately a factor of two faster.

PUBLICATIONS AND PRESENTATIONS

Technical Reports:

[1] H. Ling and R. Bhalla, "Time-domain ray-tube integration formula for the shooting and bouncing ray technique," Tech. Rept., Univ. of Texas at Austin, April 1993.

[2] R. Bhalla and H. Ling, "Image-domain ray-tube integration formula for the shooting and bouncing ray technique," Tech. Rept., Univ. of Texas at Austin, July 1993.

[3] R. Bhalla and H. Ling, "A fast algorithm for signature prediction and image formation using the shooting and bouncing ray technique," Tech. Rept., Univ. of Texas at Austin, January 1994.

Journal Publications:

[1] R. Bhalla and H. Ling, "ISAR image formation using bistatic data computed from the shooting and bouncing ray technique," *J. Electromag. Waves Applications*, vol. 7, pp. 1271-1287, September 1993.

[2] R. Bhalla and H. Ling, "Image-domain ray-tube integration formula for the shooting and bouncing ray technique," submitted for publication in *Radio Science*, July 1993.

[3] S. K. Jeng, R. Bhalla, S. W. Lee, H. Ling and D. J. Andersh, "A time-domain SBR technique for range-profile computation," submitted for publication in *IEEE Trans. Antennas Propagat.*, September 1993.

[4] R. Bhalla and H. Ling, "A fast algorithm for signature prediction and image formation using the shooting and bouncing ray technique," submitted for publication in *IEEE Trans. Antennas Propagat.*, January 1994.

[5] R. Bhalla, H. Ling, S. W. Lee and D. J. Andersh, "Dynamic simulation of Doppler spectra of targets with rotating parts," submitted for publication in *Elect. Lett.*, May 1994.

Conference Presentations:

[1] R. Bhalla and H. Ling, "ISAR image formation using bistatic data from XPATCH," 9th Annual Review of Progress in Applied Computational Electromagnetics, pp. 150-157, Monterey, CA, March 1993.

[2] R. Bhalla and H. Ling, "Fast ISAR image formation of complex targets using the shooting and bouncing ray method," National Radio Science Meeting, p. 114, Boulder, CO, January 1994.

[3] R. Bhalla and H. Ling, "A fast algorithm for signature prediction and image formation using the shooting and bouncing ray technique," International IEEE AP-S Symposium, pp. 1990-1993, Seattle, WA, June 1994 (Finalist, Student Paper Competition).

[4] R. Bhalla and H. Ling, "ISAR image simulation of targets with moving parts using the shooting and bouncing ray technique," International IEEE AP-S Symposium, pp. 1994-1997, Seattle, WA, June 1994.

[5] D. J. Andersh, S. W. Lee, H. Ling and C. L. Yu, "Xpatch: A high frequency electromagnetic scattering prediction code using shooting and bouncing rays," to be presented at the Fifth Annual Ground Target Modeling and Validation Conference, Houghton, MI, August 1994.

[6] R. Bhalla and H. Ling, "Fast inverse synthetic aperture radar image simulation of complex targets using ray shooting," to be presented at the IEEE International Conference on Image Processing, Austin, TX, November 1994.

Other Presentations:

[1] H. Ling, "ARTI Reseach at the University of Texas," ARTI Synthesis Program Review, Wright-Patterson Air Force Base, Dayton, Ohio, July 10, 1992.

[2] H. Ling, "Simulating monostatic data using bistatic data," ARTI Synthesis Program Review, Wright-Patterson Air Force Base, Dayton, Ohio, January 15, 1993.

[3] H. Ling, "ISAR image generation via Xpatch," MIT Lincoln Laboratory, Lexington, MA, July 22, 1993.

[4] H. Ling, "ISAR image generation via Xpatch," Sandia National Laboratory, Albuquerque, NM, August 23, 1993.

[5] H. Ling, "Fast ISAR generation via Xpatch," Noncorporative Target Identification Program Review, Wright-Patterson Air Force Base, Dayton, Ohio, October 28, 1993.

M.S. Theses:

[1] R. Bhalla, "ISAR image formation using bistatic data from the shooting and bouncing ray technique," May 1993.

Time Domain Ray-Tube Integration Formula for the Shooting-and-Bouncing-Ray Technique

- Fast Time and Frequency Calculation Using Xpatch3 -

Hao Ling and Rajan Bhalla

Department of Electrical and Computer Engineering
The University of Texas at Austin
Austin, TX 78712-1084

April 1993

Air Force Wright Laboratory
through NASA Grant NCC 3-273

TIME DOMAIN RAY-TUBE INTEGRATION FORMULA FOR THE SHOOTING-AND-BOUNCING-RAY TECHNIQUE

Hao Ling and Rajan Bhalla

Department of Electrical and Computer Engineering
The University of Texas at Austin
Austin, TX 78712-1084

Abstract In this report, we present a rederivation of the time-domain ray-tube integration scheme proposed by Jeng and Lee [1]. Our new derivation and subsequent implementation of the new formula corrects the inaccuracies which exist in the current version of Xpatch3, Version 1. Using the new corrected Xpatch3, exact agreement with the data generated by Xpatch1 is achieved for the target "turkey" and "v1old". In addition, we show time improvement on the Silicon Graphics for "v1old" from 7.4 hours using Xpatch1 (all bounces by SBR) to 1.6 hours using Xpatch3. Further time improvements in Xpatch3 can be anticipated if the first bounce contribution can be computed by time domain physical optics utilizing the z-buffer.

This work is supported by NASA Grant NCC 3-273.

1. INTRODUCTION

In the shooting and bouncing ray (SBR) technique, the scattered field is usually computed in the frequency domain. The scattered field at one frequency is calculated as the sum of individual ray contributions at that frequency. To generate the range profile, the scattered field must be computed over a band of frequencies. Then by inverse Fourier transforming the resulting data, the demodulated time-domain return, or the range profile, can be obtained. This concept is described by the following expression:

$$E^s(t) = \text{F.T.}^{-1} \{ \sum_{\text{i rays}} E_i^s(\omega) \} \tag{1}$$

This is the philosophy adopted in the code Xpatch1.

Recently, Jeng and Lee [1] reported a closed form expression for the time-domain response contributed by each exit ray. The summation of this contribution over all rays then results directly in the range profile:

$$E^s(t) = \sum_{\text{i rays}} E_i^s(t) \tag{2}$$

where $E_i^s(t)$ is the closed form time-domain formula for each ray. Subsequent implementation of this idea is now in the code Xpatch3. However, the accuracy of the calculated range profile using Xpatch3 does not agree very well with that generated using Xpatch1, which is considered as the reference standard. Furthermore, the time-domain sampling requirement in computing (2) is quite dense, with the required Δt being proportional to the reciprocal of the center frequency (10 GHz).

In this report, we present a rederivation of the time-domain ray-tube integration scheme of Jeng and Lee. Our new derivation and subsequent implementation of the new formula corrects the inaccuracies which exist in the current version of Xpatch3, Version 1. Using the new corrected Xpatch3, exact agreement with the data generated by Xpatch1 is

achieved for the target "turkey" and "v1old". The derivation of the time-domain formula is carried out in Sec. 2, followed by several numerical examples and timing results in Sec. 3. We will discuss how the computation time of Xpatch3 can be further reduced in Sec. 4.

2. TIME-DOMAIN RAY-TUBE INTEGRATION FORMULA

We will now derive the time-domain ray-tube integration formula in the SBR technique. At an observation point (r,θ,ϕ) the frequency domain scattered far field is given by

$$E(\omega,\theta,\phi) = \frac{e^{-jkr}}{r}(\hat{\theta}A_\theta + \hat{\phi}A_\phi) \tag{3a}$$

where the explicit expression for $E(\omega,\theta,\phi)$ was derived in [2] and takes on the form

$$\begin{bmatrix} A_\theta \\ A_\phi \end{bmatrix} = \sum_{\substack{i \\ \text{all rays}}} \begin{bmatrix} B_\theta \\ B_\phi \end{bmatrix} \left(\frac{jk}{2\pi}\right) (\Delta A)_{exit}\, S(\theta,\phi)\; e^{jk\cdot r_A} \tag{3b}$$

In the above expression, B_θ, B_ϕ are related to the aperture electric and magnetic fields associated with the ray tube, $S(\theta,\phi)$ is the ray-tube shape function (which is usually assumed to be unity), $(\Delta A)_{exit}$ is the exit ray-tube cross section and r_A is the position vector of the last hit point on the scatterer for the ray. Assuming that the scatterer is perfectly conducting, we can factor out the explicit frequency dependence in B_θ, B_ϕ as

$$B_\theta = f_1(E_{ap}, H_{ap}, \hat{s}, \theta, \phi)\; e^{-j\omega\frac{d}{c}} \tag{4}$$

$$B_\phi = f_2(E_{ap}, H_{ap}, \hat{s}, \theta, \phi)\; e^{-j\omega\frac{d}{c}}$$

where d_i is the total distance traveled by the ith ray. Substituting them in (3b) and replacing k by ω/c, we obtain an expression where the the overall frequency dependence is explicitly indicated:

$$\begin{bmatrix} A_\theta \\ A_\phi \end{bmatrix} = \sum_{\substack{i \\ \text{all rays}}} \frac{j\omega}{2\pi c} \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} (\Delta A)_{exit}\, S(\theta,\phi)\, e^{-j\omega(d_i - \hat{k}\cdot r_A)/c} \tag{5}$$

We shall lump all the frequency independent terms into a new vector $\mathbf{C}(\theta,\phi)$ and rewrite the field as (while suppressing the [exp(-jkr)/r] factor):

$$\mathbf{E}(\omega,\theta,\phi) = \sum_{\substack{i \\ \text{all rays}}} \mathbf{C}(\theta,\phi)\,(\frac{j\omega}{2\pi c})\,e^{-j\omega\frac{d_i'}{c}} \tag{6}$$

where

$$\mathbf{C}(\theta,\phi) = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} (\Delta A)_{exit}\, S(\theta,\phi)$$

and $d_i' = d_i - \hat{k}\cdot\mathbf{r}_A$. To generate the time domain expression for the scattered field, we take the inverse Fourier transform of the frequency domain data with bandwidth $\Delta\omega$ and center frequency ω_o:

$$\mathbf{E}(t,\theta,\phi) = (\frac{1}{2\pi}) \int_{\omega_o - \Delta\omega/2}^{\omega_o + \Delta\omega/2} \mathbf{E}(\omega,\theta,\phi)\, e^{j\omega t}\, d\omega \tag{7}$$

Denoting sinc (u) = sin(u)/u and sinc'(u) = d(sinc(u))/du = (-sinc(u) + cos(u))/u, we arrive at a closed form expression for the time-domain scattered field:

$$\mathbf{E}(t,\theta,\phi) = e^{j\omega_o t} \sum_{\substack{i \\ \text{all rays}}} \mathbf{C}(\theta,\phi)\,(\frac{\Delta\omega}{2\pi c})\,e^{-j\frac{d_i'}{c}\omega_o} \cdot$$

$$\cdot \left\{ \frac{\Delta\omega}{4\pi}\, \text{sinc}'(\frac{\Delta\omega}{2}(t - \frac{d_i'}{c})) + j\,\frac{\omega_o}{2\pi}\, \text{sinc}(\frac{\Delta\omega}{2}(t - \frac{d_i'}{c})) \right\} \tag{8}$$

The above closed form expression is in the form of (2) and is the desired expression for numerical implementation.

The time domain technique has been implemented in Xpatch3. Our implementation strategy is as follows. For a given target, we first choose a range window [-R/2,+R/2],

where zero range corresponds to the chosen origin of the target and R is roughly twice the size of the target. We then divide the range window into N equally spaced bins with binwidth $\Delta r = R/N$ and

$$r_n = \Delta r\,(n-1), \qquad n = 1,2,....,N+1 \tag{9}$$

The binwidth is user chosen but should be no greater than (0.15m / bandwidth in GHz) to ensure enough sampling in the time domain. To make the implementation more efficient, we precompute and store the function $W(t_n)$, which is defined as

$$W(t_n) = \left(\frac{\Delta\omega}{2\pi c}\right)\left\{\frac{\Delta\omega}{4\pi}\,\mathrm{sinc}\left(\frac{\Delta\omega}{2}\,t_n\right) + j\,\frac{\omega_o}{2\pi}\,\mathrm{sinc}\left(\frac{\Delta\omega}{2}\,t_n\right)\right\} \tag{10}$$

where $t_n = 2\,r_n/c$. The magnitude of the time-domain scattered field, which is exactly the desired range profile, can be rewritten in terms of $W(t_n)$ as

$$\left|E(t_n,\theta,\phi)\right| = \left|\sum_{\substack{i \\ \text{all rays}}} C(\theta,\phi)\; e^{-j\frac{d_i}{c}\omega_0}\; W\left(t_n - \frac{d_i}{c}\right)\right| \tag{11}$$

In the implementation, once d_i for the ith ray is obtained following the ray tracing, we can look up the precomputed table of W and updating each range bin by the appropriate value. Note that d_i determines the amount of time (or range) shift there is in placing W into the range bins. The cumulative sum of all the rays will give the required range profile. We can also Fourier transform the data to generate the frequency response. Two differences are noted here when comparing the present time-domain implementation to the incorrect Version 1 implementation of the time-domain formula. First, complex algebra is used. Second, the required time-domain sampling in the new formulation is proportional to the reciprocal of the bandwidth (2 GHz). This is in contrast to the old implementation where the time sampling must be proportional to the reciprocal of the center frequency (10 GHz).

The new scheme allows a much coarser time sampling to be used due to the factorization of the carrier modulation $\exp(j\omega_0 t)$ in (8).

3. RESULTS

Two numerical examples are presented next to demonstrate the accuracy and computation time of the time-domain technique. The range profile for "turkey" has been generated in the time domain using a range window of 90 inches, a center frequency of 10 GHz and a bandwidth of 4 GHz. Fig. 1 compares the range profile generated by the revised Xpatch3 to that generated by Xpatch1. The two results show excellent agreement. By Fourier transforming the time domain data from Xpatch3, we can also easily generate the frequency domain response. This result is shown in Fig. 2. The frequency domain data generated from Xpatch1 is also plotted in Fig. 2 for comparison. The agreement between Xpatch3 and Xpatch1 is again excellent. Figs. 3 and 4 show respectively the time domain and frequency domain data generated by Xpatch3 and Xpatch1 for "v1old". A range window of 900 inches, a center frequency of 10 GHz and a bandwidth of 2 GHz are used in generating the data. Again, very good agreement between the two results is observed for this much more complex target.

In Table 1, a run time comparison between the code Xpatch1 and Xpatch3 on our Silicon Graphics Indigo XS-24 (R3000) is tabulated. For "turkey", the run time of Xpatch3 is 4.8 minutes versus 9.6 minutes for Xpatch1. For "v1old" the run time of Xpatch3 is 1.6 hours versus 7.4 hours for Xpatch1. The time saving of using Xpatch3 over Xpatch1 is a factor of 2 for "turkey" and a factor of almost 5 for "v1old". It is important to point out here that in generating the Xpatch1 results, we have computed all bounces by SBR. This is a "fair" comparison in that Xpatch3 also computes all bounces by SBR and a direct comparison in both run time and accuracy can be made. If we use the option of computing the first bounce by physical optics in Xpatch1, the run time is greatly reduced to 2.5 hours. This implies that if we can implement the time-domain physical optics feature (using either software or hardware based z-buffer), the run time of Xpatch3 can be further reduced. This projection is indicated in Table 2. Further discussion on the computation time will be made in Sec. 4.

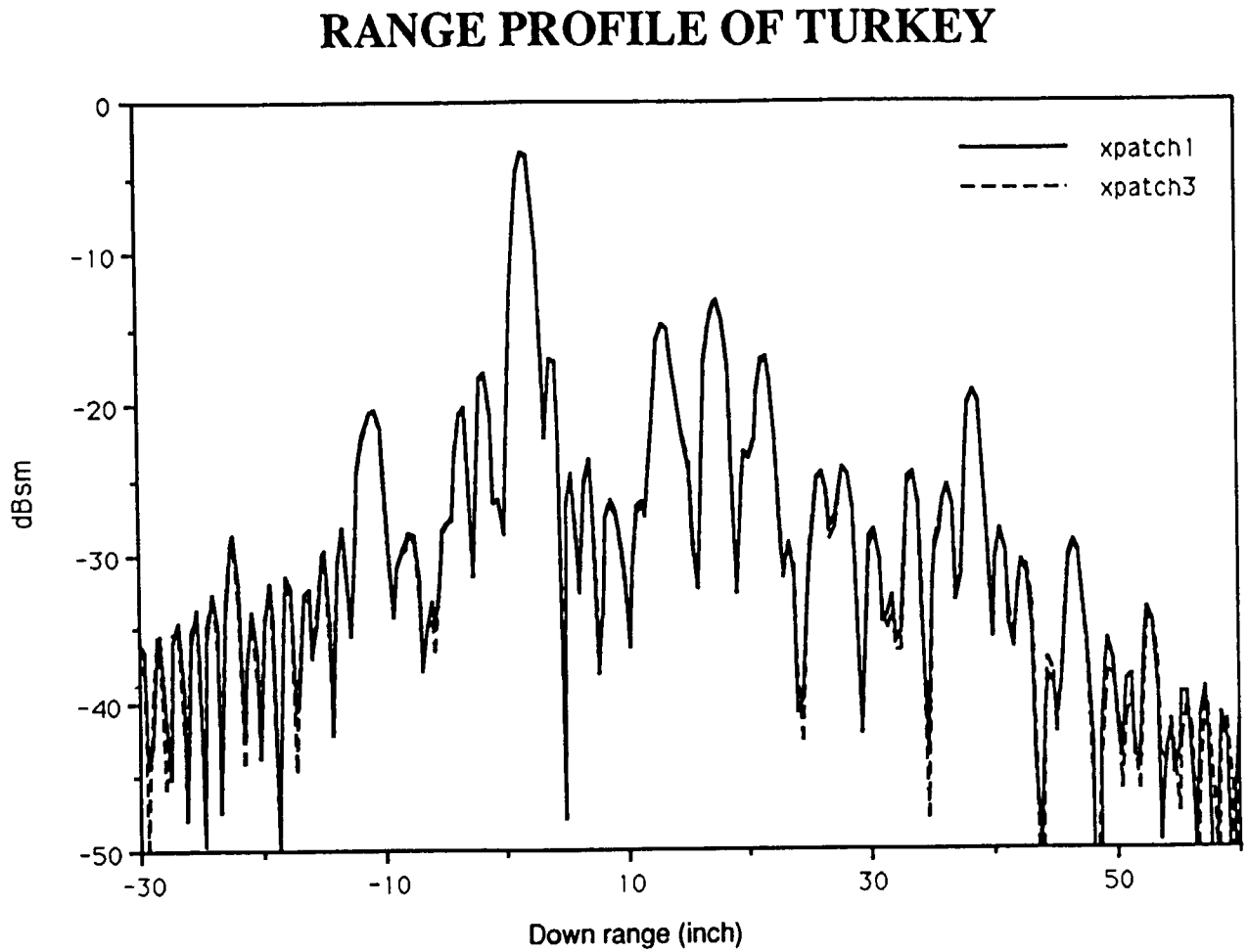


Fig. 1. Comparison of range profile of turkey computed using xpatch1 and xpatch3. The range profile is computed at the center frequency of 10 GHz and a bandwidth of 4 GHz.

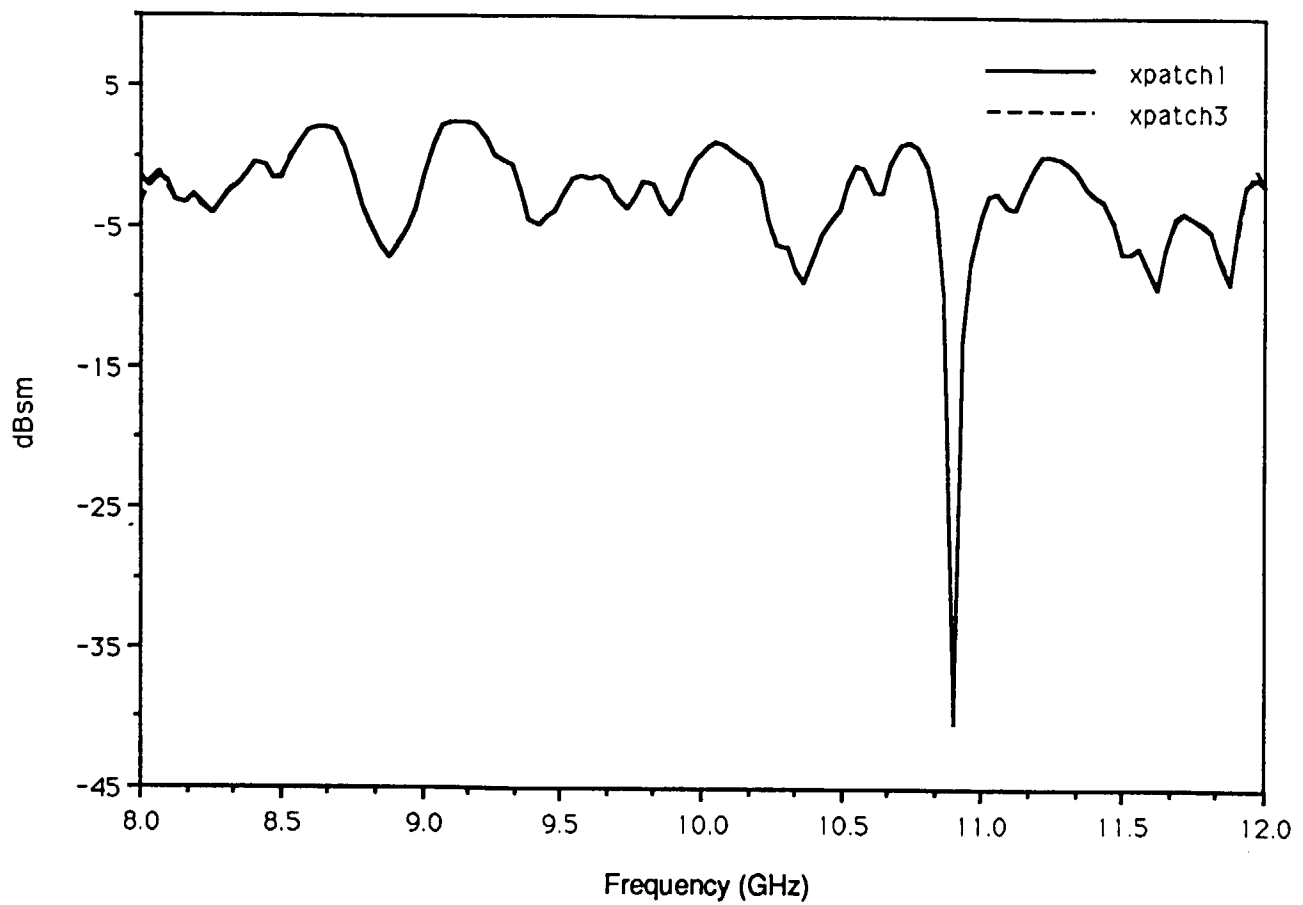


Fig. 2. Comparison of the RCS of turkey computed using xpatch1 and xpatch3. In xpatch3 the RCS is computed in the time domain and then fourier transformed to give the frequency domain RCS.

