

Wave scattering with UV multilevel partitioning method: 2. Three-dimensional problem of nonpenetrable surface scattering

Leung Tsang,¹ Qin Li,² Peng Xu,¹ Dong Chen,¹ and Vikram Jandhyala²

Received 18 November 2003; revised 21 April 2004; accepted 1 July 2004; published 9 October 2004.

[1] A UV multilevel partitioning method (UV-MLP) is developed to solve scalar wave three-dimensional (3-D) scattering problem. The method consists of setting up a table of transmitting and receiving block size and their separation using fast coarse-coarse sampling. For a specific scattering problem with given geometry, the scattering structure is partitioned into multilevel blocks. By looking up the rank in the static problem, the impedance matrix for a given transmitting and receiving block is expressed into a product of U and V matrix. In this paper the method is illustrated by applying to a 3-D scattering problem of random nonpenetrable rough surface. The cases of Dirichlet and Neumann boundary conditions are treated. Numerical simulation results are illustrated. For 65,536 boundary unknowns on a rough surface, and using a single processor of 2.66 GHz, it takes about 34 CPU min and 1.8 Gb of memory to compute the solution using conjugate gradient iterations and multilevel UV to accelerate the matrix-column vector multiplication. *INDEX TERMS*: 0644 Electromagnetics: Numerical methods; 0659 Electromagnetics: Random media and rough surfaces; 0669 Electromagnetics: Scattering and diffraction; *KEYWORDS*: wave scattering, fast solver, rough surface

Citation: Tsang, L., Q. Li, P. Xu, D. Chen, and V. Jandhyala (2004), Wave scattering with UV multilevel partitioning method: 2. Three-dimensional problem of nonpenetrable surface scattering, *Radio Sci.*, 39, RS5011, doi:10.1029/2003RS003010.

1. Introduction

[2] With the advent of modern computers and the development of fast numerical methods, Monte Carlo simulations of the wave scattering problem, e.g., random rough surface scattering, have become an attractive approach. A common method used in numerical simulations is the surface integral equation method [Maradudin *et al.*, 1990; Nieto-Vesperinas and Soto-Crespo, 1987; Tsang *et al.*, 1993, 1994; Pak *et al.*, 1995; Jandhyala *et al.*, 1998] and its solution by the method of moments (MoM). Conventional implementation of the MoM requires an $O(N^3)$ operation and an $O(N^2)$ computer memory storage. Two fast numerical methods have been used and they are the sparse matrix canonical grid method (SMCG) [Tsang *et al.*, 1993, 1994; Pak *et al.*, 1995] and the Fast Multipole Method (FMM) [Jandhyala *et al.*, 1998]. Both methods

have been applied to perfect electric conducting surfaces and dielectric surfaces.

[3] Recently, Kapur and Long [1997] proposed a QR decomposition method by using the property of the smoothness of the impedance matrix elements. The QR is applied to the matrix blocks with recursive partitioning and merging. The method has so far been demonstrated only for object of moderate size.

[4] In this paper, a UV method with multilevel partitioning (UV-MLP) is developed to solve wave scattering problem. The method consists of setting up a table of transmitting and receiving block size and their separation. For a specific scattering problem with given geometry, the scattering structure is partitioned into multilevel blocks. By looking up the rank in the static problem, the impedance matrix for a given transmitting and receiving block is expressed into a product of U and V matrix. In the method, we also apply the multilevel matrix partitioning (MLP) to partition the impedance matrix \bar{Z} . We demonstrate the technique for a 3-D rough surface scattering problem. The surface area of the rough surface when projected onto the xy plane has a surface area up to 256 square wavelengths with the number of surface unknowns of 65,536. The RMS height is 0.5 wavelength. In this case, the multilevel partitioning used is as done in the multilevel Fast multipole method. Because the UV is

¹Department of Electrical Engineering, City University of Hong Kong, Hong Kong, China.

²Department of Electrical Engineering, University of Washington, Seattle, Washington, USA.

applied independently to each level and each block, the procedure facilitates parallel implementation.