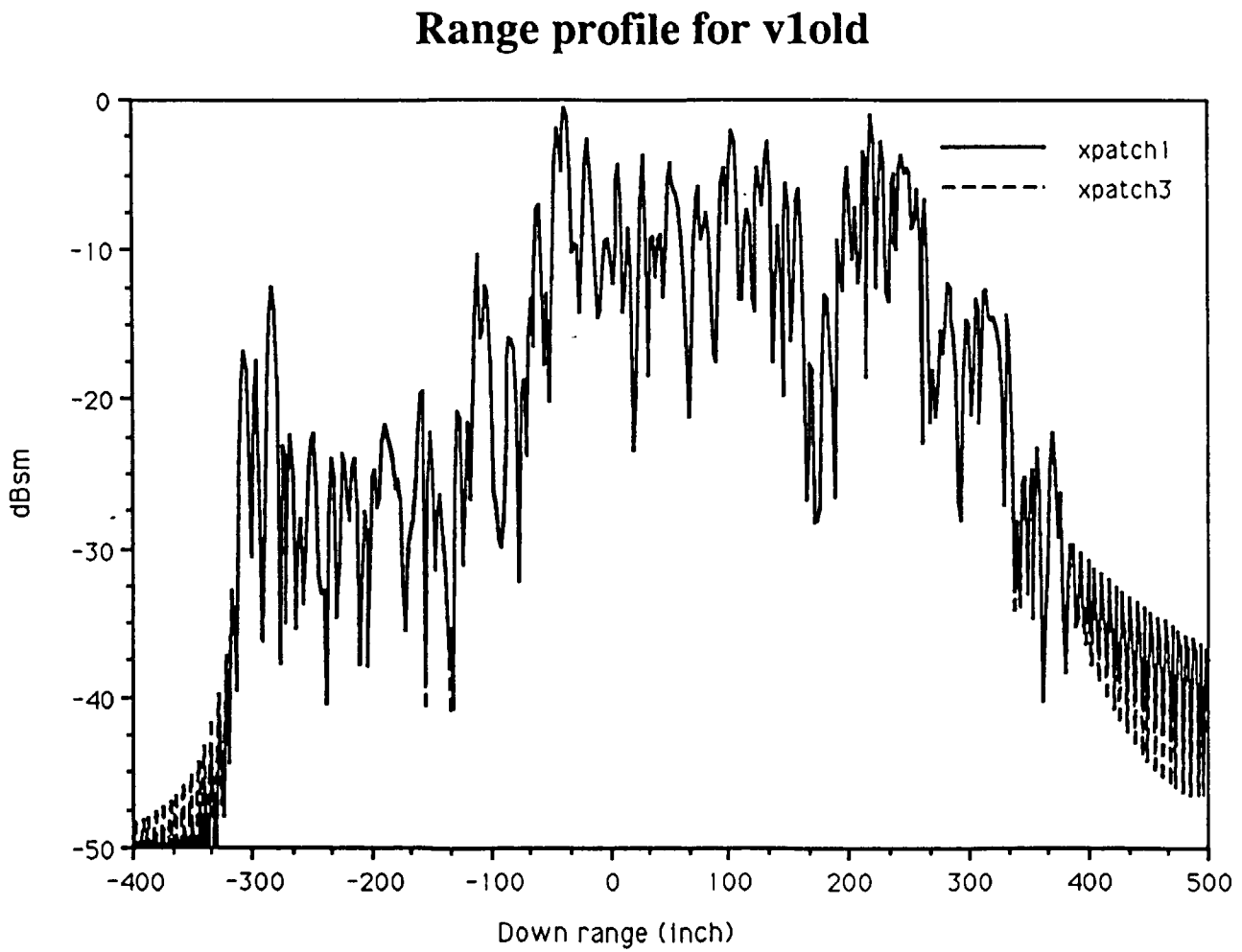


Fig. 3. Comparison of range profile of v1old computed from xpatch1 and xpatch3.

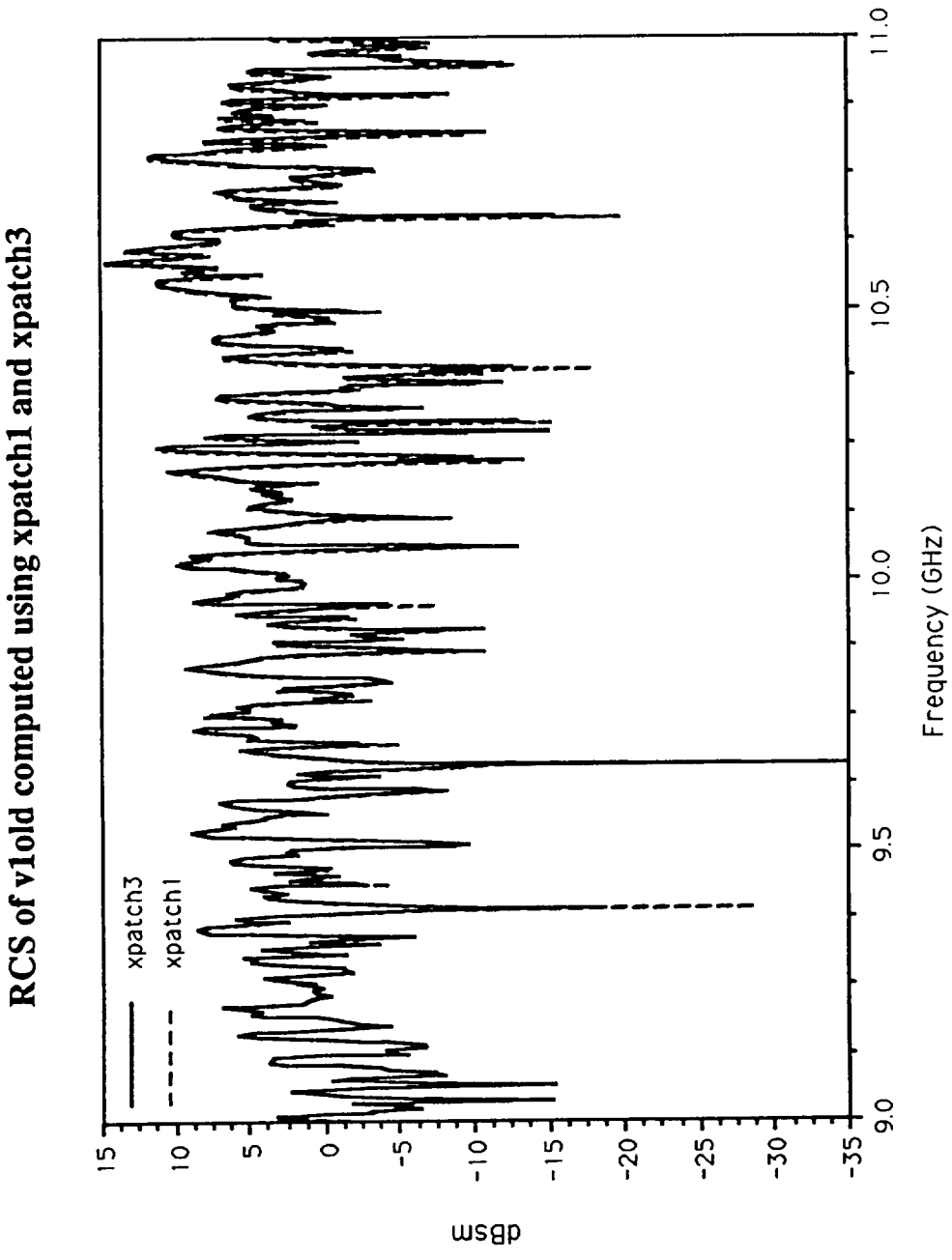


Fig. 4. Comparison of RCS of v1old computed using xpatch1 and xpatch3. In xpatch3 the RCS is computed in the time domain and then fourier tranformed to give the frequency domain RCS.

TABLE 1

	xpatch1 (all bounce by SBR)	**xpatch3**
Turkey	9.7 min	4.8 min
V1old	7.4 hrs	1.6 hrs

TABLE 2

	All bounce by SBR	1st bounce by PO	Approximations
xpatch1	7.4 hrs	2.5 hrs	Nausbaum 1.3 hrs
xpatch3	1.6 hrs	< 1 hr. **	Truncate W(t) < 0.5 hr. **

** To be implemented.

4. IDEA FOR ACHIEVING FURTHER REDUCTION IN RUN TIME

The computation time for the time-domain technique can be divided into two parts: ray-tracing time for computing d_i' and time to update the contribution of each ray to the N range bins. Step 2 is directly proportional to the number of range bins we need to update. Fig. 5 shows a plot of |W(t)| vs. t for a center frequency of 10 GHz and a bandwidth of 4 GHz. It is evident that |W(t)| decays very much like a sinc function and becomes relatively small (30 dB below the main lobe) after about 10 sidelobes. If we update only M bins for each ray, where M is the range extent of W with 10 sidelobes, substantial time savings in step 2 can be achieved. Of course as we decrease the number of sidelobes the dynamic range over which the range profile is accurate also decreases. This is demonstrated in Figs. 6 and 7 for the target "turkey". In Fig. 6, the Xpatch3 range profile is obtained by updating only 4 sidelobes of W (or a total of 9 lobes) for each ray. We notice the error in truncating W. The run time for Xpatch3 is reduced from 4.8 minutes to 3.6 minutes. Fig. 7 shows the improvement in the accuracy as we go out to 10 sidelobes (or a total of 21 lobes). The run time is 4 minutes. We find that 10 sidelobes give a 40 dBsm dynamic range in the range profile, but the exact number of sidelobes needed will depend on the required accuracy as well as the target.

Fig. 8 shows the frequency domain data generated from Fourier transforming the time domain data in Fig. 7. In comparison with the Xpatch1 data, we observe that there is good agreement in the middle of the frequency band from around 8.5-11.5 GHz, and the inaccuracy caused by truncating W exists mainly near the two band edges. We can easily understand this edge noise as follows. Since the original frequency domain data are assumed to be band-limited, the time-domain range profile is, strictly speaking, of infinite extent. When the range profile is truncated in time, Gibb's phenomenon will occur in the frequency domain. In conjunction with the FFT operation we use to perform the Fourier transform, aliasing noise can be expected at the two edges of the frequency band. Usually this edge noise is not noticeable if a large enough range window is calculated in the time

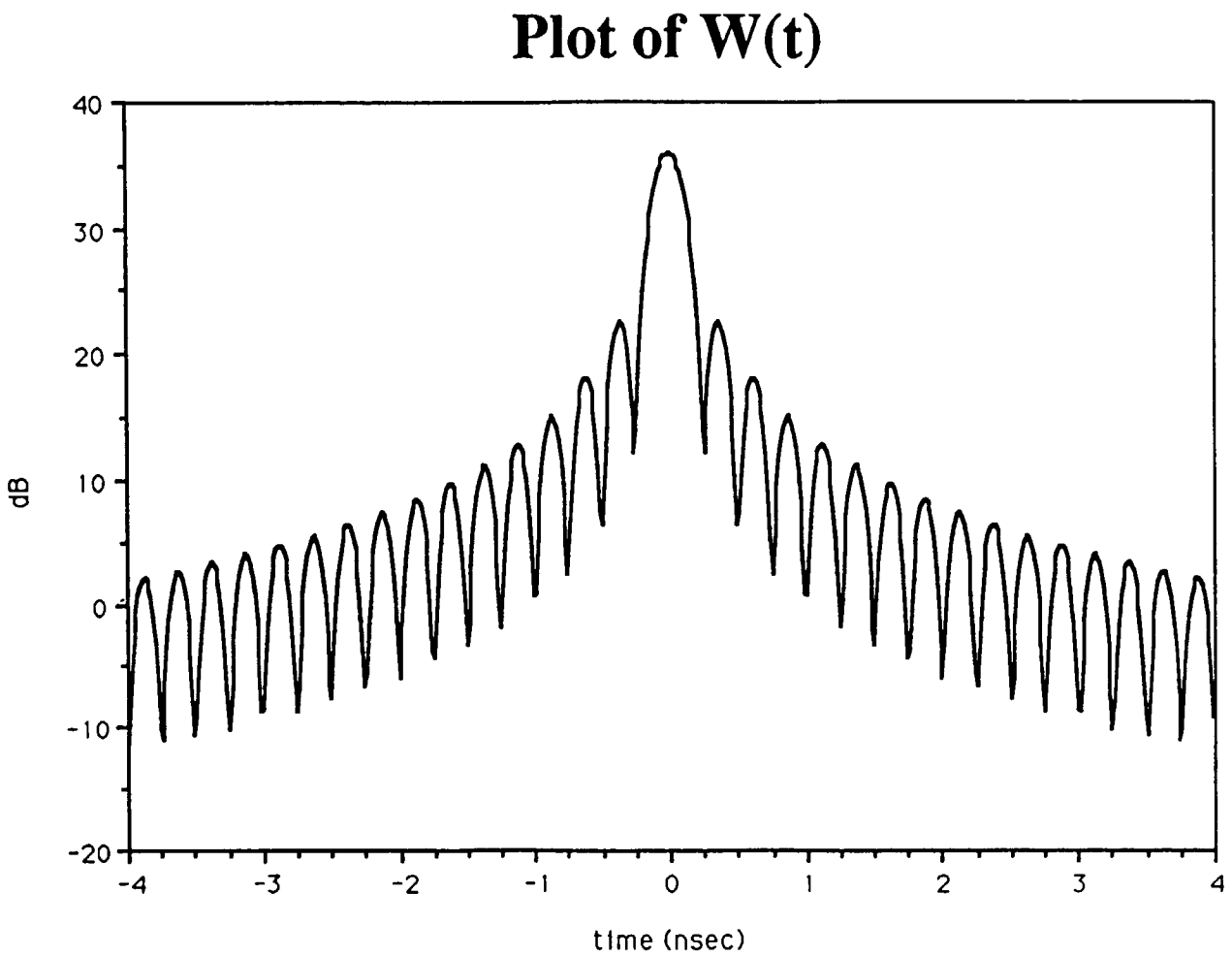
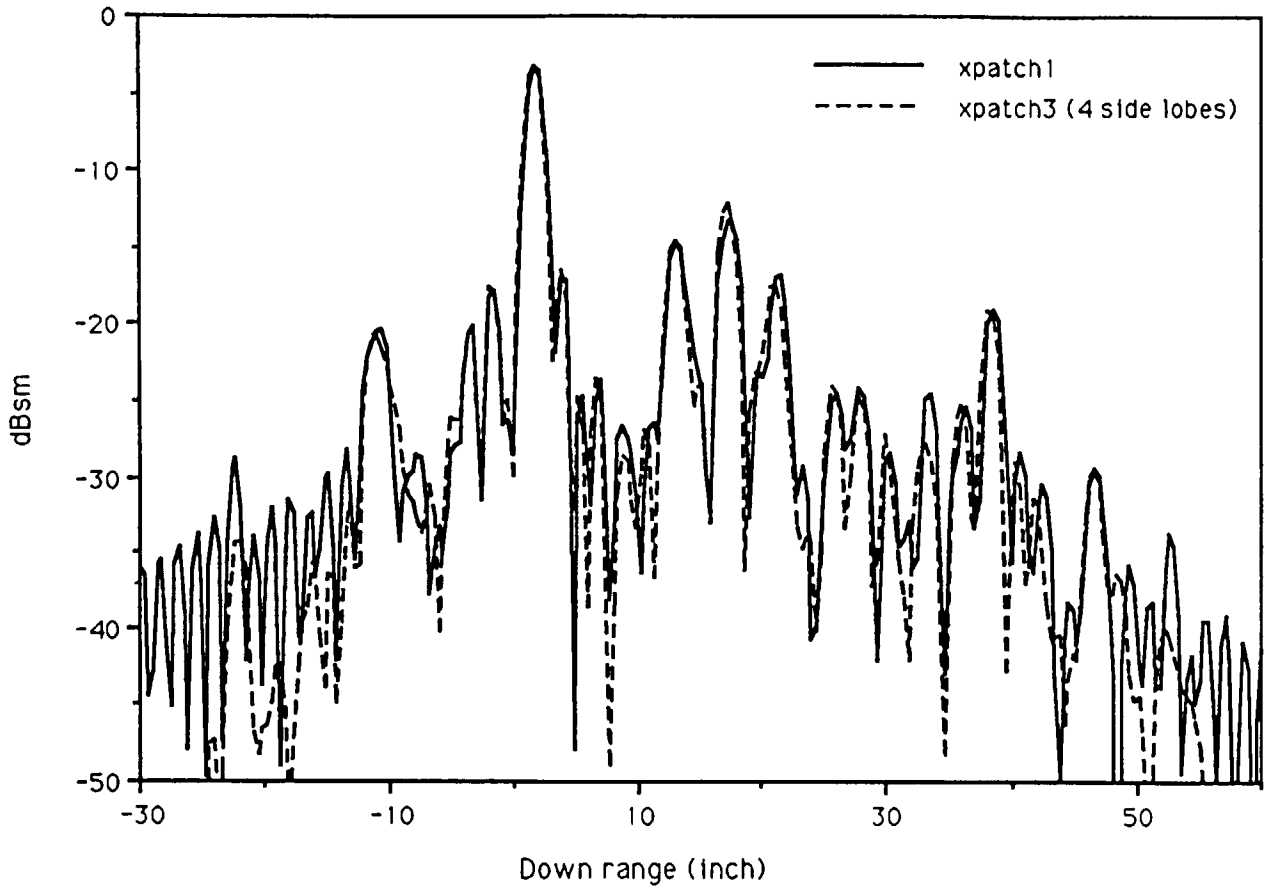


Fig. 5. Plot of W(t) versus time for a center frequency of 10 GHz and bandwidth of 4 GHz. W(t) is not significantly different from a sinc function.

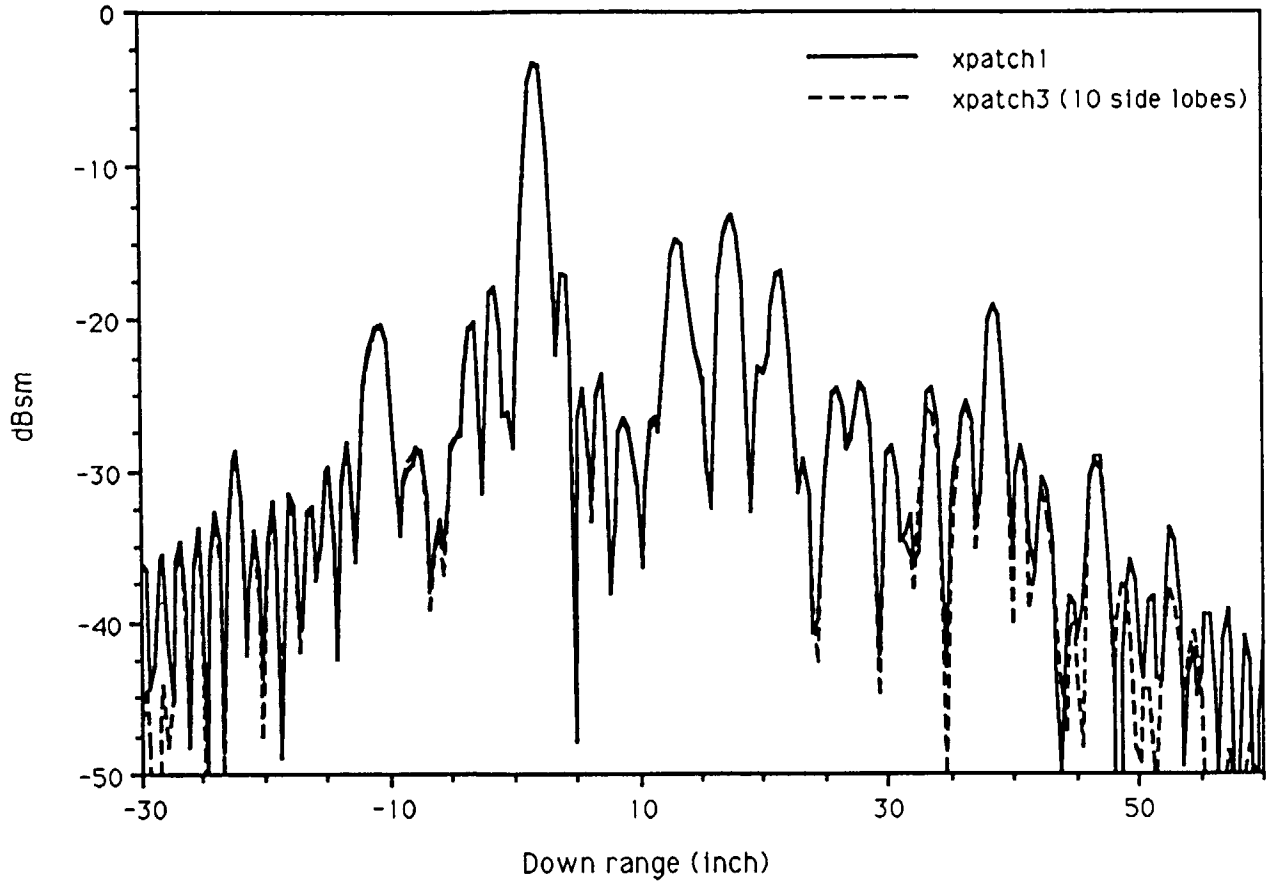
There is a tradeoff between accuracy and computation time



Computation time: xpatch1 - 7.4 min
xpatch3 - 3.6 min

Fig. 6. Comparison of range profile of turkey using xpatch1 and xpatch3. xpatch3 range bins are updated till 4 sidelobes of W(t).

10 side lobes give a good range profile match



Computation time : xpatch1 - 9.7 min
xpatch3 - 4 min

Fig. 7. Comparison between the range profile of turkey using xpatch1 and xpatch3. xpatch3 range bins are updated till 10 sidelobes of W(t).

Trucation of W(t) in the time domain results in edge noise and spillover in the frequency domain.

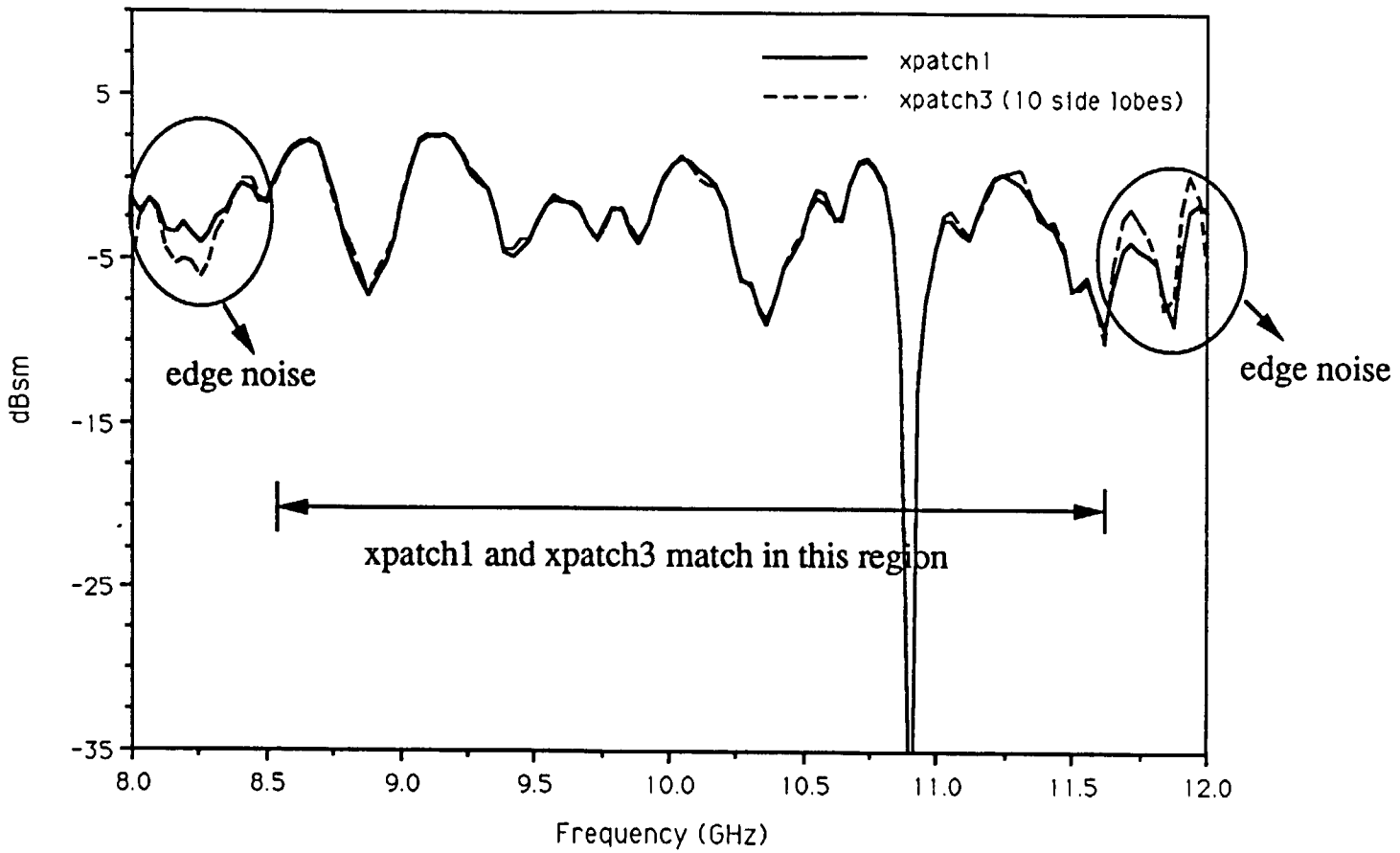
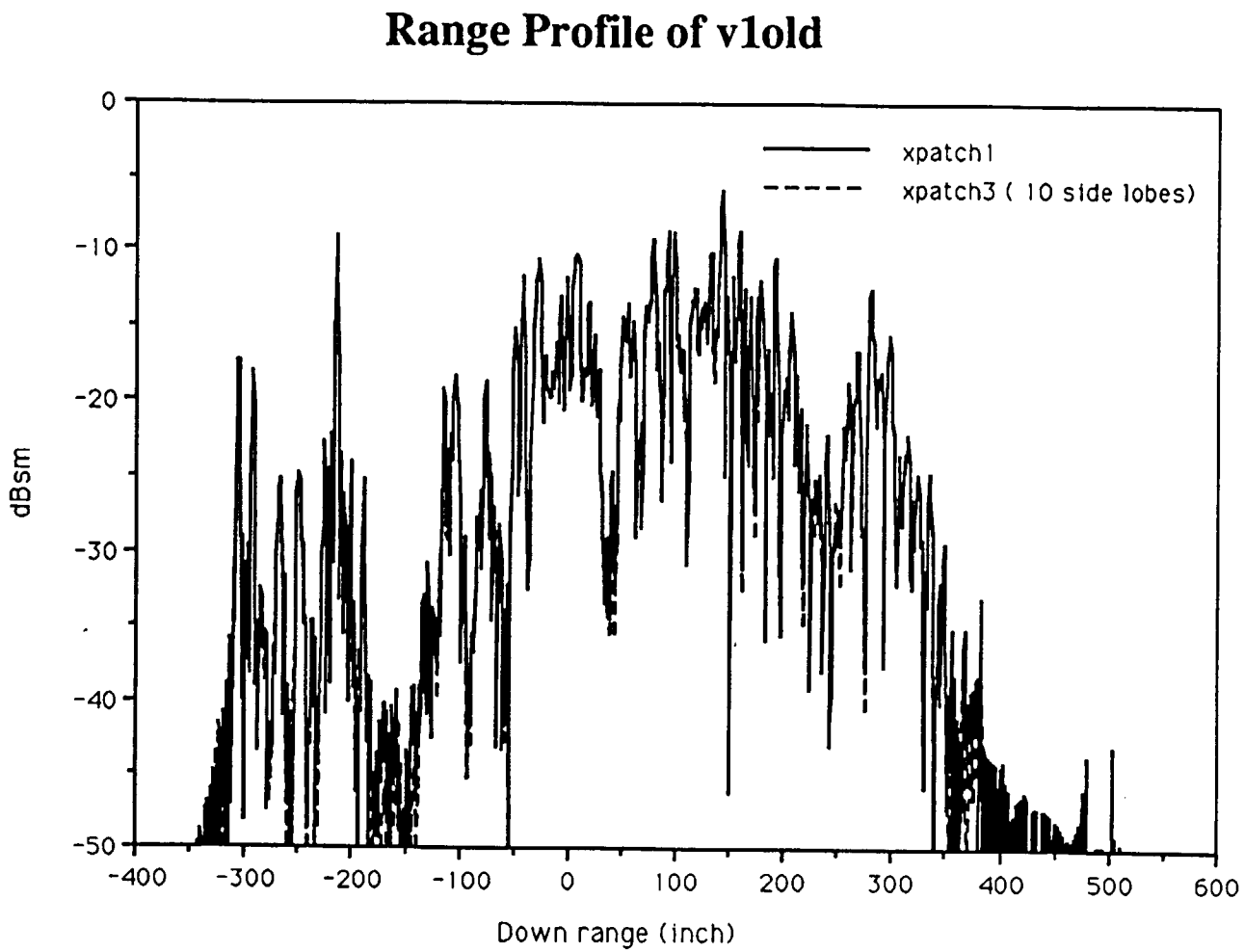


Fig. 8. Truncation of W(t) results in Gibbs phenomenon and xpatch1 and xpatch3 RCS do not match. Windowing is not helpful due to presence of edge noise.

domain. However, when W(t) is truncated in the time domain, the range profile is truncated to zero outside the range where rays can arrive. Consequently, the edge noise becomes significant. We have attempted to use windowing in the frequency domain to alleviate this problem but the results were not satisfactory.

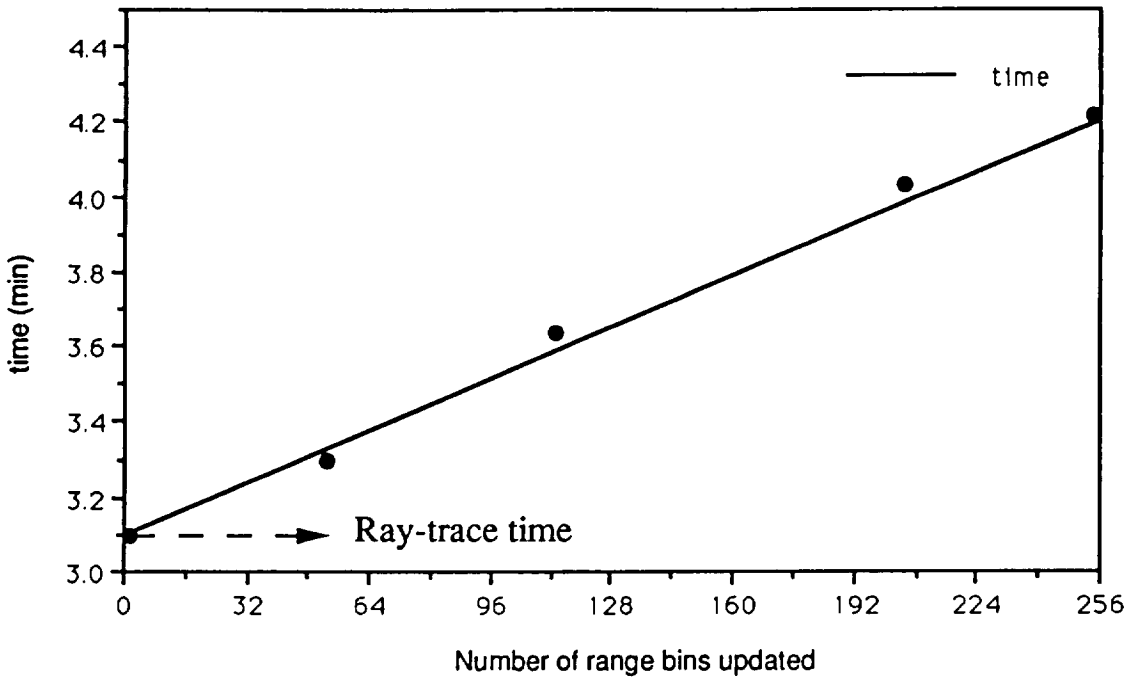
Fig. 9 shows the range profile of "v1old" computed by truncating W to 10 sidelobes. The computation time for Xpatch3 is reduced from 1.6 hours to 1.45 hours. The optimum choice of the number of sidelobes to include in the computation of the range profiles will depend on the user's requirement for accuracy and time, since these two requirements are contradictory. A timing study for "turkey" is shown in Fig. 10. The more we truncate W(t), the fewer the number of bins to be updated, and the smaller the overall computation time. We observe the expected linear variation between the number of bins to be updated and the overall computation time. It is up to the user to decide how much accuracy can be sacrificed for the reduction in computation time. The role of the present truncation approximation in Xpatch3 can perhaps be best described as being analogous to the role of the Nausbaum method in Xpatch1. The reduction in run time is achieved at the expense of some degradation in accuracy. This analogy is indicated in Table 2.



Computation time: xpatch1 - 7.4 hrs.
xpatch3 - 1.45 hrs.

Fig. 9. Comparison of range profile of v1old computed from xpatch1 and xpatch3. In xpatch3 W(t) has been trucated to 10 side lobes.

There is a linear dependence between the number of range bins to be updated and total computation time.



TOTAL COMPUTATION TIME BREAKDOWN

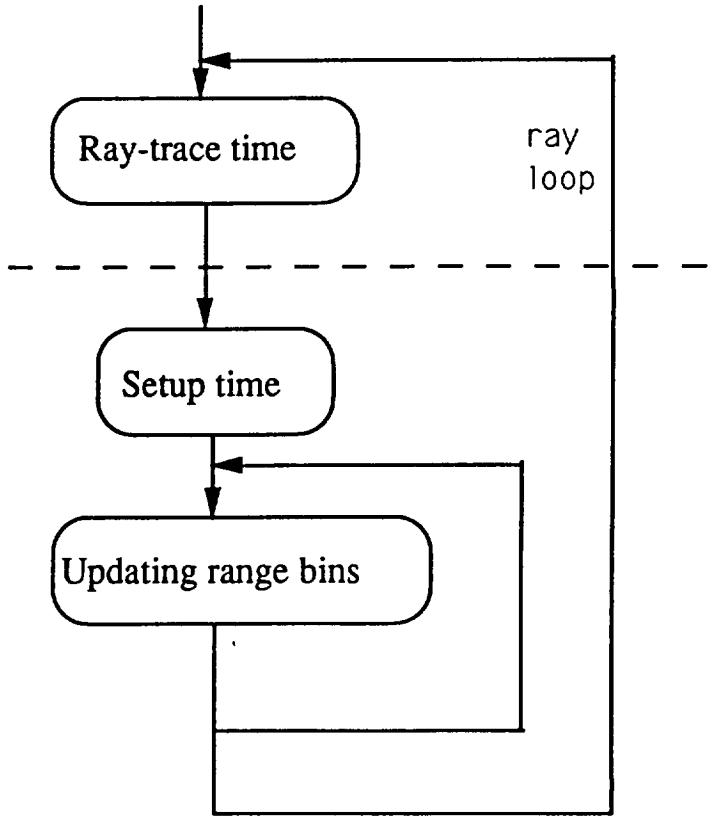


Fig. 10. Timing study for the turkey.

5. CONCLUSION

In this report, we presented a rederivation of the time-domain ray-tube integration scheme proposed by Jeng and Lee [1]. Our new derivation and subsequent implementation of the new formula corrected the inaccuracies which exist in the current version of Xpatch3, Version 1. Using the new corrected Xpatch3, exact agreement with the data generated by Xpatch1 was achieved for the target "turkey" and "v1old". In addition, we showed time improvement on the Silicon Graphics for "v1old" from 7.4 hours using Xpatch1 (all bounces by SBR) to 1.6 hours using Xpatch3. Further time improvements in Xpatch3 can be anticipated if the first bounce contribution can be computed by time domain physical optics utilizing the z-buffer. In addition, if some accuracy can be sacrificed, further computation time reduction can be achieved by truncating the ray contribution function W in the time domain. This truncation will result, however, in edge noise in the frequency domain.

REFERENCES

[1] S. K. Jeng and S. W. Lee, "A time-domain SBR technique for range-profile computation," pre-print, March 1993.

[2] S. W. Lee, H. Ling, R. Chou, "Ray tube integration in shooting and bouncing ray method," *Microwave Opt. Tech. Lett.*, vol. 1, pp. 286-289, Oct. 1988.

Image-Domain Ray-Tube Integration Formula for the Shooting and Bouncing Ray Technique

- Fast ISAR Image Simulation Using Xpatch3 -

Rajan Bhalla and Hao Ling

Department of Electrical and Computer Engineering
The University of Texas at Austin
Austin, TX 78712-1084

July 1993

Air Force Wright Laboratory
through NASA Grant NCC 3-273

IMAGE-DOMAIN RAY-TUBE INTEGRATION FORMULA FOR THE SHOOTING AND BOUNCING RAY TECHNIQUE

Rajan Bhalla and Hao Ling

Department of Electrical and Computer Engineering
The University of Texas at Austin
Austin, TX 78712-1084

Abstract A simple image-domain ray-tube integration formula is presented to efficiently compute the inverse synthetic aperture radar (ISAR) image of a complex target by the shooting and bouncing ray (SBR) technique. Contrary to the conventional approach where the ISAR image is obtained by inverse Fourier transforming the computed scattered field data over frequency and aspect, this new formula gives the contribution of each ray to the overall ISAR image directly. Under the small angle approximation and utilizing the bistatic-monostatic equivalence, the image-domain ray-tube integration formula is determined in closed form. Simulation results using the SBR-based code "Xpatch" show that the direct image domain method results in good image quality and superior time performance when compared to the conventional frequency-aspect approach.

This work was supported by NASA Grant NCC3-273 and in part by the Joint Services Electronics Program.

1. INTRODUCTION

The shooting and bouncing ray (SBR) method is a high frequency electromagnetic simulation technique for predicting the radar returns from realistic aerospace vehicles and the scattering by complex media [1]-[4]. The basic idea behind the SBR method is very simple. Given the geometrical description of a target, a large set of geometrical optics rays is shot towards the target (Fig. 1). Rays are traced according to the laws of geometrical optics as they bounce around the target. At the exit point of each ray, a ray-tube integration is performed to sum up its contribution to the total scattered field [5]. While the basic idea behind the SBR methodology is simple, when combined with CAD tools for geometrical modeling and fast ray-tracing algorithms developed in computer graphics, this technique becomes a very general tool for characterizing the scattering from large, complex targets. One such development is the general-purpose SBR code, Xpatch [6], [7], which is currently used by the Air Force and the aerospace industry in programs related to target identification and low-observable vehicle design. In this work, we will address the issue of fast inverse synthetic aperture radar (ISAR) image formation using SBR-based codes such as Xpatch.

The ISAR image of a target is a powerful visualization tool that is often used to pinpoint key scattering centers on a target in radar signature studies [8]. ISAR image formation is typically achieved by utilizing the monostatic scattered field data, obtained through either measurement or numerical simulation, over a finite range of look angles and frequencies. From numerical simulation point of view, this is usually a rather time-consuming process. For SBR-based calculations, the angular scan is a particularly expensive operation, since for every new look angle on the target a new set of rays must be launched and traced through the target. To alleviate this problem, we have recently utilized bistatic scattered field data to obtain the ISAR image [9]. Since ray tracing is performed once for the incident direction and only the ray-tube integration is needed for every look angle, the bistatic image formation scheme results in time savings. In a separate

development, a closed form time-domain ray-tube integration formula was recently derived for the fast computation of the time-domain response (or range profile) of a conducting target [10],[11]. This formula gives the explicit contribution of each exit ray in the time domain. Therefore, by summing the contribution from each ray in the time domain directly, the overall time domain response of the target can be obtained without resorting to any multi-frequency calculations:

$$E^s(t) = \sum_{i \text{ rays}} E_i^s(t) \tag{1}$$

where $E_i^s(t)$ is the closed form time-domain formula for each ray given in [11]. In this work, we shall show that the time domain concept can be further extended, with the aid of the bistatic imaging scheme in [9], to directly compute, in *closed form*, the contribution of each ray to the overall ISAR image of a target.

To illustrate our idea, we first note that in the conventional image formation process, the scattered field must be computed over a band of frequencies and angles. Subsequently, by inverse Fourier transforming the resulting two-dimensional data, the ISAR image of the target, $O(x,z)$, can be obtained [12],[13]. This concept is described by the following expression:

$$O(x,z) = \text{F.T.}^{-1} \{ \sum_{i \text{ rays}} E_i^s(\omega,\theta) \} \tag{2}$$

where the quantity in the parentheses is the total scattered field at frequency ω and aspect θ and is obtained through the summation over all exit rays. By interchanging the order of the inverse Fourier transform and the ray summation, the ISAR image can be formed by:

$$O(x,z) = \sum_{i \text{ rays}} O_i(x,z) \tag{3}$$

where $O_i(x,z)$ is the contribution of each exit ray to the ISAR image. It is $O_i(x,z)$ for which we shall derive a closed form expression. The detailed derivation of this formula

will be presented in Section 2. In Section 3 simulation results using the new image-domain method will be compared against the conventional frequency-aspect approach for various targets. Time performance between the two methods will also be discussed.

2. RAY-TUBE INTEGRATION FORMULA

Before deriving the new image-domain formula, we will first summarize the frequency-domain ray-tube integration formula previously derived in [5]. The scattered far field from a target in the frequency domain at an observation point (r,θ,ϕ) can be written as:

$$\mathbf{E}^s(\omega,\theta,\phi) = \frac{e^{-jkr}}{r}\left(\hat{\theta}A_\theta + \hat{\phi}A_\phi\right) \tag{4}$$

where $k=\omega/c$. When the SBR method is used, the contribution of the exiting rays to the scattered field is given by:

$$\begin{bmatrix} A_\theta \\ A_\phi \end{bmatrix} = \sum_{\text{i rays}} \begin{bmatrix} B_\theta \\ B_\phi \end{bmatrix} \left(\frac{jk}{2\pi}\right) (\Delta A)_{exit}\, S(\theta,\phi)\; e^{jk\cdot r_A} \tag{5}$$

In the above expression, $\mathbf{r}_A$ is the position vector of point A where the ray-tube integration is carried out. Point A is usually chosen to be the last hit point on the target for the ray (see Fig. 2). $(\Delta A)_{exit}$ is the cross section of the exit ray tube at A and $S(\theta,\phi)$ is the shape function corresponding to the radiation pattern from the ray tube. $S(\theta,\phi)$ can usually be assumed to be unity if the ray tube area is sufficiently small, since the radiation from the ray tube will be nearly isotropic. B_θ, B_ϕ are explicitly related to the aperture fields at A as:

$$\begin{aligned} B_\theta = 0.5\big[& -s_1\cos\phi\, E_3 - s_2\sin\phi\, E_3 + s_3(\cos\phi\, E_1 + \sin\phi\, E_2)\big] \\ & + 0.5\, Z_o\big[s_1(\cos\theta\sin\phi\, H_3 + \sin\theta\, H_2) \\ & + s_2(-\sin\theta\, H_1 - \cos\theta\cos\phi\, H_3) + s_3(\cos\theta\cos\phi\, H_2 - \cos\theta\sin\phi\, H_1)\big] \end{aligned} \tag{6}$$

$$\begin{aligned} B_\phi = 0.5\big[& s_1(\cos\theta\sin\phi\, E_3 + \sin\theta\, E_2) + s_2(-\sin\theta\, E_1 - \cos\theta\cos\phi\, E_3) \\ & + s_3(\cos\theta\cos\phi\, E_2 - \cos\theta\sin\phi\, E_1)\big] \\ & + 0.5\, Z_o\big[s_1\cos\phi\, H_3 + s_2\sin\phi\, H_3 + s_3(-\cos\phi\, H_1 - \sin\phi\, H_2)\big] \end{aligned} \tag{7}$$

where $\mathbf{E(A)} = E_1\hat{x} + E_2\hat{y} + E_3\hat{z}$ and $\mathbf{H(A)} = H_1\hat{x} + H_2\hat{y} + H_3\hat{z}$ are respectively the electric and magnetic field associated with each ray at A, and $\hat{s} = s_1\hat{x} + s_2\hat{y} + s_3\hat{z}$ is the exit ray direction.