[5] In section 2, we describe the problem-independent rank determination. This is done for a flat surface and volume scattering. In section 3, the formulation of the problem of wave impinging upon a rough surface is given in terms of surface integral equation, which is converted into a matrix equation using the method of moments (MoM). In section 4, we describe the multi-level partitioning process and in section 5 we describe the UV method. In section 6, the computational complexity of the proposed algorithm is derived. In section 7, numerical results are illustrated and discussed, and we give conclusions in section 8.

2. Problem-Independent Rank Table

[6] Consider a matrix $\bar{\bar{Z}}$ of dimension $N \times N$, which represents the interactions of two nonnear groups and each of them has N scatterers in the group, the rank of $\bar{\bar{Z}}$ is r . We can use the SVD to determine rank. Let σ_1 be the largest singular value, and the singular values be arranged in decreasing magnitude. Given a threshold ε , the rank r is such that $|\sigma_{r+1}/\sigma_1| \leq \varepsilon$. The threshold used in the simulation is 1.e-5 in the paper. Owing to the smoothness of the matrix elements, the rank r is much less than the number of scatterers N . The rank r is mainly determined with the transmitting and receiving block size and their separation distance between two block centers.

[7] Consider a transmitting block and a receiving block and there are N points in each block. The points are label \bar{r}_m , $m = 1, 2, \dots, N$, for the receiving block and \bar{r}_n , $n = 1, 2, \dots, N$, for the transmitting block. Then we have the matrix $\bar{\bar{Z}}$ of dimension $N \times N$ with elements for Dirichlet boundary condition

$$Z_{mn} = \frac{\exp(jk|\bar{r}_m - \bar{r}_n|)}{4\pi|\bar{r}_m - \bar{r}_n|}. \quad (1)$$

[8] For N , we assume 10 points per wavelength that corresponds to 100 points per square wavelength and 1000 points per cubic wavelength.

[9] The rank of the matrix can be determined. It is strongly dependent on the size of the transmitting and receiving blocks and the separation between them.

2.1. Two Flat Sheets With Zero Vertical Separation

[10] Consider two flat sheets of sizes L_x and L_y , and they are both placed on the same horizontal plane. The centers of the two planes are on the x axis and let R be the separation between the two centers. This case can be met for the problem of microstrip line and it is usually assumed to be current sheet. To determine the rank

Table 1. Rank Table of Volume

$L_x = L_y, \lambda$	Points in Block	Distance, λ	L_z, λ	Rank
1.0	100	2.00	0	14, ^a 13, ^b 12 ^c
1.0	100	2.23	0	13, 13, 12
1.0	100	2.83	0	11, 9, 9
1.0	100	3.60	0	9, 9, 8
1.0	100	4.24	0	9, 8, 8
2.0	400	4.00	0	17, 15, 15
2.0	400	4.47	0	15, 14, 14
2.0	400	5.65	0	13, 12, 12
2.0	400	7.21	0	11, 11, 11
2.0	400	8.48	0	9, 10, 10
4.0	1600	8.00	0	23, 20, 20
4.0	1600	8.94	0	21, 18, 18
4.0	1600	11.3	0	17, 16, 16
4.0	1600	14.4	0	14, 13, 13
4.0	1600	17.0	0	11, 12, 12
1.0	600	2.00	0.6	30, ^d 34, ^e 32 ^f
1.0	800	2.00	0.8	35, 37, 36
1.0	1000	2.00	1.0	38, 43, 41
2.0	3200	4.00	0.8	37, 41, 41
2.0	4800	4.00	1.2	NA, ^g 49, 50
2.0	8000	4.00	2.0	NA, 65, 67
4.0	12,800	8.00	0.8	NA, 49, 49
4.0	38,400	8.00	2.4	NA, NA, 81
4.0	64,000	8.00	4.0	NA, NA, 116

^aArea sampling (100 points per square wavelength).

^bBoundary (line) dense sampling (10 points per wavelength).

^cBoundary (line) coarse sampling (4 points per wavelength).

^dVolume sampling (1000 points per cubic wavelength).

^eBoundary (surface) dense sampling (100 points per square wavelength).

^fBoundary (surface) coarse sampling (16 points per square wavelength).

^gNA, not available due to the limitation of computer resource.

for this problem traditionally, the area sampling has to be used. The area sampling leads to large number of sampling points and requires large CPU time and memory allocation for computer. However, based on Huygen's principle, the equivalent sources are the boundary lines which are the four sides of the sheet. Thus the radiation can be considered as radiation from the enclosing boundary lines of the transmitting block to the boundary lines of the receiving block. We can actually use the boundary line sampling in the rank determination. Furthermore, due to the smoothness of the Green's function, we know the rank is much less than number of sampling points. Thus we can use coarse boundary sampling to further reduce the computational cost for the rank determination. In Table 1, we list the ranks determined by these three ways and they are denoted by the cases with $L_z = 0$. For the area sampling, 100 points per square wavelength are used. For the boundary dense sampling, 10 points per wavelength are used. For the boundary coarse sampling, 4 points per wavelength are used. The ranks determined through them are comparable for the given physical sizes of the

problem. Note that the ranks are substantially smaller than the block size N .

2.2. Volumetric Blocks and Enclosing Boundary Radiation

[11] Next we consider volumetric blocks of sizes $L_x \times L_y \times L_z$. The centers of the blocks are placed on the x axis and separated by a distance R . We also used three ways to determine the ranks for a given physical sizes of the problem, they are volume sampling, boundary dense sampling, and boundary coarse sampling.

[12] For the boundary sampling, the equivalent sources are now the boundary surfaces which are the six sides of the block. Thus the radiation can be considered as radiation from the enclosing boundary surface of the transmitting block to the boundary surface of the receiving block. The ranks are given in Table 1, too. For volume sampling, 1000 points per cubic wavelength are used. For boundary dense sampling, 100 points per square wavelength are used. For boundary coarse sampling, 16 points per square wavelength are used. Note that for the $4\lambda \times 4\lambda \times 4\lambda$, the number of points for volume sampling is 64,000. While for boundary coarse sampling, the number of sampling points is only 1536. This will speed up the determination of the ranks greatly for the case. We note that for the case of the $4\lambda \times 4\lambda \times 4\lambda$ with $R = 8\lambda$ case, the number of points is $n = 40 \times 40 \times 40 = 64,000$. The rank from this case is only 116 meaning that large matrix compression is possible. The table also shows that coarse sampling essentially gives the same rank as dense sampling. The rank can be determined speedily using coarse-coarse sampling.

2.3. Small Overestimation of Rank

[13] In the SVD, the rank determination is based on a threshold. Thus there is built in variation of rank due to variations of threshold. Furthermore, in applications as illustrated in this paper, the exact rank is not required. Thus we usually are on the safe side, and use 10% to 20% above the ‘‘actual’’ rank.

[14] We note that the selections of blocks are dependent on the type of problem. However, once the static rank table is determined, the rank table can be applied to all cases within the same type of problem. Analytical treatment of rank determination for 2-D problem has been done in the work of *Tsang et al.* [2004, Appendix A]. Analytical treatment of 3-D can be done in a similar manner.

3. Wave Scattering Formulation

[15] Consider a tapered scalar plane wave $\psi_{inc}(x, y, z)$ [*Tsang et al.*, 2001, equation 6.1.1] impinging upon a 2-D random rough surface with Dirichlet and Neumann boundary condition and with a random height profile

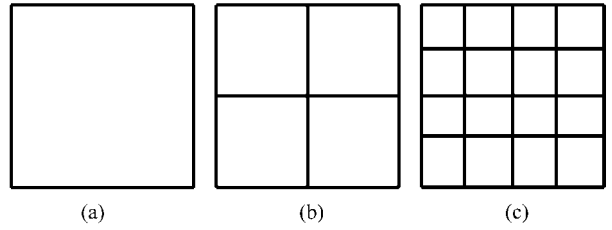


Figure 1. Illustration of multilevel partitioning process.