To derive an image-domain ray-tube integration formula, we will first choose the image plane as the x-z plane as shown in Fig. 3. Note that this is done without any loss of generality since the target can always be rotated to conform to the present image coordinates. Because it is computationally expensive to generate monostatic data using the SBR method, the bistatic imaging scheme is preferred [9]. We will collect bistatic scattered field data about the z-axis, i.e., with the incident wave from the -z direction and with a series of observations made at $\phi = 0$ and over a set of small look angles about $\theta = 0$. Under the bistatic scenario where the incident direction is fixed, the ray paths and the associated ray fields remain unchanged for the different observations angles. Furthermore, for small θ we can approximate $\cos\theta \approx 1$ and $\sin\theta \approx \theta$ and B_θ, B_ϕ reduce to:

$$B_\theta = (-s_1E_3 + s_3E_1 - s_2Z_oH_3 + s_3Z_oH_2) + \theta(s_1Z_oH_2 - s_2Z_oH_1) \quad (8)$$

$$B_\phi = (-s_2E_3 + s_3E_2 + s_1Z_oH_3 - s_3Z_oH_1) + \theta(s_1E_2 - s_2E_1) \quad (9)$$

Assuming the target is perfectly conducting we can factor out the explicit frequency dependence in B_θ, B_ϕ as follows:

$$B_\theta = ((-s_1E_3' + s_3E_1' - s_2Z_oH_3' + s_3Z_oH_2') + \theta(s_1Z_oH_2' - s_2Z_oH_1'))\, e^{-jkd_i} \quad (10)$$

$$B_\phi = ((-s_2E_3' + s_3E_2' - s_1Z_oH_3' + s_3Z_oH_1') + \theta(s_1E_2' - s_2E_1'))\, e^{-jkd_i} \quad (11)$$

where d_i is the total distance traveled by the i'th ray to the last hit point on the target and the primed field quantities have no frequency dependence. From eqns. (10) and (11) we see that B_θ, B_ϕ take on the form

$$B_{\theta,\phi} = (\alpha + \theta\beta)\, e^{-jkd_i} \quad (12)$$

In the rest of the derivation we will use the form in eqn. (12) to represent $B_{\theta,\phi}$, with the subscript denoting the appropriate polarization.

With the explicit angular and frequency dependence of the scattered field in hand, we will now proceed to the imaging algorithm. The ISAR image and the bistatic scattered field are related through a two-dimensional Fourier transform [9]:

$$O_{\theta,\phi}(x,z) = \frac{1}{(2\pi)^2} \iint_S \tilde{O}_{\theta,\phi}(k_x,k_z)\, e^{-jk_x x}\, e^{-jk_z z}\, dk_x\, dk_z \tag{13}$$

where $O_{\theta,\phi}(x,z)$ is the ISAR image of the target and $\tilde{O}_{\theta,\phi}(k_x,k_z)$ is the range-corrected scattered field given by

$$\tilde{O}_{\theta,\phi}(k_x,k_z) = \frac{4\pi r}{-jk\, e^{-jkr}} E^s_{\theta,\phi} \tag{14}$$

Under the bistatic condition, the two-dimensional k-space integration in (13) is performed over the shaded area S shown in Fig. 4. The k-space is accessed by stepping the frequency from k_{min} to k_{max} and the observation angle from $-\theta_o$ to θ_o. By substituting eqns. (4), (5) and (14) in eqn. (13) we get (with $S(\theta,\phi)$ set to 1):

$$O_{\theta,\phi}(x,z) = -\frac{1}{2\pi^2} \sum_{\text{i rays}} \iint_S B_{\theta,\phi}\, (\Delta A)_{\text{exit}}\; e^{j\mathbf{k}\cdot\mathbf{r}_A}\, e^{-jk_x x} e^{-jk_z z}\, dk_x\, dk_z \tag{15}$$

We will next evaluate the double integration in closed form under the small θ approximation. As is shown in Fig. 4, under the small-angle imaging condition the area S is nearly rectangular, and k_x and k_z can be approximated as $k_x \approx k_o\theta$, and $k_z \approx 2k$, where $k_o = (k_{max} + k_{min})/2$. The differentials can hence be written as $dk_x = k_o d\theta$ and $dk_z = 2dk$. Similarly, the wave vector $\mathbf{k}$ can be approximated as $\mathbf{k} = k\left[(\hat{x}\cos\phi + \hat{y}\sin\phi)\sin\theta + \hat{z}\cos\theta\right] \approx k_o\theta\hat{x} + k\hat{z}$. Consequently, we can rewrite $\mathbf{k}\cdot\mathbf{r}_A \approx k_o\theta\, x_i\hat{x} + k\, z_i\hat{z}$ where (x_i, z_i) are the coordinates of the last hit point on the target. Substituting these approximations and eqn. (12) into eqn. (15) we get:

$$O_{\theta,\phi}(x,z) = -\frac{k_o}{\pi^2}\sum_{i\ rays}(\Delta A)_{exit}\left(\int_{k_{min}}^{k_{max}} e^{-jk(2z+d_i-z_i)}\,dk\right)\left(\int_{-\theta_o}^{\theta_o}(\alpha+\theta\beta)\,e^{-jk_o\theta(x-x_i)}\,d\theta\right) \qquad (16)$$

The above integrations can be easily carried out analytically. Denoting sinc (u) = sin(u)/u and sinc'(u) = d(sinc(u))/du = (-sinc(u) + cos(u))/u, we arrive at a closed form expression for the ISAR image of the target in terms of the exit ray field:

$$O_{\theta,\phi}(x,z) = -\frac{k_o}{\pi^2}\sum_{i\ rays}(\Delta A)_{exit}\left\{\Delta k\, e^{-jk_o(2z+d_i-z_i)}\,\mathrm{sinc}\left[\Delta k\left(z+\frac{d_i-z_i}{2}\right)\right]\right\}\cdot$$

$$\cdot\left\{2\alpha\theta_o\,\mathrm{sinc}[k_o\theta_o(x-x_i)] + 2j\beta\theta_o^2\,\mathrm{sinc}'[k_o\theta_o(x-x_i)]\right\} \qquad (17)$$

The above expression is in the form of (3) and gives the explicit contribution of each ray to the ISAR image. Several remarks are in order: (i) We have utilized the bistatic configuration to arrive at the image-domain formula. The monostatic ISAR image is often the more desirable result. However, for small-angle imaging, the equivalence between the monostatic and bistatic configuration is a simple one, i.e., the ISAR image generated from (17) should be equivalent to the ISAR image generated using monostatic data with the same frequency bandwidth and half the angular window from $-\theta_0/2$ to $\theta_0/2$. Detailed discussions on the image quality generated under the two configurations can be found in [9], [12] and [14]. In Section 3, we will make direct comparisons between the bistatic image generated from (17) and the monostatic image generated from conventional frequency-aspect data. (ii) The quantity in the second parentheses in (17) contains both a sinc and a sinc' term. However, the sinc term is of order θ_0 while the sinc' term is of order θ_0^3. For small angle imaging, the sinc' term can be ignored. As a result, the image domain ray-tube integration formula essentially consists of the product of two sinc functions. In the -z (or down range) direction, the sinc function has its maximum at $-z=(d_i-z_i)/2$ which is half of the total distance traveled by the ray to the far field measured with respect to the origin. The width of the sinc function in range is inversely proportional to

the bandwidth. In the x (or cross range) direction, the sinc function has its maximum at the $x=x_i$ which is the last hit-point on the target. The width of the sinc function in cross range is inversely proportional to $k_0\theta$. The role of this two-dimensional sinc function is very similar to that played by the "point spread function" in synthetic aperture radars [8]. However, in the present context it should more appropriately be called the "ray spread function". Under the present interpretation, the image-domain ray-tube formula becomes quite intuitive. Each exit ray contribute to the overall ISAR image by producing a ripple with its peak at the appropriate range and cross range locations. The range maximum is proportional to the half of the total distance traveled by the ray and the cross range maximum corresponds to the cross range hit point on the target. In the numerical implementation, we will take advantage of the spatial decay nature of the two-dimensional sinc function by only updating those image bins close to the peak for each ray. This will result in tremendous time savings over the conventional method where each ray contributes to the entire frequency-aspect plane. (iii) It is well known that when multiple bounces on the target are present, the monostatic and bistatic images tend to differ [9]. Under the bistatic interpretation, the cross range contribution from a multi-bounce ray is centered around the last hit point x_i on the target. One heuristic way to improve the bistatic image so that it more closely resemble the monostatic image is to use as x_i an average of some or all of the hit point x-coordinates in (17). From our numerical results, using the average of the first and the last hit points seems to work quite well. A more rigorous derivation of the correct monostatic cross-range is currently being investigated.

3. NUMERICAL IMPLEMENTATION AND RESULTS

Numerical implementation of eqn. (17) has been incorporated into the SBR-based code Xpatch3. The implementation strategy is as follows. For a given target, we first choose a range window [-R/2, R/2] and a cross-range window [-L/2, L/2]. Here the zero range and cross-range point corresponds to the origin of the target and (R, L) are chosen to be approximately twice the dimensions of the target. The image plane is then divided into N equally spaced range bins with binwidth $\Delta r=R/N$ and M equally spaced cross-range bins with binwidth $\Delta cr=L/M$. The image plane is discretized with grid points:

$$x_m = \Delta cr\,(m-1) - L/2, \quad m = 1,2,...,M+1 \tag{18}$$

$$z_n = \Delta r\,(n-1) - R/2, \quad n = 1,2,....,N+1 \tag{19}$$

The binwidths should be chosen so as to ensure proper sampling in the range and cross range directions. As a general guideline, the range binwidth should be no greater than (0.15 m) / (bandwidth in GHz) and the cross range binwidth should be no greater than (0.15 m) / (center frequency in GHz) / (θ_0). Using eqns. (18) and (19) in (17) we arrive at the desired expression for numerical implementation:

$$O_{\theta,\phi}(x_m,z_n) = -\frac{k_o}{\pi^2}\sum_{i\ rays}(\Delta A)_{exit}\left\{\Delta k\, e^{-jk_o(2z_n+d_i-z_i)}\,\mathrm{sinc}\left[\Delta k\,(z_n+\frac{d_i-z_i}{2})\right]\right\}\cdot$$

$$\cdot\left\{2\alpha\theta_o\,\mathrm{sinc}\left[k_o\theta_o(x_m-x_i)\right]\right\} \tag{20}$$

In the implementation once the total distance traveled by the ray d_i and the last hit point $\mathbf{r}_A$ for the i'th ray is obtained following the ray tracing, we can update the image plane according to the above formula. In practice it is not necessary to update the entire image plane, since the two-dimensional sinc function appearing in the above formula can be adequately truncated after a few sidelobes. In addition, the sinc function is pre-computed and stored in a lookup table to save computation time. When other forms of ray spread

function is needed (due to different forms of windowing in frequency and aspect), there is little difficulty in constructing the proper table before the computation is carried out.

Several numerical examples are presented next to demonstrate the accuracy and speed of the above formula. The ISAR images generated using this technique are compared to those generated using the conventional Fourier processing of the monostatic frequency-aspect data. The first example is a rhombic cube. The target geometry is shown in Fig. 5(a). To generate the ISAR image using the conventional approach, monostatic scattered field was computed with the frequency scanned from 11 to 16 GHz in 128 steps and the aspect observed from -6° to +6° in 128 steps. The angle is measured from the z-axis in the xz-plane. The ISAR image was also computed using the direct image-domain ray-tube formula on a 128 x 128 grid. We used the the same frequency window, with the incident angle set at 0° and the bistatic observation window of ±12°. The ISAR images from the two schemes are shown in Figs. 5(b) and 5(c). The target geometry is overlaid on the images. These two images bear excellent resemblance to each other. The principal scattering centers corresponding to the three corners are distinctly captured by both images. In addition, the peak magnitude of the scattering centers are in good agreement to within 2 dBsm.

The second example is a 90° dihedral, as shown in Fig. 6(a). The ISAR image was formed using the conventional approach by computing the monostatic scattered field within a frequency window of 11-16 GHz in 128 steps and an aspect window of ± 6° (centered about 35° with respect to one side of the dihedral) in 128 steps. The ISAR image was also computed using the direct image-domain ray-tube formula on a 128 x 128 grid using the same frequency window and an equivalent bistatic observation window of ±12°. To account for the strong multiple scattering effects, we used as the cross range the average of the x-coordinates of the first and the last hit points, as described earlier in Section 2. The ISAR images from the two schemes are shown in Figs. 6(b) and 6(c). Again, excellent agreement between the two images is observed.

In the last example, a complex aircraft is considered. The target is an generic aircraft similar to that shown in Fig. 1. The target model consists of 10,746 planar triangular facets. To generate the ISAR image using the conventional approach, monostatic scattered field was computed with the frequency scanned from 9.5 to 10.5 GHz in 133 steps. The angular look window was ±3° in 124 steps centered about 30° with respect to the nose-on direction of the aircraft (see Fig. 7(a)). The frequency domain data were then zero-padded and processed to generate the 256 x 256 pixel image shown in Fig. 7(b). Fig. 7(c) shows the ISAR image generated using the direct image-domain ray-tube formula on a 256 x 256 grid for the same frequency window. The incident angle was 30° with respect to nose-on and the equivalent bistatic observation window was ± 6°. We observe that the image generated by the direct image-domain ray-tube formula bears very good fidelity to that produced by the conventional method, even for such a highly complex target. Figs. 8(a)-(c) show a different look angle on the same target at 120° with respect to the nose-on direction of the aircraft. Again, the image generated by the direct image-domain method agrees very well with that produced by the conventional approach.

The computation time to form each image using the direct image-domain method for the generic aircraft is approximately 4 hours on our Silicon Graphics Iris Indigo R3000 workstation. This computation time is obtained using a ray density of 10 rays per wavelength and 50 maximum bounces. It is orders of magnitude faster than using the conventional image formation scheme. To generate the monostatic frequency-aspect data needed in the conventional scheme, we estimate that it would have taken over 300 hours of computation time on the Indigo. The actual data were generated on an Intel iPSC/860 hypercube at Sandia National Laboratory. There are two factors which give rise to superior time performance of the direct image-domain scheme. First, the bistatic approximation allows the ray tracing to be performed for one incident angle only. This advantage has already been explored in [9]. Second, with the newly derived image-domain formula, only a small portion of the image plane near the peak of the ray spread function

needs to be updated for each ray. In the complex aircraft example, to achieve a 30-dB dynamic range in the ISAR image, only a 20x20 grid (corresponding to approximately 5 sidelobes) needs to be updated in the image plane for each ray, as oppose to the entire 256x256 grid. This is in contrast to the conventional data collection in the frequency-aspect plane where each ray contributes to every frequency and every angle in general.

4. CONCLUSION

In this paper we derived an image-domain ray-tube integration formula. It was shown that using this formula the ISAR images of targets can be generated directly in the image domain without resorting to any multiple frequency-aspect calculations. The performance of the direct image-domain scheme was evaluated by comparing the ISAR images generated by using the conventional frequency-aspect approach to those generated using the new formulation. Excellent agreement was observed between the images obtained from the new scheme and the conventional approach. The image simulation time of the new scheme is orders of magnitude faster than the conventional frequency-aspect approach. The reasons for the time savings are twofold. First, the image-domain formula takes advantage of bistatic data and does not require calculation of monostatic scattered field at multiple look angles. Second, the image-domain formula is spatially limited in the range/cross-range plane and only the contribution near the peak of the ray spread function needs to be taken into account. The new image generation scheme is limited to small-angle imaging scenarios and to perfect conducting targets. When these conditions are met, the tremendous time advantage makes this new scheme the much preferred method for ISAR image simulation.

ACKNOWLEDGEMENTS

The authors are grateful to Prof. S. W. Lee for providing the various versions of Xpatch and for many helpful discussions. The authors would also like to thank Captain D. J. Andersh for providing the monostatic Xpatch simulation data and for his continued support of this work.

REFERENCES

[1] H. Ling, R. Chou and S. W. Lee, "Shooting and bouncing rays: calculating the RCS of an arbitrary shaped cavity," *IEEE Trans. Antennas Propagat.*, vol. AP-37, pp. 194-205, Feb. 1989.

[2] J. Baldauf, S. W. Lee, L. Lin, S. K. Jeng, S. M. Scarborough and C. L. Yu, "High frequency scattering from trihedral corner reflectors and other benchmark targets: SBR versus experiment," *IEEE Trans. Antennas Propagat.*, vol. AP-39, pp. 1345-1351, Sept. 1991.

[3] H. Ling, H. Kim, G. A. Hallock, B. W. Birkner and A. Zaman, "Effect of arcjet plume on satellite reflector performance," *IEEE Trans. Antennas Propagat.*, vol. AP-39, pp. 1412-1420, Sept. 1991.

[4] H. Kim and H. Ling, "Electromagnetic scattering from an inhomogeneous body by ray tracing," *IEEE Trans. Antennas Propagat.*, vol. AP-40, pp. 517-525, May 1992.

[5] S. W. Lee, H. Ling, R. Chou, "Ray tube integration in shooting and bouncing ray method," *Microwave Opt. Tech. Lett.*, vol. 1, pp. 286-289, Oct. 1988.

[6] S. W. Lee, "Test cases for Xpatch," Electromagnetics Lab. Tech. Rept. ARTI-92-4, Univ. of Illinois, Feb. 1992.

[7] S. W. Lee and D. J. Andersh, "Xpatch: a high frequency RCS code", 9th Annual Review of Progress in Applied Computational Electromagnetics, Monterey, CA, March 1993.

[8] D. L. Mensa, *High Resolution Radar Imaging*. Artech House, Dedham, MA, 1981.

[9] R. Bhalla and H. Ling, "ISAR image formation using bistatic computed from shooting and bouncing ray technique," to appear in *J. of Electromag. Waves and Appl.*, 1993.

[10] S. K. Jeng and S. W. Lee, "A time-domain SBR technique for range-profile computation," pre-print, March 1993.

[11] H. Ling and R. Bhalla, "Time-domain ray-tube integration formula for the shooting and bouncing ray technique," Tech. Rept., Univ. of Texas, April 1993.

[12] N. H. Farhat, C. L. Werner and T. H. Chu, "Prospects of three-dimensional projective and tomographic imaging radar networks," *Radio Sci.,* vol. 19, pp. 1347-1355, Sept.-Oct. 1984.

[13] H. J. Li and N. H. Farhat, "Image understanding and interpretation in microwave diversity imaging," *IEEE Trans. Antennas Propagat.,* vol. AP-37, pp. 1048-1057, Aug. 1989.

[14] J. F. Shaeffer, "MOM3D method of moments code theory manual," NASA Contractor Report 189594, March 1992.

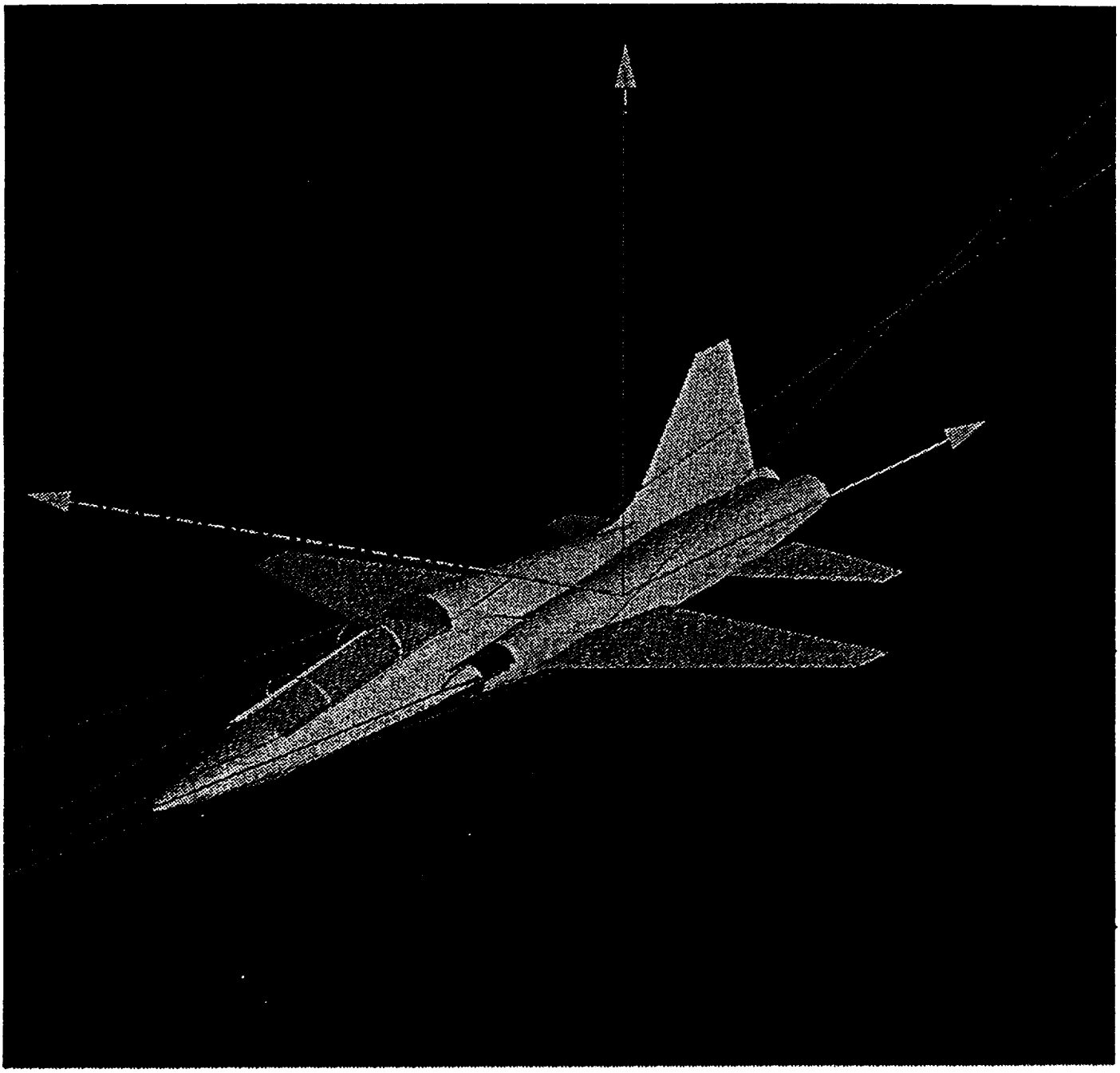


Fig. 1. The Shooting and Bouncing Ray (SBR) method is a high frequency simulation technique used for predicting the radar cross section of complex targets. Rays are launched from the incident direction toward the target. The scattered field is computed by summing the exit ray contributions.