$z = f(x, y)$. A Fredholm integral equation of the first kind can be formed. Let $\vec{r}' = \hat{x}x' + \hat{y}y' + \hat{z}f(x', y')$ denote a field point and $\vec{r} = \hat{x}x + \hat{y}y + \hat{z}f(x, y)$ denote a source point on the rough surface. Then the surface integral equation for Dirichlet and Neumann boundary condition are [*Tsang et al.*, 2001], respectively,

$$0 = \psi_{inc}(\vec{r}') - \iint_P dx dy U(x, y) \cdot G_0(x, y, f(x, y); x', y', f(x', y')) \quad (2a)$$

$$\frac{1}{2} \psi(x', y') = \psi_{inc}(\vec{r}') + \iint_P dS \Psi(x, y) \cdot [\hat{n} \nabla G_0(x, y, f(x, y); x', y', f(x', y'))], \quad (2b)$$

where $G_0 = \frac{\exp(ik|\vec{r}-\vec{r}'|)}{4\pi|\vec{r}-\vec{r}'|}$ is the free-space Green's function, \iint_P denotes a principle value of integral and the unknown surface field are $U(x, y)$ and $\psi(x, y)$, respectively for the Dirichlet case and the Neumann case.

[16] The method of moments (MoM) is used to discretize the integral equation. We use the pulse basis function and point-matching method. The resulting matrix equation is

$$\overline{\overline{Z}} \cdot \overline{\overline{u}} = \overline{\overline{b}}. \quad (3)$$

4. Multilevel Partitioning Process

[17] Assuming that we have a square area as shown in Figure 1a. We first split it as 4 blocks as shown in Figure 1b. Each of 4 blocks is subgroup at the P th level, which has the largest group size. Then we split each subblock as another 4 small groups as shown in Figure 1c. This splitting process is continued until we reach the smallest group size, which is at the first level. We decompose the full impedance matrix in equation (3) as the sum of P sparse matrixes as in follows.

$$\overline{\overline{Z}} = \overline{\overline{Z}}^{(0)} + \overline{\overline{Z}}^{(1)} + \overline{\overline{Z}}^{(2)} + \dots + \overline{\overline{Z}}^{(P-1)} \quad (4)$$

Matrix $\overline{\overline{Z}}^{(0)}$ includes all the interactions among neighboring groups (including self group) at the first level. Matrix

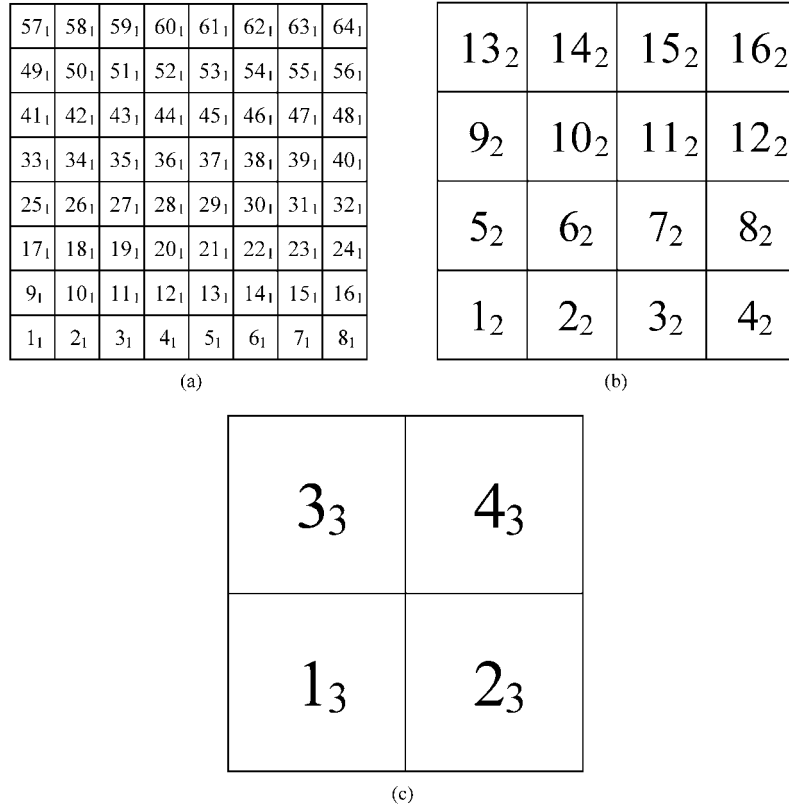


Figure 2. Illustration of multilevel partitioning process with 64 subgroups at the first level.

$\bar{Z}^{(1)}$ includes all the interactions among neighboring groups at the second level. It consists of blocks in the first level. Similarly, matrix $\bar{Z}^{(i)}$ includes all the interactions among neighboring groups at the $(i + 1)$ th level, but these consist of blocks in the i th level.

[18] To facilitate understanding of the multilevel partitioning process, we give an example that has 64 subgroups at the first level. Assume each group has M elements. For this example, the highest level is $P = 3$. Thus $\bar{Z} = \bar{Z}^{(0)} + \bar{Z}^{(1)} + \bar{Z}^{(2)}$ and the impedance matrix of \bar{Z} is $64 M$ by $64 M$ and has $4096 M$ by M blocks.

[19] In Figure 2a, the 64 subgroups of the first level are marked as $1_1, 2_1, \dots, 64_1$. In Figure 2b, the level 2 groups are shown and they are marked as $1_2, 2_2, \dots, 16_2$. Each of the level 2 groups has 4 level 1 groups. For example, the group 3_2 has 4 groups of $5_1, 6_1, 13_1$, and 14_1 . In Figure 2c, the level 3 groups are shown and they are marked as $1_3, 2_3, 3_3$, and 4_3 . Each of the level 3 groups has 4 level 2 groups. For example, the group 2_3 has 4 groups of $3_2, 4_2, 7_2$, and 8_2 .

[20] We use \underline{m}_i to represent the group m of the level i . Then matrix $\bar{Z}_{m,n}$ represents the interactions between the

receiving group m and transmitting group n of the i th level. Since the four level 1 groups form a level 2 group and 4 level 2 groups form a level 3 group, we have $\bar{Z}_{m,n}$, dimension of M by M and \bar{Z}_{m_2,n_2} , dimension of $4M$ by $4M$. The examples are

$$\bar{Z}_{3_2,5_2} = \begin{bmatrix} \bar{Z}_{5_1,17_1} & \bar{Z}_{5_1,18_1} & \bar{Z}_{5_1,25_1} & \bar{Z}_{5_1,26_1} \\ \bar{Z}_{6_1,17_1} & \bar{Z}_{6_1,18_1} & \bar{Z}_{6_1,25_1} & \bar{Z}_{6_1,26_1} \\ \bar{Z}_{13_1,17_1} & \bar{Z}_{13_1,18_1} & \bar{Z}_{13_1,25_1} & \bar{Z}_{13_1,26_1} \\ \bar{Z}_{14_1,17_1} & \bar{Z}_{14_1,18_1} & \bar{Z}_{14_1,25_1} & \bar{Z}_{14_1,26_1} \end{bmatrix}, \quad (5)$$

$$\bar{Z}_{6_2,5_2} = \begin{bmatrix} \bar{Z}_{19_1,17_1} & \bar{Z}_{19_1,18_1} & \bar{Z}_{19_1,25_1} & \bar{Z}_{19_1,26_1} \\ \bar{Z}_{20_1,17_1} & \bar{Z}_{20_1,18_1} & \bar{Z}_{20_1,25_1} & \bar{Z}_{20_1,26_1} \\ \bar{Z}_{27_1,17_1} & \bar{Z}_{27_1,18_1} & \bar{Z}_{27_1,25_1} & \bar{Z}_{27_1,26_1} \\ \bar{Z}_{28_1,17_1} & \bar{Z}_{28_1,18_1} & \bar{Z}_{28_1,25_1} & \bar{Z}_{28_1,26_1} \end{bmatrix}, \quad (6)$$