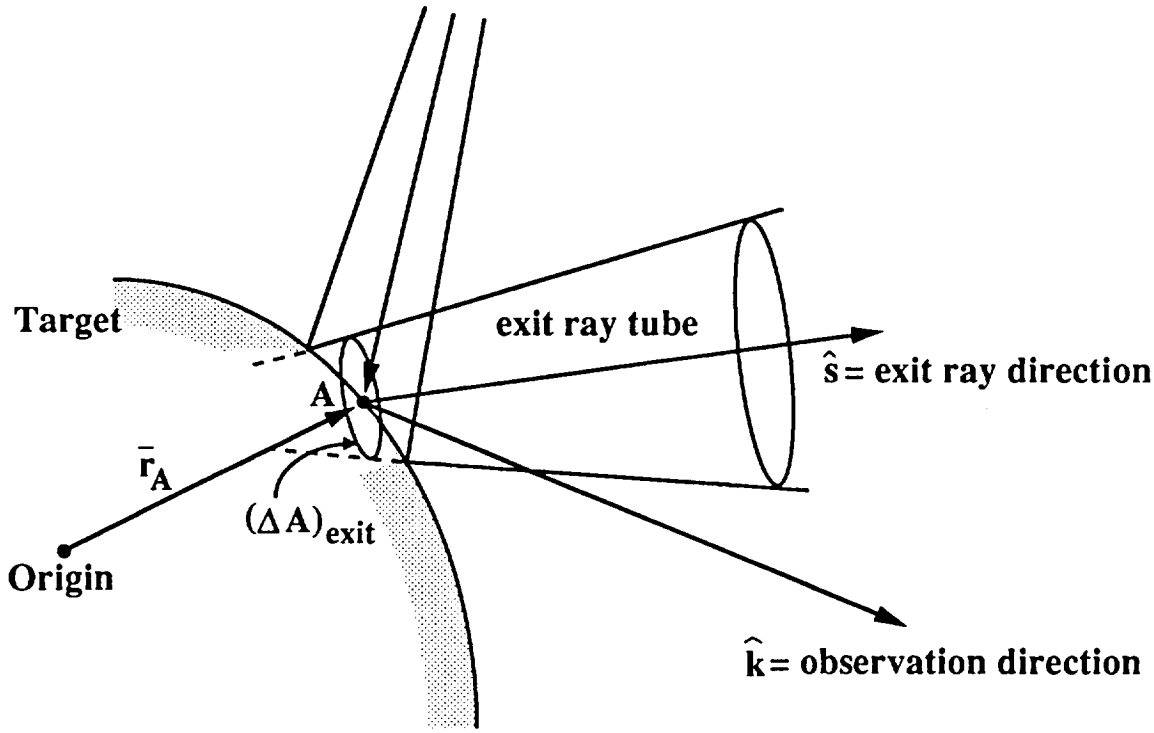


Fig. 2. The ray-tube integration is carried out at the last hit point on the target (point A).

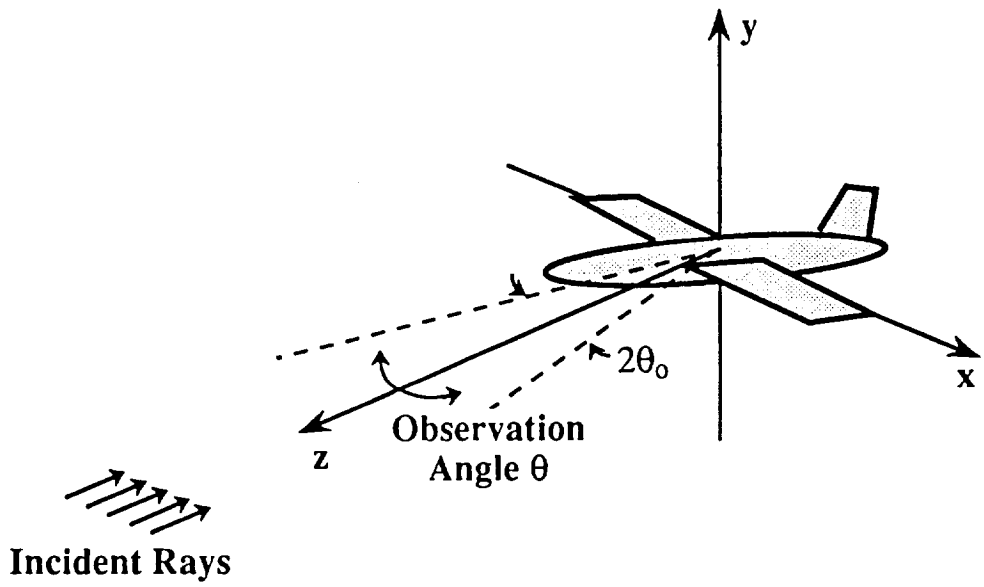


Fig. 3. In the ISAR imaging coordinate, rays are incident from the -z direction. The observation directions are centered about the z-axis in the xz-plane.

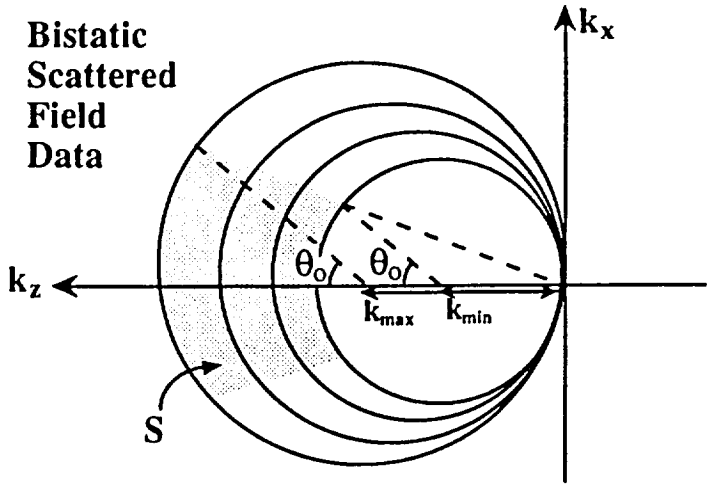


Fig. 4. The shaded area S represents the Fourier space data collected under the bistatic scattering arrangement. The frequency is stepped from k_{min} to k_{max} and the observation angle is stepped from $-\theta_o$ to θ_o.

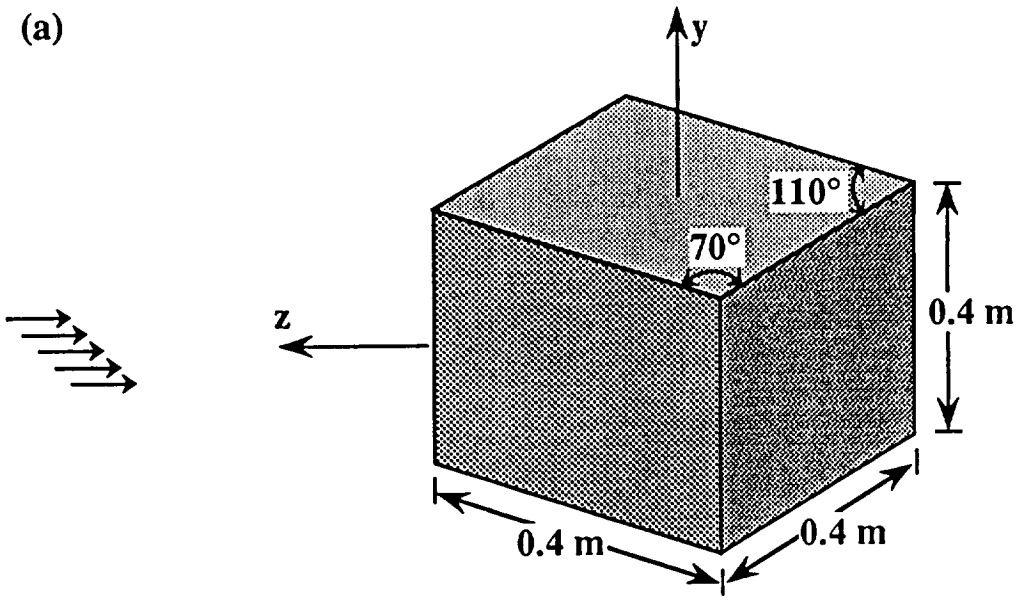
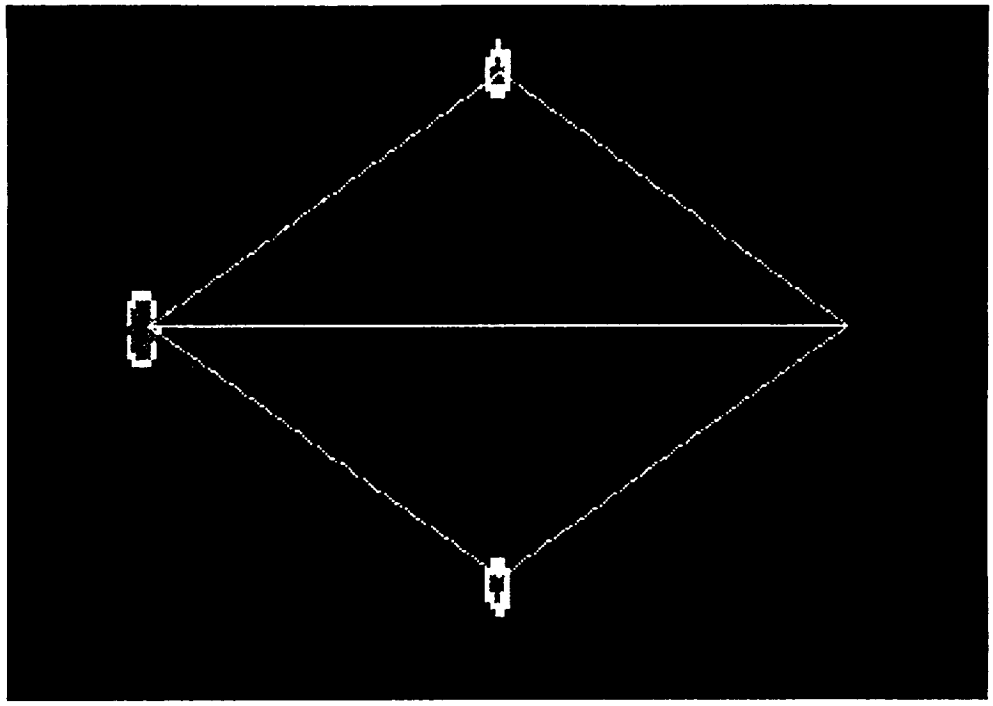


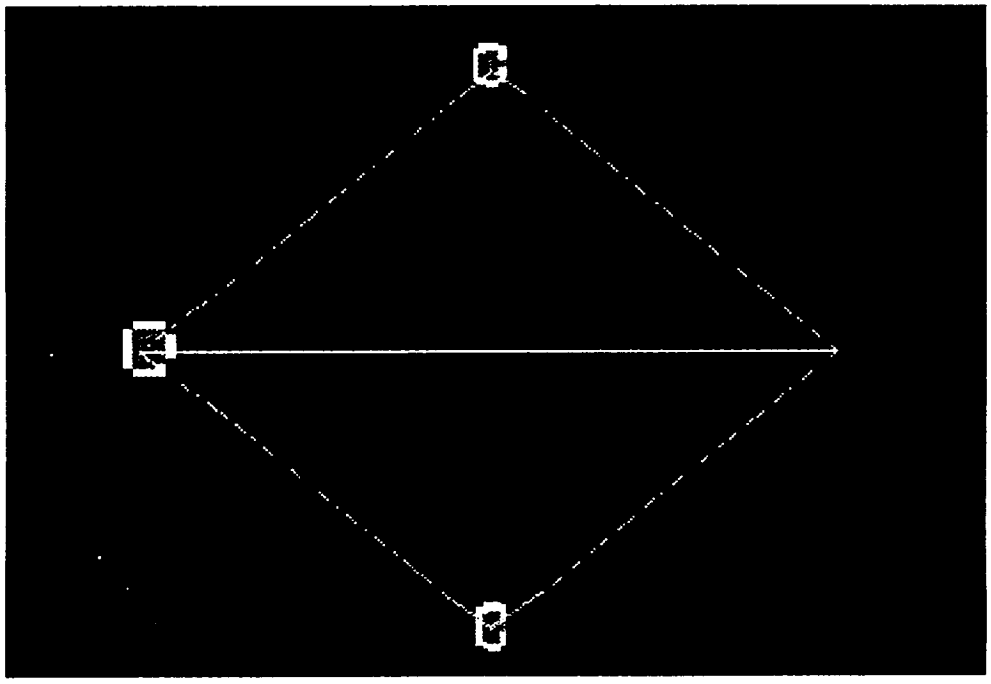
Fig. 5. The ISAR images of a rhombic cube. The images were generated for a frequency scan from 11 to 16 GHz and a monostatic angular scan from -6° to 6°.

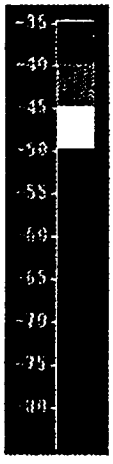
(a) Target geometry.
(b) Conventional Fourier processing of monostatic frequency-aspect data.
(c) Direct image-domain ray-tube integration formula.

(b) Conventional frequency–aspect approach



(c) Direct image–domain ray–tube integration formula





dBsm level

Fig. 5. The ISAR images of a rhombic cube. The images were generated for a frequecncy scan from 11 to 16 GHz and a monostatic angular scan from -6^{o} to 6^{o}.
(a) Target geometry.
(b) Conventional Fourier processing of monostatic frequency–aspect data.
(c) Direct image–domain ray–tube integration formula.

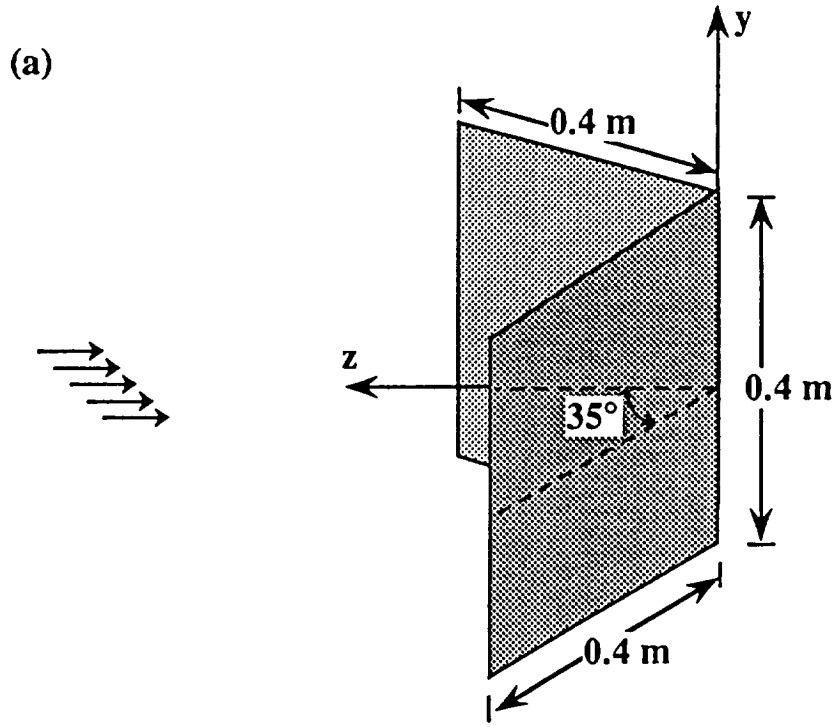
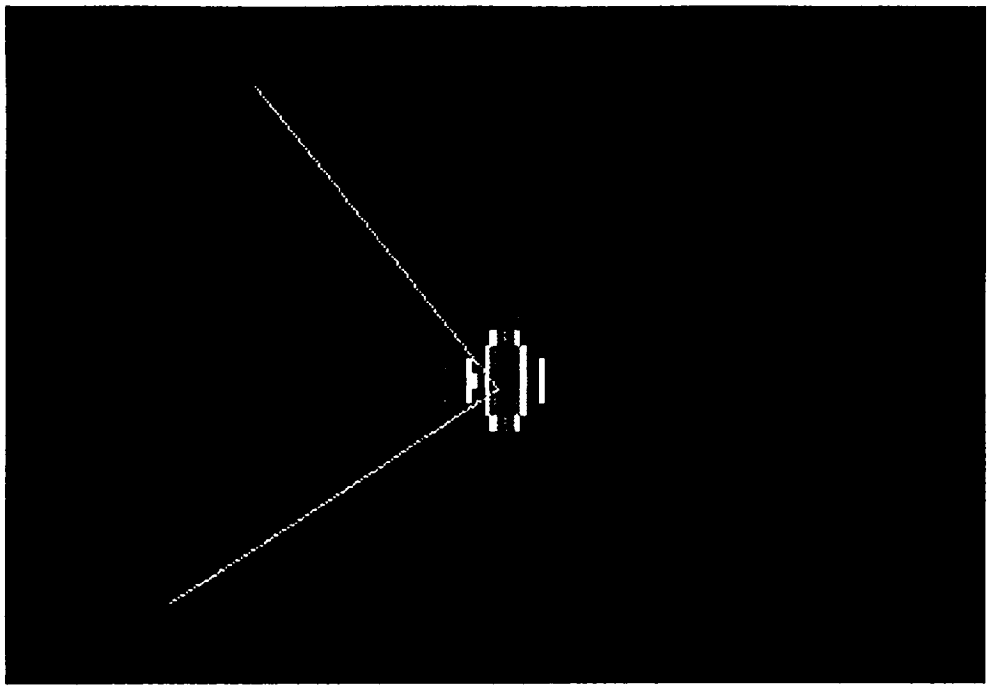


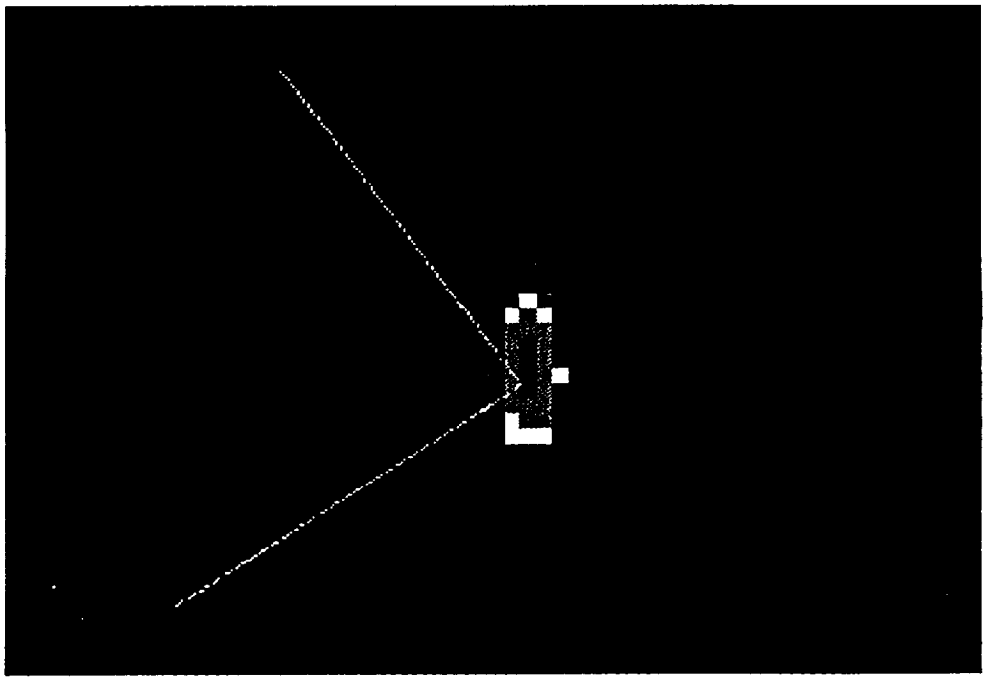
Fig. 6. The ISAR images of a 90° dihedral. The images were generated for a frequency scan from 11 to 16 GHz and a monostatic angular scan from -6° to 6°.

(a) Target geometry.
(b) Conventional Fourier processing of monostatic frequency-aspect data.
(c) Direct image-domain ray-tube integration formula.

(b) Conventional frequency–aspect approach



(c) Direct image–domain ray–tube integration formula



dBsm level

Fig. 6. The ISAR images of a 90° dihedral. The images were generated for a frequecncy scan from 11 to 16 GHz and a monostatic angular scan from –6° to 6°.
(a) Target geometry.
(b) Conventional Fourier processing of monostatic frequency–aspect data.
(c) Direct image–domain ray–tube integration formula.