and

$$\bar{\bar{Z}}_{1,3,2_3} = \begin{bmatrix} \bar{\bar{Z}}_{1,2,3_2} & \bar{\bar{Z}}_{1,2,4_2} & \bar{\bar{Z}}_{1,2,7_2} & \bar{\bar{Z}}_{1,2,8_2} \\ \bar{\bar{Z}}_{2,2,3_2} & \bar{\bar{Z}}_{2,2,4_2} & \bar{\bar{Z}}_{2,2,7_2} & \bar{\bar{Z}}_{2,2,8_2} \\ \bar{\bar{Z}}_{5,2,3_2} & \bar{\bar{Z}}_{5,2,4_2} & \bar{\bar{Z}}_{5,2,7_2} & \bar{\bar{Z}}_{5,2,8_2} \\ \bar{\bar{Z}}_{6,2,3_2} & \bar{\bar{Z}}_{6,2,4_2} & \bar{\bar{Z}}_{6,2,7_2} & \bar{\bar{Z}}_{6,2,8_2} \end{bmatrix}. \quad (7)$$

4.1. Blocks in $\bar{\bar{Z}}^{(0)}$

[21] In $\bar{\bar{Z}}^{(0)}$, we select interactions of the level 1 groups with their nearest neighbors. For example, 20_1 has 8 neighbors of 11_1 , 12_1 , 13_1 , 19_1 , 21_1 , 27_1 , 28_1 , and 29_1 . Thus $\bar{\bar{Z}}^{(0)}$ includes $\bar{\bar{Z}}_{11,20_1}$, $\bar{\bar{Z}}_{12,20_1}$, $\bar{\bar{Z}}_{13,20_1}$, $\bar{\bar{Z}}_{19,20_1}$, $\bar{\bar{Z}}_{20,20_1}$, $\bar{\bar{Z}}_{21,20_1}$, $\bar{\bar{Z}}_{27,20_1}$, $\bar{\bar{Z}}_{28,20_1}$, and $\bar{\bar{Z}}_{29,20_1}$, a total of 9 matrices. Note that self-interaction is also included in here. We note that: (1) block size of $\bar{\bar{Z}}_{m_1, n_1}$ is M by M ; (2) there are 36 interior first-level groups of 9 blocks = $36 \times 9 = 324$ (M by M) blocks; (3) 24 edge first-level groups of 6 blocks = $24 \times 6 = 144$ (M by M) blocks; and (4) 4 corner first-level groups of 4 blocks = $4 \times 4 = 16$ (M by M) blocks. Thus $\bar{\bar{Z}}^{(0)}$ has total of $324 + 144 + 16 = 484$ (M by M) blocks.

4.2. Blocks in $\bar{\bar{Z}}^{(1)}$

[22] In $\bar{\bar{Z}}^{(1)}$, we select the interactions between level 2 groups and their nearest neighbors. For example, we need to include $\bar{\bar{Z}}_{6,5_2}$. However, we need to exclude those that have been included $\bar{\bar{Z}}^{(0)}$. We define the impedance matrix primes:

$$\bar{\bar{Z}}_{6,5_2} = \begin{bmatrix} \bar{\bar{Z}}_{19,17_1} & \bar{\bar{0}}_1 & \bar{\bar{Z}}_{19,25_1} & \bar{\bar{0}}_1 \\ \bar{\bar{Z}}_{20,17_1} & \bar{\bar{Z}}_{20,18_1} & \bar{\bar{Z}}_{20,25_1} & \bar{\bar{Z}}_{20,26_1} \\ \bar{\bar{Z}}_{27,17_1} & \bar{\bar{0}}_1 & \bar{\bar{Z}}_{27,25_1} & \bar{\bar{0}}_1 \\ \bar{\bar{Z}}_{28,17_1} & \bar{\bar{Z}}_{28,18_1} & \bar{\bar{Z}}_{28,25_1} & \bar{\bar{Z}}_{28,26_1} \end{bmatrix}, \quad (8)$$

where $\bar{\bar{0}}_1$ is the zero matrix of dimension (M by M). Thus the $\bar{\bar{Z}}^{(1)}$, includes all the $\bar{\bar{Z}}_{m_2, n_2}$ where m_2 and n_2 are neighbors. As shown in equation (8), the matrix $\bar{\bar{Z}}_{m_2, n_2}$ (1) consists of blocks with size of M by M , (2) each block consists of a transmitting and a receiving region that are not neighbors of each other in level one groups, e.g., in $\bar{\bar{Z}}_{19,17_1}$ the receiving region is 19_1 and the transmitting region is 17_1 and the two are not neighbors of each other, and (3) the separation R between the transmitting and the receiving regions is at a minimum of $2S$ for $\bar{\bar{Z}}_{19,17_1}$, where S is the block size. It is at a maximum of $R = 3\sqrt{2}S$ for $\bar{\bar{Z}}_{44,17_1}$ which is in $\bar{\bar{Z}}_{10,5_2}$. Thus $\bar{\bar{Z}}^{(1)}$ includes:

[23] 1. Four interior level 2 groups which have 8 neighbors. 4 of them have 12 blocks while 4 of them have 15 blocks. The total blocks are $4 \times (4 \times 12 + 4 \times 15) = 432$.

[24] 2. Eight edge level 2 groups which have 5 neighbors. 3 of them have 12 blocks while 2 of them have 15 blocks. The total blocks are $8 \times (3 \times 12 + 2 \times 15) = 528$.

[25] 3. Four corner level 2 groups which have 3 neighbors. 2 of them have 12 blocks and 1 has 15 blocks. The total blocks are $4 \times (2 \times 12 + 1 \times 15) = 156$.

[26] Thus matrix $\bar{\bar{Z}}^{(1)}$ includes $432 + 528 + 156 = 1116$ (M by M) blocks.

4.3. Blocks in $\bar{\bar{Z}}^{(2)}$

[27] In $\bar{\bar{Z}}^{(2)}$, we select the interactions between level 3 groups and their nearest neighbors. For example, we need to include $\bar{\bar{Z}}_{1,3,2_3}$. However, some of the interactions have already been included in $\bar{\bar{Z}}^{(0)}$ and $\bar{\bar{Z}}^{(1)}$ and need to be excluded. We define the prime impedance matrices

$$\bar{\bar{Z}}'_{1,3,2_3} = \begin{bmatrix} \bar{\bar{Z}}_{1,2,3_2} & \bar{\bar{Z}}_{1,2,4_2} & \bar{\bar{Z}}_{1,2,7_2} & \bar{\bar{Z}}_{1,2,8_2} \\ \bar{\bar{0}}_2 & \bar{\bar{Z}}_{2,2,4_2} & \bar{\bar{0}}_2 & \bar{\bar{Z}}_{2,2,8_2} \\ \bar{\bar{Z}}_{5,2,3_2} & \bar{\bar{Z}}_{5,2,4_2} & \bar{\bar{Z}}_{5,2,7_2} & \bar{\bar{Z}}_{5,2,8_2} \\ \bar{\bar{0}}_2 & \bar{\bar{Z}}_{6,2,4_2} & \bar{\bar{0}}_2 & \bar{\bar{Z}}_{6,2,8_2} \end{bmatrix}. \quad (9)$$

Note that the building blocks of $\bar{\bar{Z}}'_{1,3,2_3}$ are level 2 blocks and are of size of $(4 M$ by $4 M) = 16 M$ by M . This is important that as the level goes up, the block size will grow by 4 times as we go up each level. We have 4 level 3 groups and each of them has 2 neighbors with 12 level 2 blocks and 1 neighbor with 15 level 2 blocks. Thus we have $4 \times (2 \times 12 + 1 \times 15) = 156$ level 2 blocks. Since each level 2 block has 16 level 1 blocks. Thus we have a total of $16 \times 156 = 2496$ level 1 blocks in $\bar{\bar{Z}}^{(2)}$.