(a)

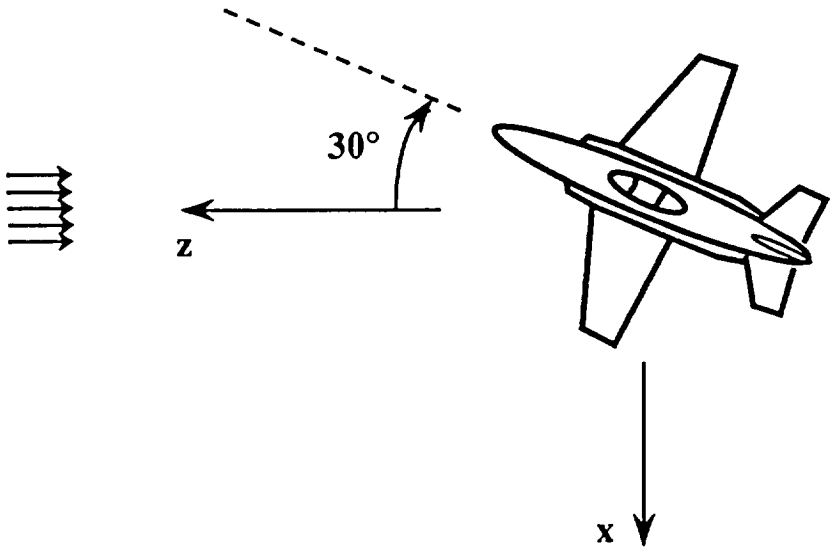
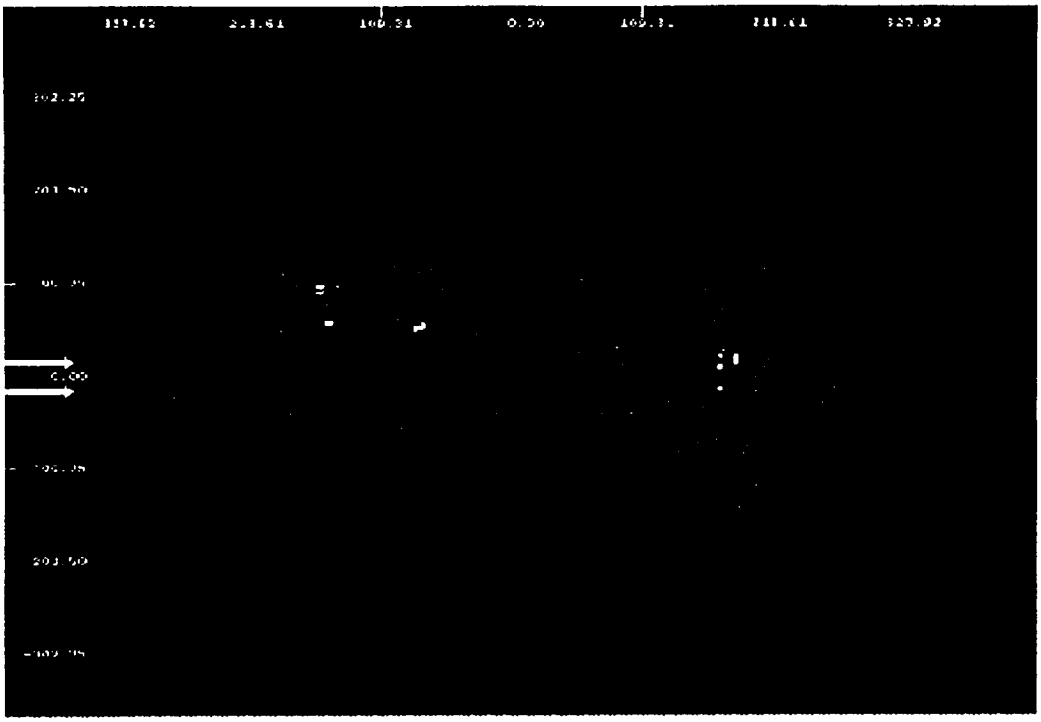


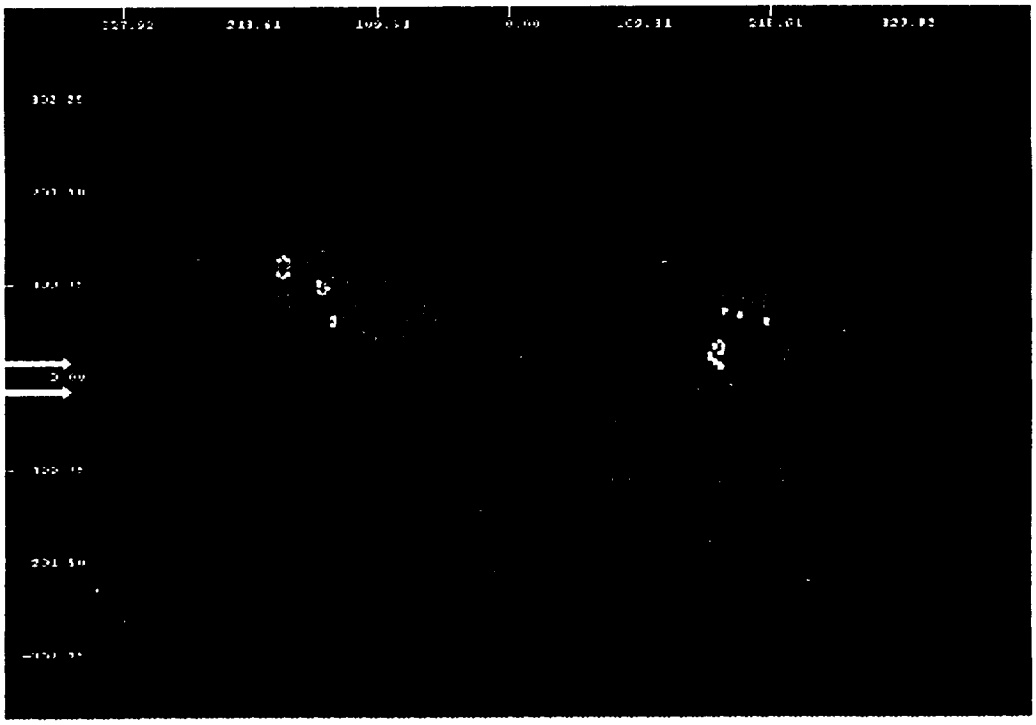
Fig. 7. The ISAR images of a generic aircraft. The images were generated for a frequency scan from 9.5 to 10.5 GHz and a monostatic angular scan from -3° to 3°.

(a) Target orientation.
(b) Conventional Fourier processing of monostatic frequency-aspect data.
(c) Direct image-domain ray-tube integration formula.

(b) Conventional frequency–aspect approach



(c) Direct image–domain ray–tube integration formula



dBsm level

Fig. 7. The ISAR images of a generic aircraft . The images were generated for a frequecncy scan from 9.5 to 10.5 GHz and a monostatic angular scan from -3^{o} to 3^{o}.
(a) Target orientation.
(b) Conventional Fourier processing of monostatic frequency–aspect data.
(c) Direct image–domain ray–tube integration formula.

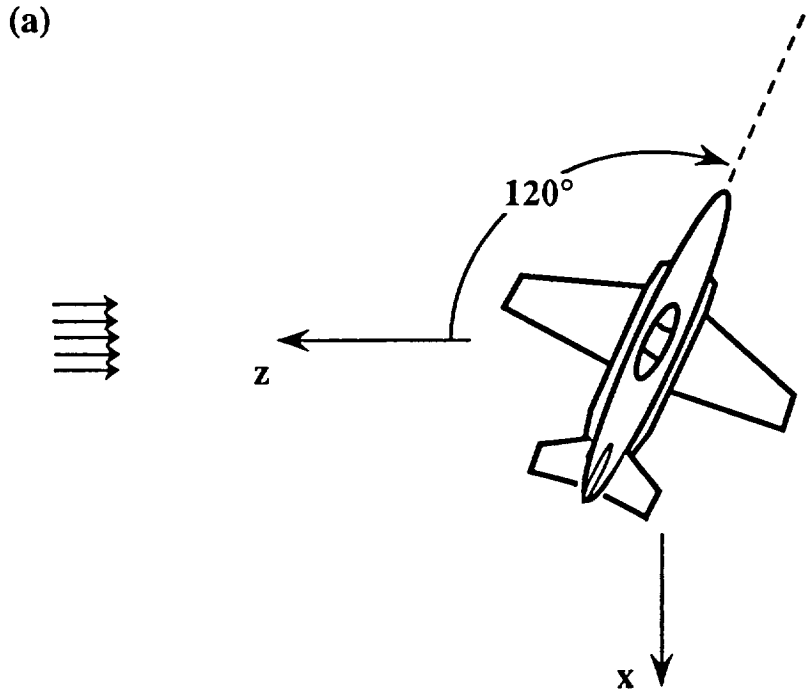
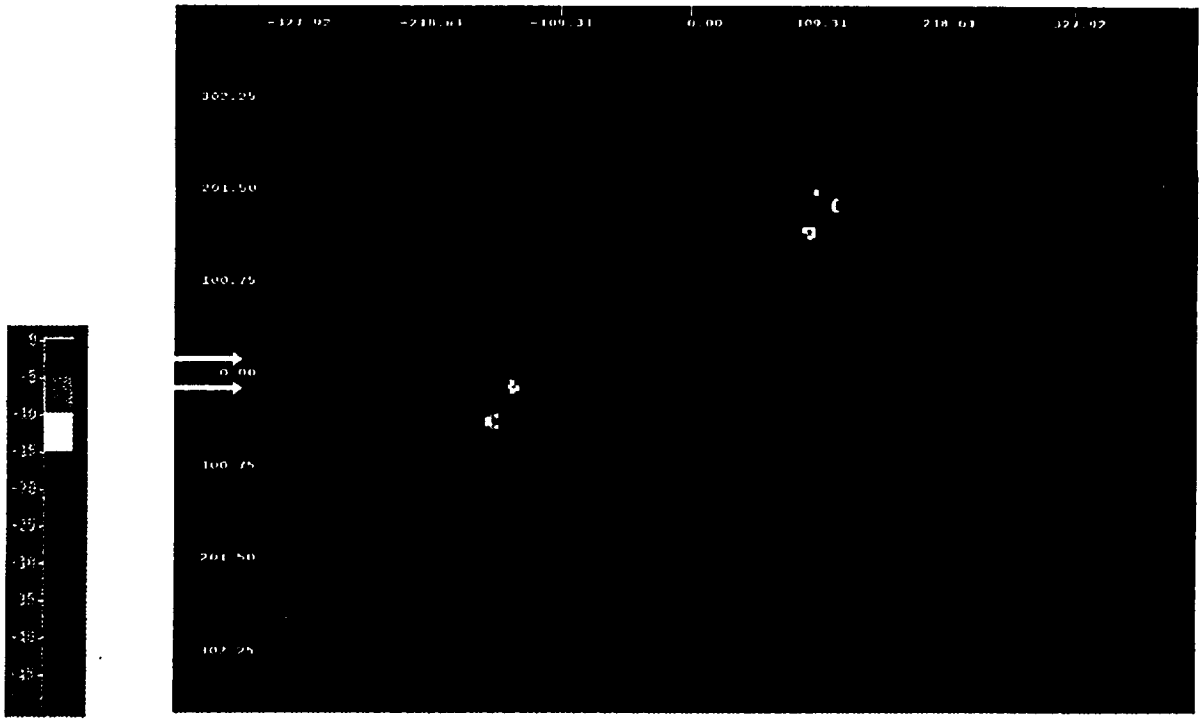


Fig. 8. The ISAR images of a generic aircraft. The images were generated for a frequency scan from 9.5 to 10.5 GHz and a monostatic angular scan from -3° to 3°.

(a) Target orientation.
(b) Conventional Fourier processing of monostatic frequency-aspect data.
(c) Direct image-domain ray-tube integration formula.

(b) Conventional frequency–aspect approach

(c) Direct image–domain ray–tube integration formula



dBsm level

Fig. 8. The ISAR images of a generic aircraft. The images were generated for a frequecncy scan from 9.5 to 10.5 GHz and a monostatic angular scan from -3^{o} to 3^{o}.
(a) Target orientation.
(b) Conventional Fourier processing of monostatic frequency–aspect data.
(c) Direct image–domain ray–tube integration formula.

A FAST ALGORITHM FOR SIGNATURE PREDICTION AND IMAGE FORMATION USING THE SHOOTING AND BOUNCING RAY TECHNIQUE

- Sullivan Scheme for Xpatch3 -

Rajan Bhalla and Hao Ling

Department of Electrical and Computer Engineering
The University of Texas at Austin
Austin, TX 78712-1084

January 1994

Air Force Wright Laboratory
through NASA Grant NCC 3-273

A FAST ALGORITHM FOR SIGNATURE PREDICTION AND IMAGE FORMATION USING THE SHOOTING AND BOUNCING RAY TECHNIQUE

Rajan Bhalla and Hao Ling

Department of Electrical and Computer Engineering
The University of Texas at Austin
Austin, TX 78712-1084

Abstract We present a fast simulation algorithm for generating the range profiles and inverse synthetic aperture radar (ISAR) images of complex targets using the shooting and bouncing ray (SBR) technique. Starting with the time-domain and image-domain ray-tube integration formulas we derived previously, we cast these formulas into a convolution form. The convolution consists of a non-uniformly sampled signal and a closed form time-domain or image-domain ray spread function. Using a fast scheme proposed by Sullivan [1], the non-uniformly sampled function is first interpolated onto a uniform grid before the convolution is performed by the fast Fourier transform (FFT) algorithm. Results for several complex targets are presented to demonstrate the tremendous computation time savings and excellent fidelity of the scheme. Using the fast scheme, a speed gain of a factor of 30 is achieved over the direct convolution in range profile computation and a factor of 180 in ISAR image formation for a typical aircraft at X-band.

This work is support by NASA Grant NCC3-217 and in part by the Joint Services Electronics Program.

1. INTRODUCTION

In radar signature applications it is often desirable to generate the range profiles and inverse synthetic aperture radar (ISAR) images of a target. They can be used either as identification tools to distinguish and classify the target from a collection of possible targets, or as diagnostic/design tools to pinpoint the key scattering centers on the target. The simulation of synthetic range profiles and ISAR images is usually a time intensive task and computation time is of prime importance. In this work, we present a fast algorithm to generate range profiles and ISAR images using the shooting and bouncing ray (SBR) technique.

The SBR method is a standard ray-tracing technique which is widely used for predicting the scattering from complex, realistic targets [2]-[7]. In the SBR technique a dense grid of rays is shot from the incident direction towards the target. Rays are traced according to the laws of geometrical optics as they bounce around the target. At the exit point of each ray, a ray tube integration is performed to find its contribution to the total scattered field from the target. Recently, a closed form time-domain ray-tube integration formula was derived for the computation of the time-domain response (or range profile) of a conducting target [8],[9]. This formula gives the explicit contribution of each exit ray in the time domain. Therefore, by summing the contribution from each ray in the time domain directly, the overall time-domain response of the target can be obtained without resorting to any multi-frequency calculations. More recently, we have also extended this formula to the two-dimensional ISAR plane under the small-angle approximation [10]. The closed form image-domain ray-tube integration formula gives the contribution of each ray to the overall ISAR image directly, without resorting to any multi-frequency, multi-aspect calculations.

The major computation time for the SBR technique can be attributed to two parts: geometrical ray tracing and electromagnetic computation. The latter includes the computation of the geometrical optics field associated with each ray and the frequency-

domain, time-domain, or image-domain ray-tube integration procedure. For a realistic and complex target both the ray tracing and electromagnetic computation parts are time consuming. Nussbaum [11] and Baden [12] proposed fast schemes to improve the computation speed of the frequency-domain implementation of the SBR technique. In this paper we apply a fast scheme, proposed originally by Sullivan [1], which cuts the electromagnetic computation time for the time-domain and image-domain implementations of the SBR technique to practically zero.

The application of the fast scheme is based on the observation that the time-domain ray-tube summation for computing the range profile is expressible as a convolution between a weighted impulse train and a closed form function h(t):

$$E(t) = [\sum_{\text{i rays}} \alpha_i \, \delta(t - t_i)] * [h(t)] \qquad (1)$$

In the above expression, E(t) is the time-domain scattered field (the magnitude of which is the range profile), α_i and t_i are the magnitude and delay time associated with the i'th ray and h(t) is the closed form time-domain ray-tube integration formula derived in [8], [9]. We shall term h(t) the "time-domain ray spread function." Rather than performing the direct convolution it is much more efficient to take advantage of the fast Fourier transform (FFT). The major problem is that the weighted impulse train in our problem is not a uniformly spaced function and hence its FFT cannot be readily evaluated. The scheme proposed by Sullivan overcomes this problem by interpolating the non-uniformly spaced impulse train onto a uniform grid before performing the FFT algorithm. As will be demonstrated in the numerical examples, a tremendous improvement in computation time is achieved using this scheme. The same idea can also be applied to two dimensions for the image-domain implementation of the SBR technique. The Sullivan scheme for the time domain will be presented in Sec. 2 followed by several numerical examples and timing results. In Sec. 3 the image-domain implementation of the scheme will be described.

2. RANGE PROFILE COMPUTATION

The closed form expression for the time-domain scattered field (or range profile) from a perfectly conducting target at an observation angle (θ,ϕ) based on the SBR methodology was derived in [8],[9]. It is a ray sum over the M exit rays and takes on the form

$$E(t) = e^{j\omega_o t} \sum_{\substack{i=1 \\ \text{all rays}}}^{M} C_i(\theta,\phi)\,(\frac{\Delta\omega}{2\pi c})\, e^{-j\frac{d_i'}{c}\omega_o} \cdot \cdot \left\{ \frac{\Delta\omega}{4\pi}\text{sinc}'[\frac{\Delta\omega}{2}(t-\frac{d_i'}{c})] + j\frac{\omega_o}{2\pi}\text{sinc}[\frac{\Delta\omega}{2}(t-\frac{d_i'}{c})] \right\} \qquad (2)$$

where

$$C_i(\theta,\phi) = f_i\ (\Delta A)_{exit}$$

In the above expression f_i is related to the geometrical optics fields associated with the exit ray tube, $(\Delta A)_{exit}$ is the exit ray-tube cross section, c is the speed of light in vacuum, and d_i' is the total distance traveled by the i'th ray measured with respect to phase reference at the origin. The frequency bandwidth of the calculation is assumed to be $\Delta\omega$ with a center frequency of ω_o.

For ease of notation let us introduce the time-domain ray spread function h(t) as:

$$h(t) = e^{j\omega_o t}\left\{ \text{sinc}(\frac{\Delta\omega}{2}t) - j\frac{\Delta\omega}{2\omega_o}\text{sinc}'(\frac{\Delta\omega}{2}t) \right\} \qquad (3)$$

The range profile given in eqn. (2) can then be written in terms of h(t) as:

$$E(t) = \sum_{i\ \text{rays}} \alpha_i\, h(t-\frac{d_i'}{c}) \qquad (4)$$

where

$$\alpha_i = C_i(\theta,\phi)\frac{j\omega_o\Delta\omega}{4\pi^2 c}$$

The SBR implementation of the range profile can now be carried out in two parts. In the first part we shoot rays at the target and determine the parameters α_i and d_i' for each ray.

In the second part the time-domain response (or range profile) is updated at N uniformly sampled time (or range) bins spaced Δt (or $\Delta r = c\Delta t/2$) apart as follows:

$$E(n\Delta t) = \sum_{i \text{ rays}} \alpha_i \, h(n\Delta t - \frac{d_i'}{c}) \tag{5}$$

Since for each ray the range profile has to be updated at N range bins, the total computation cost of the second part is proportional to NM where M is the total number of rays launched. For complex targets M can range into the millions and this is clearly a time-intensive operation. In [9] the decaying nature of h(t) was exploited by truncating it after several sidelobes. In this manner it is not necessary to update all N range bins. However, this gain in computation time comes at the price of loss of accuracy.

To illustrate the new scheme let us rewrite eqn. (4) as a convolution between a weighted impulse train and h(t):

$$E(t) = \Big(\sum_{i \text{ rays}} \alpha_i \, \delta(t - \frac{d_i'}{c}) \Big) * h(t) \tag{6}$$

To evaluate the convolution we can take advantage of the FFT algorithm by finding the Fourier transform of the two functions and taking the inverse Fourier transform of their product. The problem is that the weighted impulses do not occur on a uniformly sampled grid and the FFT of the weighted impulse train cannot be readily evaluated. The scheme proposed by Sullivan overcomes this problem by transforming the non-uniformly sampled impulse train into a uniformly sampled one using an interpolation algorithm. To illustrate the interpolation algorithm let us define the impulse train as

$$x(t) = \sum_{i \text{ rays}} \alpha_i \, \delta(t - t_i) \tag{7}$$

where $t_i = d_i'/c$ is the time of arrival of the i'th ray in the time domain. We want to transform x(t) into a uniformly sampled impulse train $x_s(t)$ given by

$$x_s(t) = \sum_{\text{i rays}} \alpha_i \{ \sum_{k} \beta_k \delta(t - k\Delta t) \} \tag{8}$$

in which the original impulse at t_i is approximated by a series of K impulses on the uniform grid about t_i. Different orders of interpolation, corresponding to different number of terms K, were derived by Sullivan to calculate the expansion. The coefficients β_k are chosen so as to decrease the mean square error between $x(t)$ and $x_s(t)$. A detailed derivation of β_k is given in [1] and we will only highlight the results for the zeroth and first order interpolation. In the zeroth order interpolation (K=1), the original impulse occurring at $k\Delta t \le t_i < (k+1)\Delta t$ is replaced by one occurring at $k\Delta t$:

$$\delta(t - t_i) \cong \delta(t - k\Delta t) \tag{9}$$

In other words, we simply shift the original impulse left to a sampling location. (A slightly better implementation would be to shift the impulse either left or right to the closest sampling location.) In the first order interpolation (K=2), the original impulse occurring at $k\Delta t \le t_i < (k+1)\Delta t$ is replaced by two weighted impulses occurring at $k\Delta t$ and $(k+1)\Delta t$ as:

$$\delta(t - t_i) \cong \frac{(t_i - k\Delta t)}{\Delta t} \delta(t - k\Delta t) + \frac{((k+1)\Delta t - t_i)}{\Delta t} \delta(t - (k+1)\Delta t) \tag{10}$$

Note that the new impulses on the sampling grid are linearly weighted by their distances from t_i. Sullivan generalized this scheme to account successively for higher order interpolations. Of course, the transformation of the non-uniformly sampled signal into a uniformly sampled one will always result in some error. One way to decrease the error is to use a higher order interpolation scheme. But this will result in an increased computational burden. In fact we can easily see that the computational cost of the Sullivan scheme is given by KM since for each ray only K points need to be updated to form $x_s(t)$. Once $x_s(t)$ is found the remaining FFT computations involve only a series of NlogN operations which, for N<<M, can be considered negligible. An alternative way to improve accuracy is to decrease the sampling interval Δt, which is equivalent to increasing

the number of samples N for a given range extent. Memory permitting, this is the more appealing choice since it does not add any significant penalty to the computation time.

The implementation of the Sullivan scheme into the SBR range profile computation is as follows. For a given target we choose a range window R which is roughly twice the size of the target. We then divide the range window into N equally spaced bins with binwidth $\Delta r = c\Delta t/2 = R/N$. The number of bins, N, should be chosen such that the sample spacing Δt corresponds to at least five times the Nyquist frequency of h(t), i.e., $\Delta t < 2\pi/(5\Delta\omega)$. In the implementation, once the i'th ray is traced we interpolate it onto the uniform grid. After all M rays have been traced, the function $x_s(t)$ is constructed. Finally, the range profile is computed as,

$$E(n\Delta t) = FFT^{-1}\left[\, FFT[h(n\Delta t)] \cdot FFT[x_s(n\Delta t)]\right] \qquad (11)$$

Two numerical examples are presented next to demonstrate the accuracy and computation time of the Sullivan scheme for time-domain SBR. The SBR-based code Xpatch [6],[7] is used in the simulation. For the following examples we use linear interpolation to construct $x_s(t)$ and sample at six times the Nyquist frequency of h(t). The first example is the target "Turkey." The CAD display of this target is shown in Fig. 1. The range profile of "Turkey" has been generated using a range window of 95 inches, a center frequency of 10 GHz and a bandwidth of 4 GHz. Fig. 2 compares the range profile of "Turkey" generated using the Sullivan scheme and the reference range profile from performing the brute-force direct convolution of eqn. (5). The reference range profile is sampled at the Nyquist frequency. The two results show excellent agreement. Figs. 3 and 4 illustrate the results for a generic aircraft. The CAD display of the target is shown in Fig. 3 and the range profiles shown in Fig. 4. A range window of 1150 inches, a center frequency of 10 GHz and a bandwidth of 2 GHz are used in generating the range profiles. Again, very good agreement is observed for this much more complex target. The total time to generate the range profile using the Sullivan scheme takes 1.3 hrs. as compared to 2.28

hrs. by direct convolution on a Silicon Graphics Indigo (R3000) workstation. Table 1 shows a breakdown of the computation time for the range profile using the two schemes. The time to perform the convolution using the Sullivan scheme takes 2 min. as compared to 1 hr. by direct convolution, or a speed gain of a factor of 30. It is evident that the total computation time of 1.3 hrs. is just the ray-trace time. The time spent on the Sullivan scheme is essentially zero.

3. ISAR IMAGE GENERATION

The image-domain implementation of the Sullivan scheme is a two-dimensional extension of the time-domain scheme. The present discussion will closely parallel that of Sec. 2. Using a small-angle, bistatic approximation, the closed form expression for the ISAR image of a target as a function of range (r) and cross range (xr) was derived in [10]. It is a ray sum over M exit rays and takes on the form

$$O(r,xr) = -\frac{2k_o\theta_o\Delta k}{\pi^2} \sum_{\substack{i=1 \\ \text{all rays}}}^{M} g_i \, (\Delta A)_{exit} \, e^{j2k_o(r-r_i)} \cdot$$

$$\cdot \; \text{sinc}[\Delta k(r - r_i)] \;\; \text{sinc}[k_o\theta_o(xr - xr_i)] \qquad (12)$$

In the above expression, $r_i=d_i'/2$ is the total range delay of the i'th ray where d_i' is defined earlier in eqn. (2). xr_i is the cross range location of the i'th ray. It can be shown that under the exact Doppler interpretation xr_i is simply the average of the cross-range coordinates of the first and the last hit point of the i'th ray on the target [13]. g_i is related to the geometrical optics fields associated with the i'th ray tube and $(\Delta A)_{exit}$ is the ray-tube cross section. The k-space bandwidth of the calculation is assumed to be $\Delta k=\Delta\omega/c$ with center frequency $k_o=\omega_o/c$ and a monostatic angular look window of θ_o.

To cast eqn. (12) into a convolution form, we first introduce the image-domain ray spread function h(r,xr) as:

$$h(r,xr) = e^{j2k_o r} \left\{ \text{sinc}(\Delta k \, r)\cdot\text{sinc}(k_o\theta_o xr) \right\} \qquad (13)$$

The ISAR image given in eqn. (12) can then be written in terms of h(r,xr):

$$O(r,xr) = \sum_{i \text{ rays}} \gamma_i \, h(r - r_i, xr - xr_i) \qquad (14)$$

where

$$\gamma_i = -\frac{2k_o\theta_o\Delta k}{\pi^2} \, g_i \, (\Delta A)_{exit}$$

In the SBR implementation of the ISAR simulation process we first shoot rays at the target to determine the parameters r_i, xr_i and γ_i for each ray. The image plane, which consists of a total range window of R and a cross-range window XR, is divided into N_r equally spaced range bins with binwidth $\Delta r=R/N_r$ and N_{xr} equally spaced cross-range bins with binwidth $\Delta xr=XR/N_{xr}$. The ISAR image is then formed by updating the N_r x N_{xr} uniformly sampled image grid as follows:

$$O(n_r\Delta r, n_{xr}\Delta xr) = \sum_{\text{i rays}} \gamma_i \, h(n_r\Delta r - r_i, n_{xr}\Delta xr - xr_i) \tag{15}$$

For each ray the ISAR image has to be updated at N_r x N_{xr} sampling points and the total image formation time is proportional to $N_rN_{xr}M$ where M is the number of rays launched. In [10] the decaying nature of h(r,xr) was exploited by truncating it after several sidelobes in order to speed up computation. However the Sullivan scheme is by far superior.

To apply the Sullivan scheme we rewrite eqn. (14) as a two-dimensional convolution between a weighted impulse train and h(r,xr):

$$O(r,xr) = \Big(\sum_{\text{i rays}} \gamma_i \, \delta(r - r_i, xr - xr_i) \Big) * h(r, xr) \tag{16}$$

Before performing the above convolution using the FFT, we transform the two-dimensional non-uniformly sampled impulse train

$$x(r, xr) = \sum_{\text{i rays}} \gamma_i \, \delta(r - r_i, xr - xr_i) \tag{17}$$

into a uniformly sampled one

$$x_s(r, xr) = \sum_{\text{i rays}} \gamma_i \, \{ \sum_{k,j} \beta_{k,j} \delta(r - k\Delta r, xr - j\Delta xr) \} \tag{18}$$

by approximating the original impulse at (r_i, xr_i) with a series of KxJ impulses on the uniform grid about (r_i, xr_i). For the zeroth order interpolation (K=J=1), the original

impulse occurring at $k\Delta r \le r_i < (k+1)\Delta r$, $j\Delta xr \le xr_i < (j+1)\Delta xr$ is simply replaced by one occurring at $k\Delta r$, $j\Delta xr$:

$$\delta(r - r_i, xr - xr_i) \cong \delta(r - k\Delta r, xr - j\Delta xr) \tag{19}$$

For the first order interpolation (K=J=2), the original impulse is replaced by four weighted impulses as:

$$\begin{aligned}\delta(r - r_i, xr - xr_i) \cong \frac{r_1}{r}\delta(r - k\Delta r, xr - j\Delta xr) + \frac{r_2}{r}\delta(r - k\Delta r, xr - (j+1)\Delta xr) \\ + \frac{r_3}{r}\delta(r - (k+1)\Delta r, xr - j\Delta xr) + \frac{r_4}{r}\delta(r - (k+1)\Delta r, xr - (j+1)\Delta xr)\end{aligned} \tag{20}$$

The distances r_1, r_2, r_3, r_4 are shown in Fig. 5 and $r = (r_1 + r_2 + r_3 + r_4)$. In other words, the new impulses have magnitudes which are linearly weighted by their distances from (r_i, xr_i). Again the scheme can be generalized to account successively for higher order interpolations at the price of increased computational burden. The computational cost of the Sullivan's scheme in two dimensions is given by KJM since for each ray KJ points need to be updated to form $x_s(r,xr)$. Once $x_s(r,xr)$ is found the remaining FFT computations can be considered insignificant for N_r, $N_{xr} << M$. In the simulation, the number of bins N_r,N_{xr} are chosen to be at three to five times the Nyquist sampling frequency of h(r,xr). Once the i'th ray is traced we interpolate it onto the uniform grid. After all M rays have been traced, the function $x_s(r,xr)$ is constructed. Finally the ISAR image $O(n_r\Delta r, n_{xr}\Delta xr)$ is computed as

$$O(n_r\Delta r, n_{xr}\Delta xr) = FFT2^{-1}\left[\, FFT2(h(n_r\Delta r, n_{xr}\Delta xr))\cdot FFT2(x_s(n_r\Delta r, n_{xr}\Delta xr))\right] \tag{21}$$

where FFT2 and $FFT2^{-1}$ correspond to respectively to the two-dimensional forward and inverse FFT.

Two numerical examples are presented next to demonstrate the accuracy and computation time of the Sullivan scheme for image-domain SBR. In the following examples we use zeroth order interpolation to construct $x_s(r,xr)$ and sample at four times the Nyquist frequency. The ISAR images are of the generic aircraft shown in Fig. 3 at two

different look angles using a center frequency of 10 GHz and a bandwidth of 2 GHz. The first set of ISAR images are generated at an aspect of 30° as shown in Fig. 6(a). Figs. 6(b) and 6(c) compare the ISAR image of the generic aircraft generated using the Sullivan scheme and the reference ISAR image obtained by direct convolution of eqn. (15). The ISAR image generated by direct convolution is sampled at the Nyquist frequency. The two images show excellent agreement. No noticeable visual degradation in the image fidelity from the Sullivan scheme can be detected within a dynamic range of 40 dB. Fig. 7 illustrates the results for the same aircraft at an aspect of 120°. Again, excellent agreement is observed. Table 2 compares the breakdown of the computation time for ISAR image formation using direct convolution and the Sullivan scheme. The time to perform the convolution using the Sullivan scheme takes 2 min. as compared to 6.33 hrs. by direct convolution, a speed gain of a factor of 180. It is evident that the total image simulation time of 1.3 hrs. is just the ray-trace time. The time spent on the Sullivan scheme is essentially zero.

4. SUMMARY

In this paper we presented a fast simulation algorithm to generate range profiles and ISAR images of complex targets using the shooting and bouncing ray technique. The scheme is based on the recognition that the form of the time-domain and image-domain ray-tube integration formula we derived previously can be written as a convolution between a non-uniformly sampled impulse train and a time-domain or image-domain ray spread function. The non-uniformly sampled impulse train can be interpolated onto a uniformly sampled grid using the Sullivan scheme before the convolution is performed by the fast Fourier transform algorithm. Results for several complex targets are presented to demonstrate the tremendous computation time savings and excellent fidelity of the scheme. Using the fast algorithm, a speed gain of a factor of 30 is achieved over the direct convolution in range profile computation and a factor of 180 in ISAR image formation for a typical aircraft at X-band.

ACKNOWLEDGMENTS

The authors are grateful to Prof. S. W. Lee for providing the various versions of Xpatch, to Major D. J. Andersh for bringing Sullivan scheme to our attention and for his continued support of this work, and to T. D. Sullivan and M. Baden for their helpful comments on sampling criterion.

REFERENCES

[1] T. D. Sullivan, "A technique of convolving unequally spaced samples using fast Fourier transforms," Sandia National Laboratories, SAND89-0077, Jan. 1990.

[2] H. Ling, R. Chou and S. W. Lee, "Shooting and bouncing rays: calculating the RCS of an arbitrarily shaped cavity," *IEEE Trans. Antennas Propagat.*, vol. AP-37, pp. 194-205, Feb. 1989.

[3] S. W. Lee, H. Ling and R. Chou, "Ray tube integration in shooting and bouncing ray method," *Microwave Opt. Tech. Lett.*, vol. 1, pp. 286-289, Oct. 1988.

[4] J. Baldauf, S. W. Lee, L. Lin, S. K. Jeng, S. M. Scarborough and C. L. Yu, "High frequency scattering from trihedral corner reflectors and other benchmark targets: SBR versus experiment," *IEEE Trans. Antennas Propagat.*, vol. AP-39, pp. 1345-1351, Sept. 1991.

[5] H. Ling, H. Kim, G. A. Hallock, B. W. Birkner and A. Zaman, "Effect of arcjet plume on satellite reflector performance," *IEEE Trans. Antennas Propagat.*, vol. AP-39, pp. 1412-1420, Sept. 1991.

[6] S. W. Lee, "Test cases for Xpatch," Electromagnetics Lab. Tech. Rept. ARTI-92-4, Univ. of Illinois, Feb. 1992.

[7] S. W. Lee and D. J. Andersh, "Xpatch: a high frequency RCS code," 9th Annual Review of Progress in Applied Computational Electromagnetics, Monterey, CA, March 1993.

[8] S. K. Jeng, R. Bhalla, S. W. Lee, H. Ling and D. J. Andersh, "A time-domain SBR technique for range-profile computation," submitted to *IEEE Trans. Antennas Propagat.*, Sept. 1993.

[9] H. Ling and R. Bhalla, "Time-domain ray-tube integration formula for the shooting and bouncing ray technique," Tech. Rept., Univ. of Texas, April 1993.

[10] R. Bhalla and H. Ling, "Image-domain ray-tube integration formula for the shooting and bouncing ray technique," submitted to *IEEE Trans. Antennas Propagat.*, Aug. 1993.

[11] S. W. Lee and D. J. Andersh, "On Nussbaum method for an exponential series," Electromagnetics Lab. Tech. Rept. ARTI-92-11, Univ. of Illinois, Nov. 1992.

[12] M. Baden, "Computational improvements for synthetic signatures," Tech. Rept., Georgia Tech. Res. Institute, March 1993.

[13] H. Ling and R. Bhalla, "Doppler and cross-range relationship - instantaneous ISAR in the shooting and bouncing ray context," preprint, Oct. 1993.

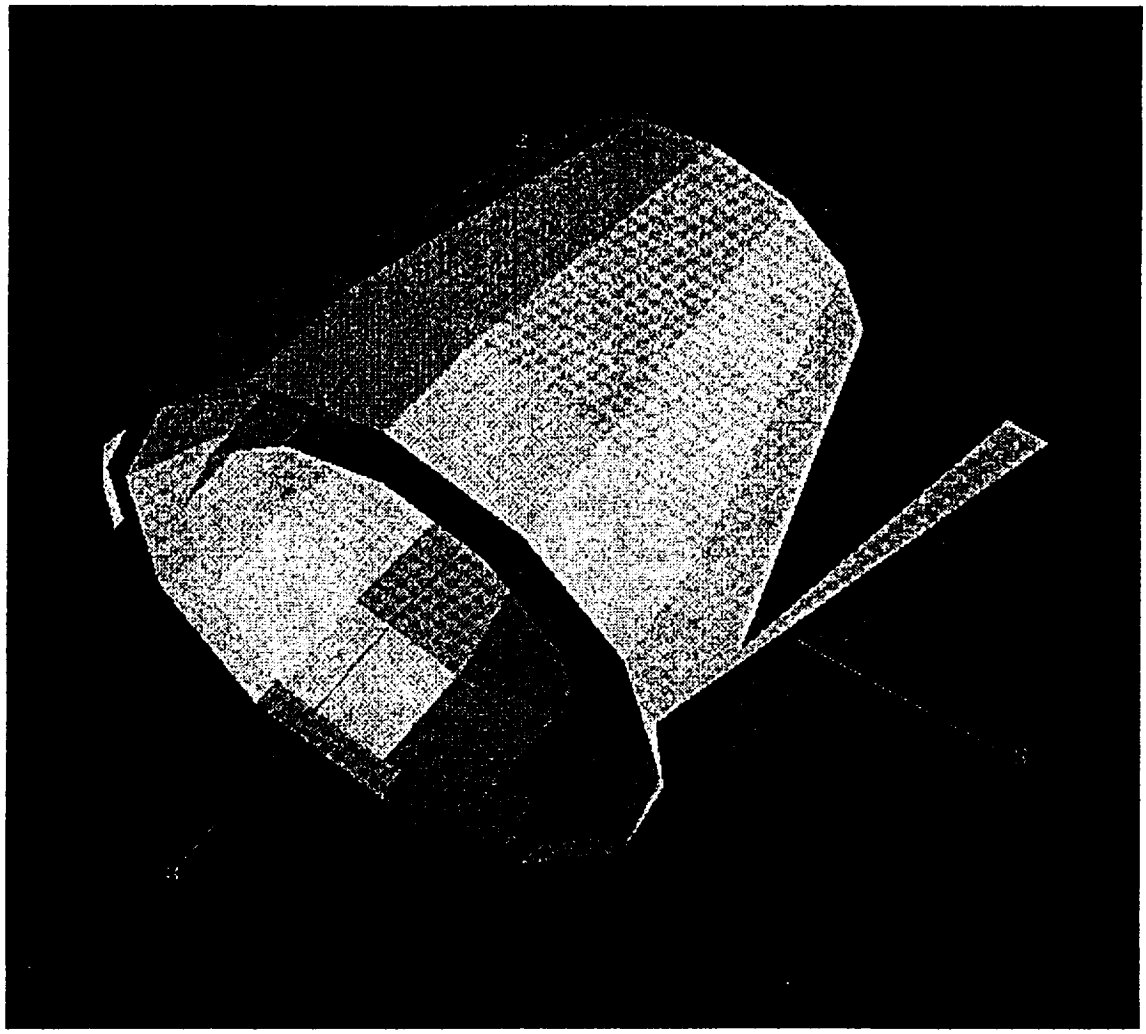


Fig. 1. CAD display of "Turkey".

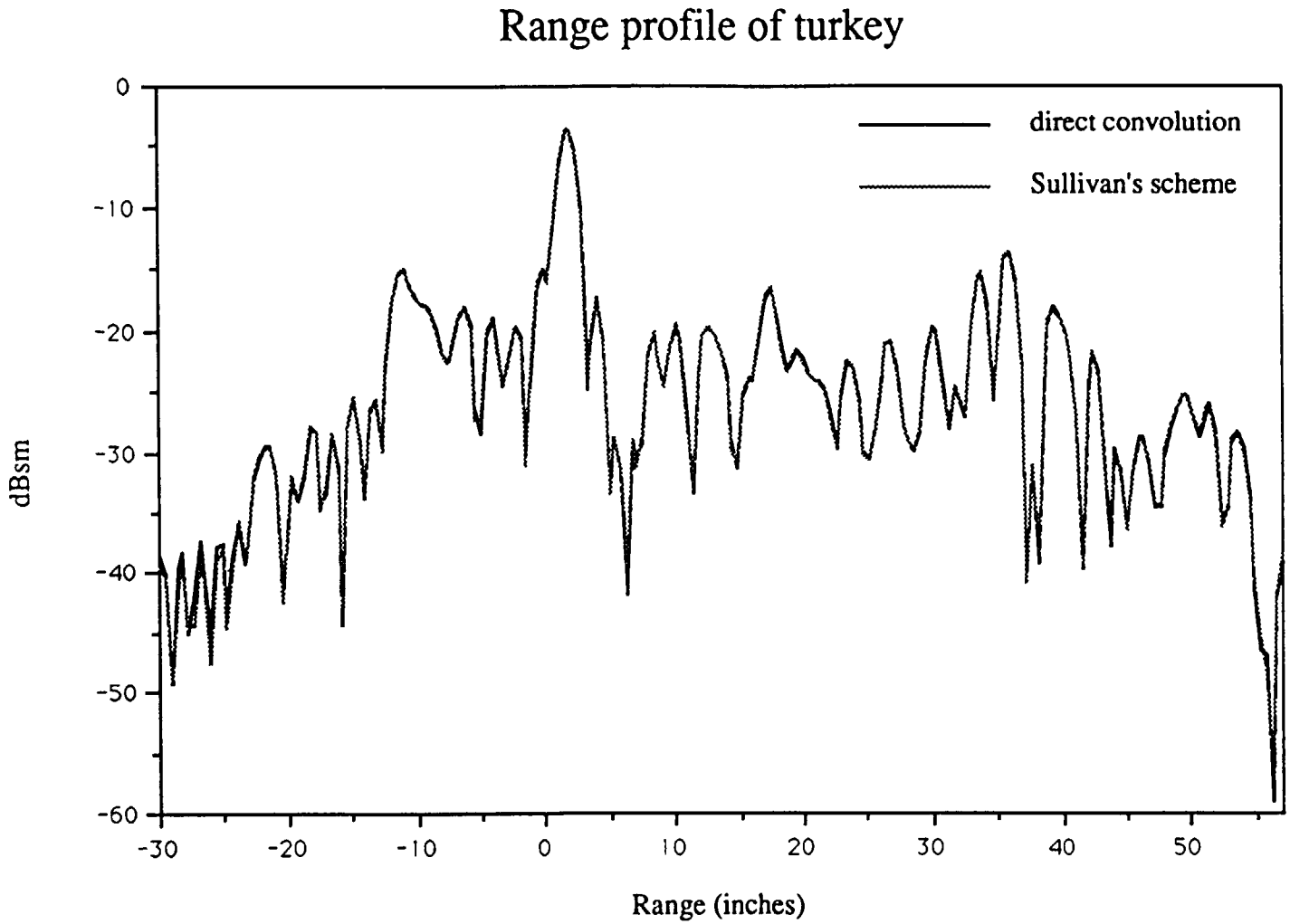


Fig. 2. Comparison of the range profile of "Turkey" computed using direct convolution and Sullivan's scheme. The range profile is computed at the center frequency of 10 GHz and a bandwidth of 4 GHz.

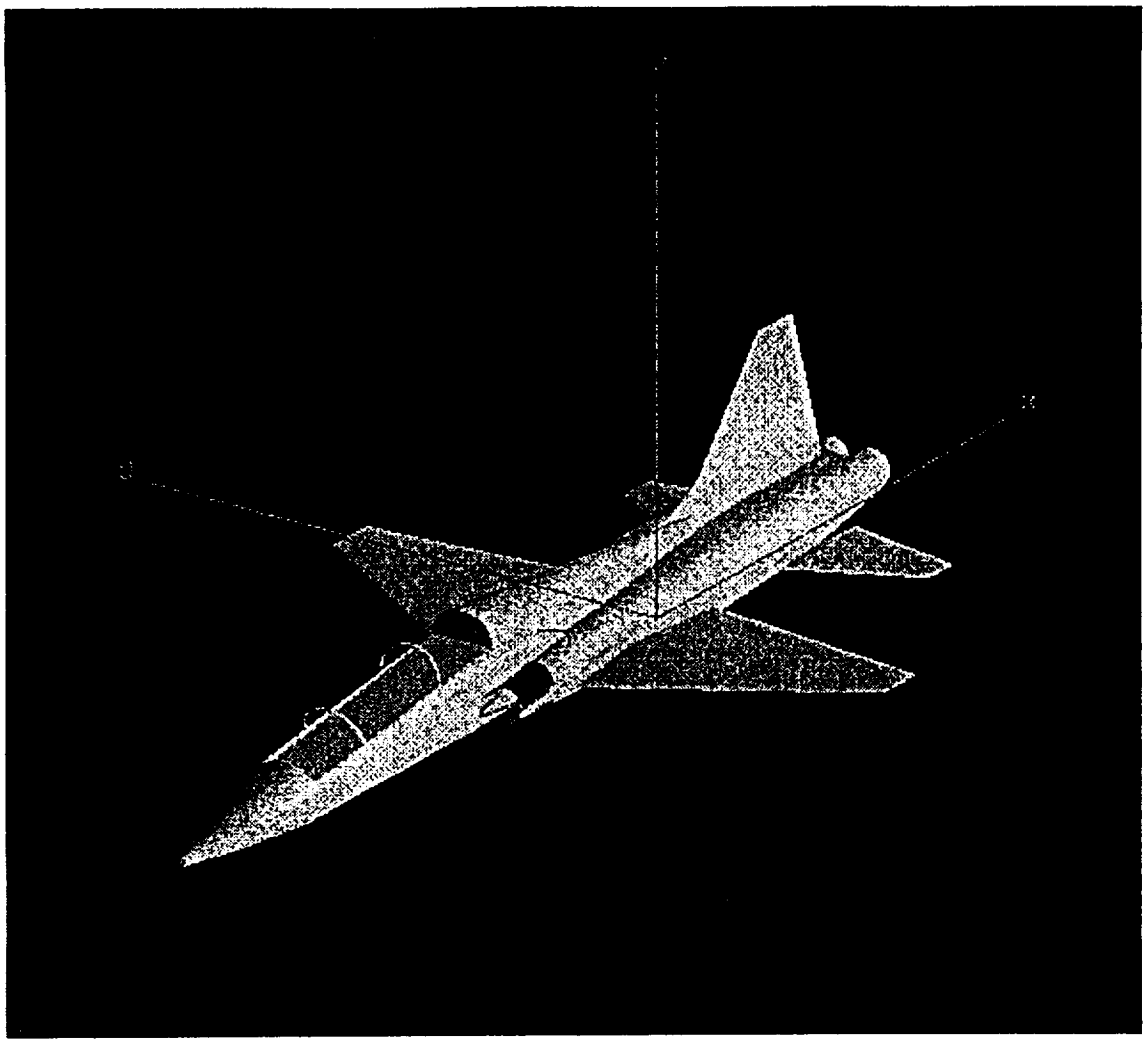


Fig. 3. CAD display of a generic airplane.

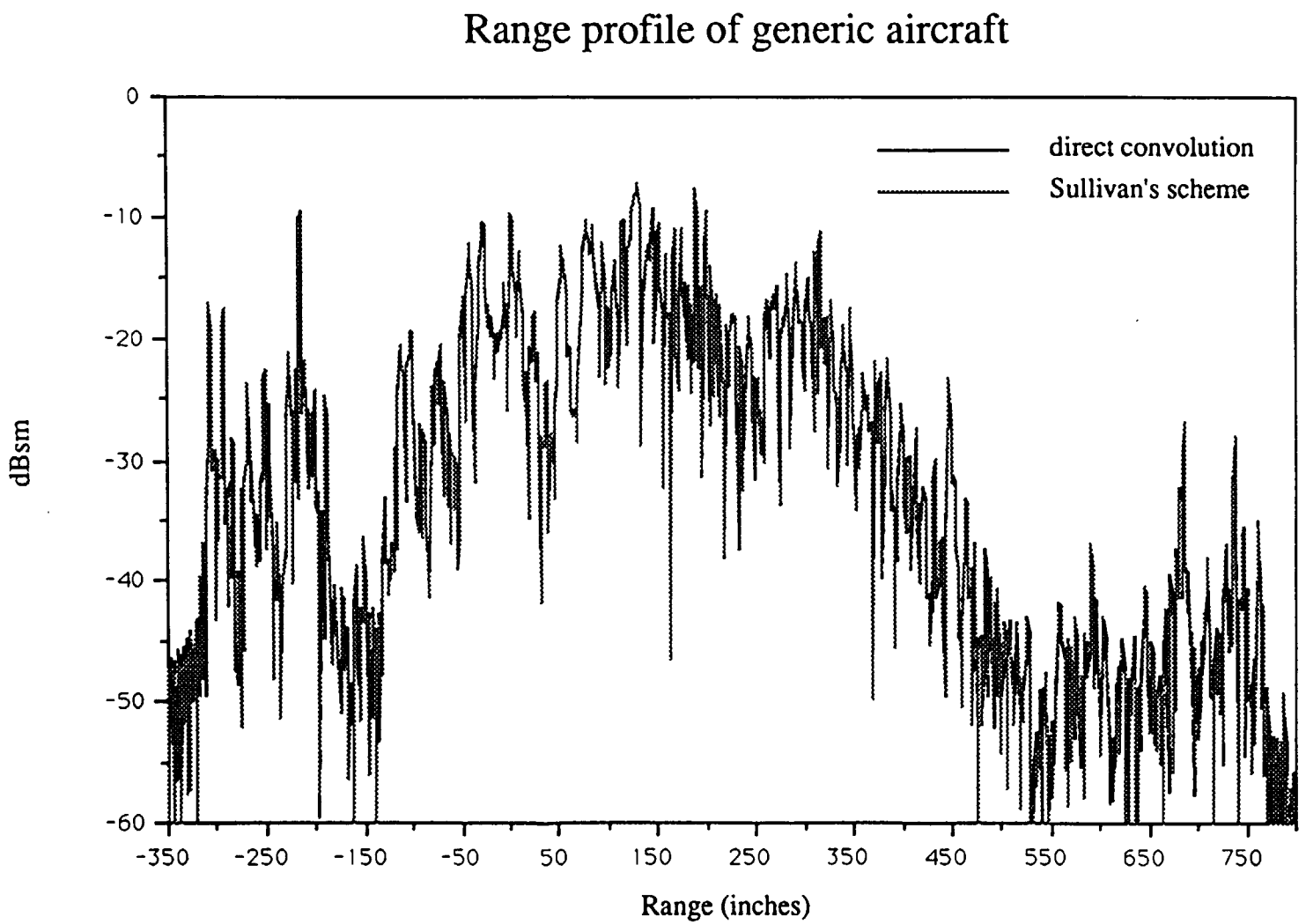


Fig. 4. Comparison of the range profile of a generic aircraft computed using direct convolution and Sullivan's scheme. The range profile is computed at a center frequency of 10 GHz and a bandwidth of 2 GHz.

TABLE 1

	Total Time	Ray - trace time	Electromagnetic computation time
Direct Convolution	2.28 hrs.	1.27 hrs.	1.01 hrs.
Sullivan's Scheme	1.30 hrs.	1.27 hrs.	2 mins.

Table 1. The range profile computation time breakdown for the generic aircraft using direct convolution and Sullivan's scheme. The timings are for the Silicon Graphics Indigo (R3000) workstation.

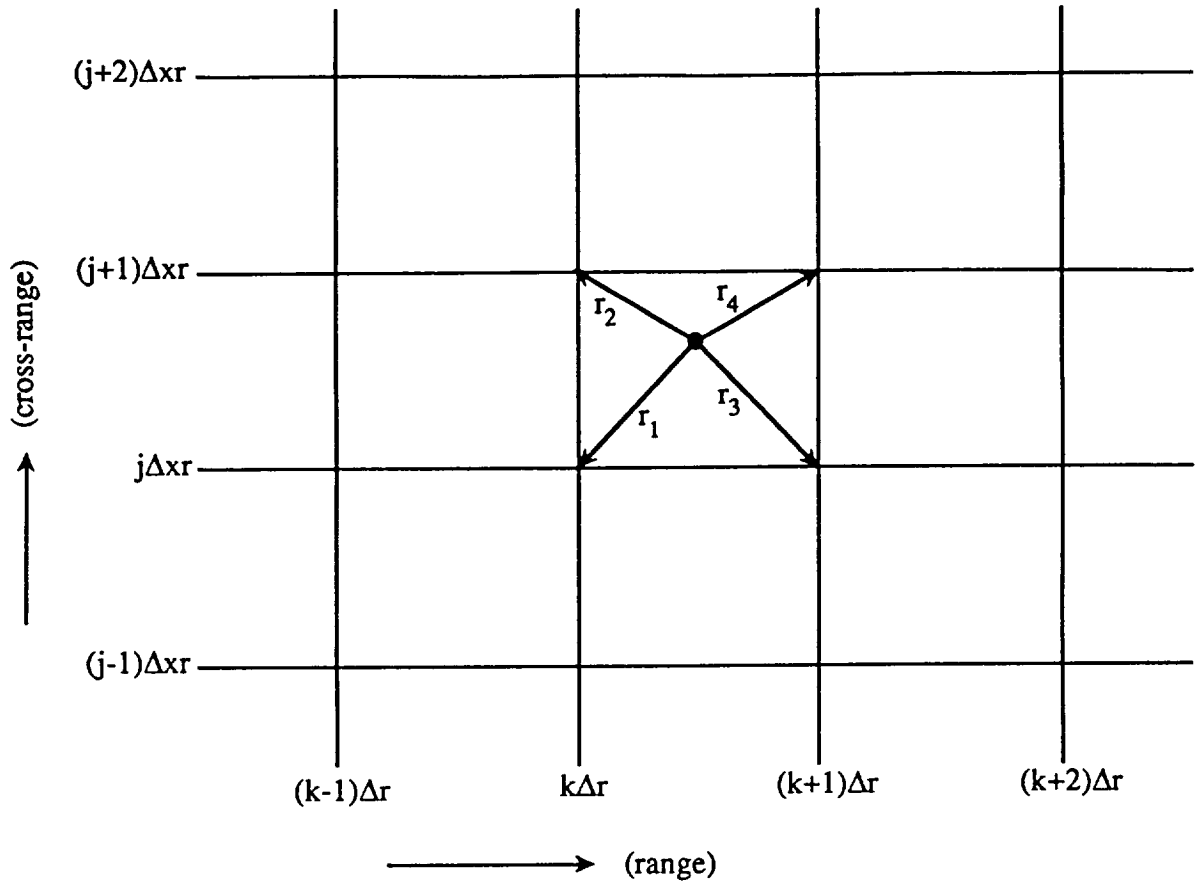


Fig. 5. The first order interpolation for the Sullivan scheme in two-dimensions. The four closest neighbors to a impulse occuring at (r_i, xr_i) are updated.

(a)

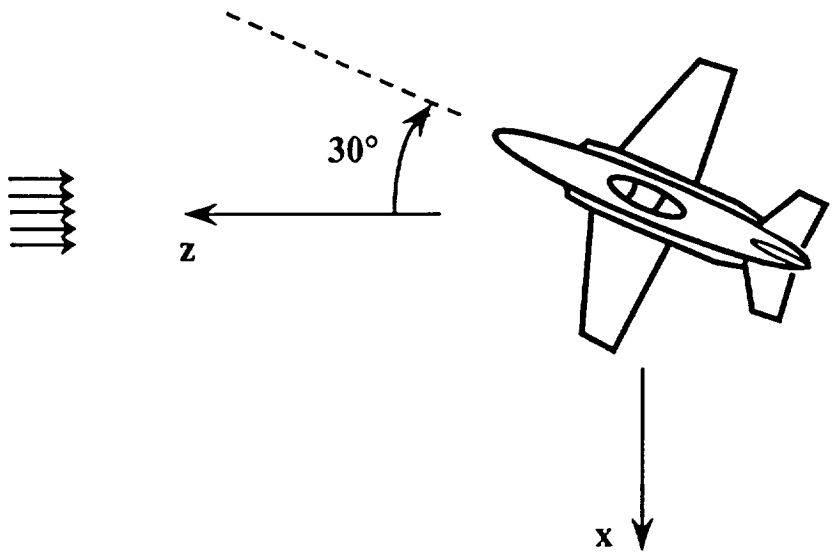
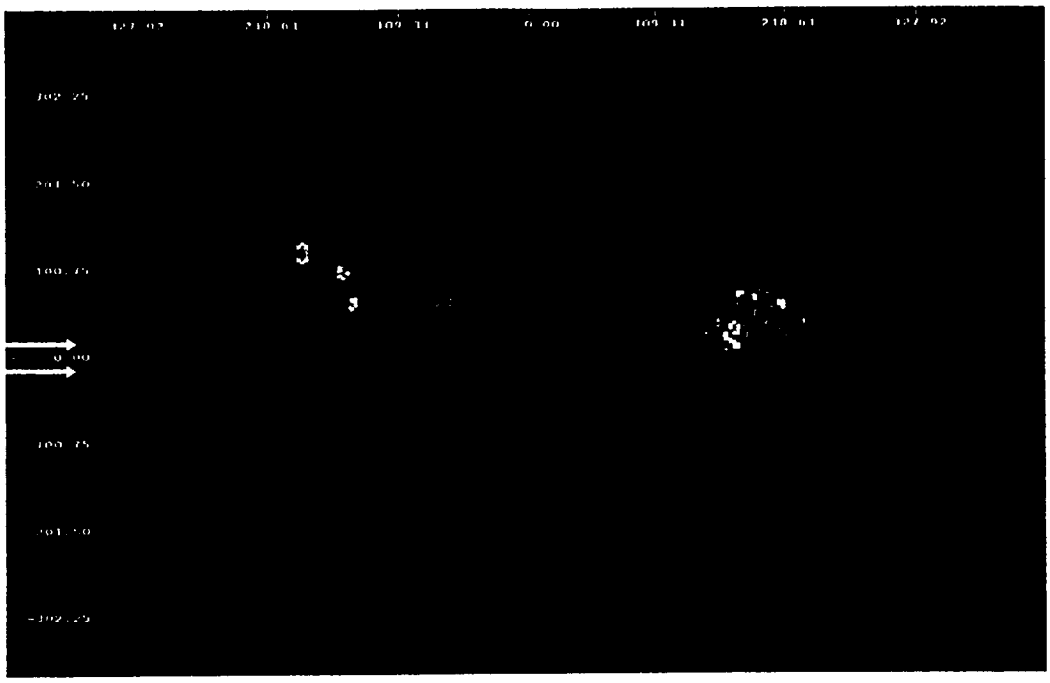


Fig. 6. The ISAR images of a generic aircraft. The images were generated for a frequency scan from 9.5 to 10.5 GHz and a equivalent monostatic angular window of 6°.

(a) Target orientation.
(b) Direct convolution.
(c) Sullivan's scheme.

(b) Direct convolution



(c) Sullivan's scheme

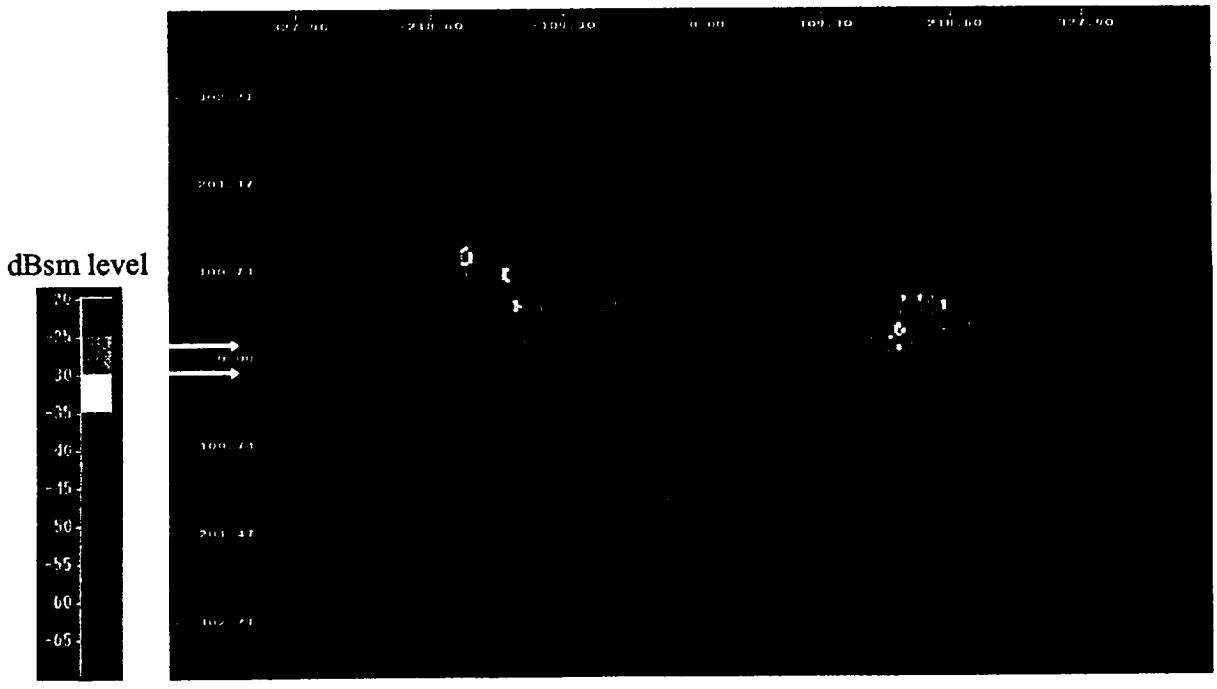


Fig. 6.

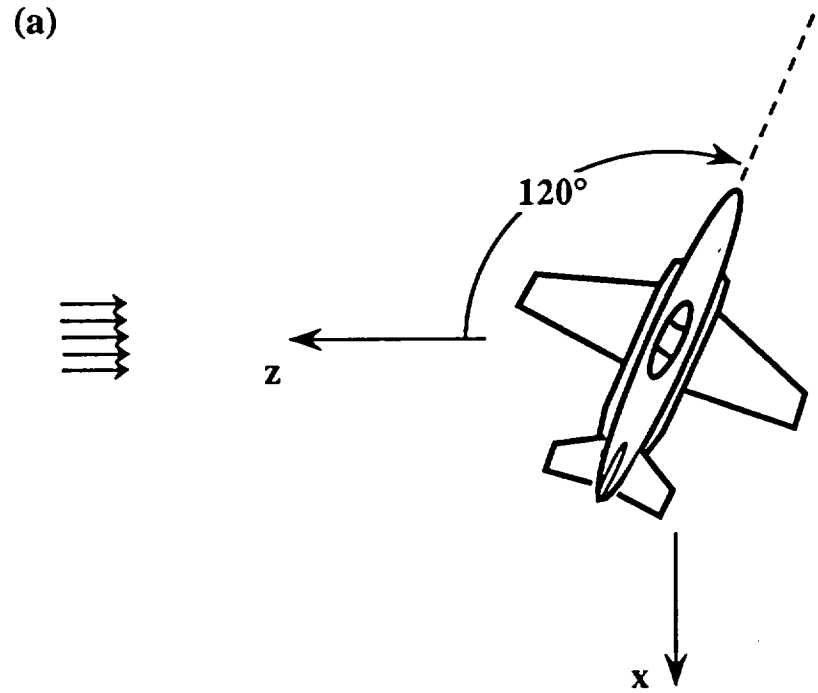


Fig. 7. The ISAR images of a generic aircraft. The images were generated for a frequency scan from 9.5 to 10.5 GHz and a equivalent monostatic angular window of 6°.

(a) Target orientation.
(b) Direct convolution.
(c) Sullivan's scheme.

(b) Direct convolution

(c) Sullivan's scheme

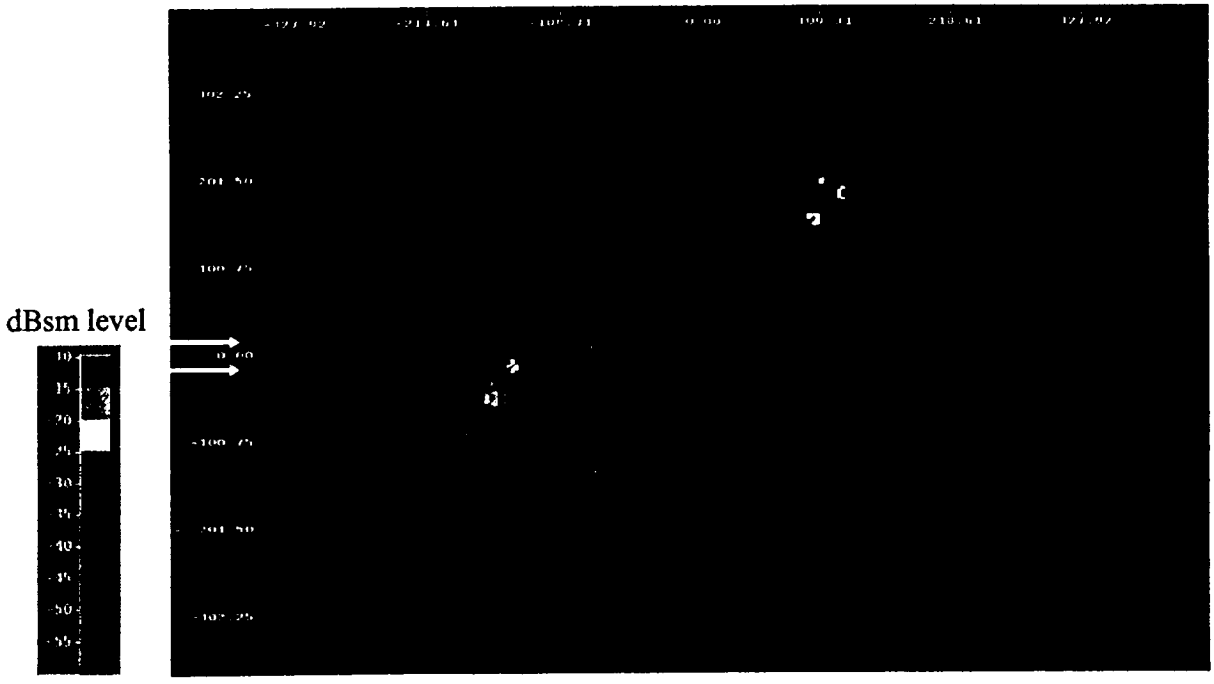


Fig. 7.

TABLE 2

	Total Time	Ray - trace time	Electromagnetic computation time
Direct Convolution	8 hrs.	1.27 hrs.	6.33 hrs.
Sullivan Scheme	1.30 hrs.	1.27 hrs.	2 mins.

Table 2. The ISAR image formation time for the generic aircraft using direct convolution and Sullivan's scheme. The timings are for the Silicon Graphics Indigo (R3000) workstation.