[28] Thus the total count is $484 + 1116 + 2496 = 4096$. Thus all the level 1 blocks are counted exactly once. To summarize, the block size in $\bar{\bar{Z}}^{(0)}$ is of M by M , in $\bar{\bar{Z}}^{(1)}$ is of M by M , and in $\bar{\bar{Z}}^{(2)}$ is of $4 M$ by $4 M$. In applying UV decomposition, the UV is applied to each block for that level. Each block consists of a transmitting and a receiving region that are not neighbors of each other. However, their separations are within a restricted range as indicated before.

5. UV Method Based on Interpolation Technique

[29] The matrix $\bar{\bar{Z}}_{m_i, n_i}$, which represents the interactions of two nonneighbor groups m_i and n_i , can be represented by UV decomposition. The matrix $\bar{\bar{Z}}_{m_i, n_i}$ is of dimensions $4^{(i-1)}M$ by $4^{(i-1)}M$. The rank of $\bar{\bar{Z}}_{m_i, n_i}$, r , is much smaller than $4^{(i-1)}M$. For simpler notation, we denote $\bar{\bar{Z}}_{m_i, n_i}$ by \bar{A} , which has dimension of N by N and rank of r with $r \ll N$. To decompose \bar{A} by the SVD and Gram-Schmidt process will consume CPU because N is large. Note that in section 2, we apply the SVD to find r . However, because of coarse sampling, the selected matrices are of dimension roughly of r by r only.

[30] Let the column of \bar{A} be denoted by \bar{a}_i , where $i = 1, 2, \dots, N$. Then $A_{mn} = (\bar{a}_n)_m$. The element A_{mn} is the

m th element of the column vector \bar{a}_n . However there are only r independent columns. In the transmitting region, we select r points from the N points. The r points must be uniformly distributed in the transmitting region.

[31] We compute the r columns \bar{u}_l , $l = 1, 2, \dots, r$. Each column is of dimension of N and coincides with a column of \bar{A} :

$$U_{ml} = (\bar{u}_l)_m = A_{mp(l)}, m = 1, 2, \dots, N, \quad (10)$$

where $p(l)$ is a column index of \bar{A} that depends on l . Note that to get U_{ml} , one needs to go through all the N points in the receiving region. Thus the matrix \bar{U} has Nr elements.

[32] Because of linear independence, any general column \bar{a}_m of \bar{A} is a linear combination of \bar{u}_l , that is,

$$\bar{a}_m = \sum_{l=1}^r v_{lm} \bar{u}_l, m = 1, 2, \dots, N, \quad (11)$$

where v_{lm} are the coefficients to be determined.

[33] We pick r rows of \bar{A} , which has total of Nr elements. The r rows correspond to r points in the receiving group. The r points must be uniformly distributed in the receiving group. We first put these rows in a matrix of \bar{R} .

$$R_{m_a p} = A_{m(m_a)p}, p = 1, 2, \dots, N \text{ and } m_a = 1, 2, \dots, r \quad (12)$$

We pick the m_a rows in \bar{u}_l , $l = 1, 2, \dots, r$. That will give us $r \times r$ matrix and we call it \tilde{U} and we have $\tilde{U}_{m_a n_a} = (\bar{u}_{n_a})_{m(m_a)}$. Then we set

$$R_{m_a l} = \sum_{n_a=1}^r \tilde{U}_{m_a n_a} v_{n_a l}, l = 1, 2, \dots, N. \quad (13)$$

In matrix notation, $\bar{V} = r \times N$, $\bar{R} = r \times N$, $\bar{U} = r \times r$, $\bar{R} = \bar{U} \bar{V}$, and $\bar{V} = (\bar{U})^{-1} \bar{R}$. This completes the UV decomposition.

$$\bar{A} = \bar{U} \bar{V} \quad (14)$$

To summarize, we take r columns of \bar{A} and then r rows of N , a total of $2rN$ elements. We need to take the inverse of an r by r matrix \bar{U} and a matrix multiplication of r by r matrix times r by N matrix. The computational and memory efficiency is achieved when $r \ll N$.

6. Computational Complexity Analysis

6.1. Multilevel Group Sizes and Number of Groups

[34] The rough surface is generated in a square area with \sqrt{N} points in x direction and \sqrt{N} points in y

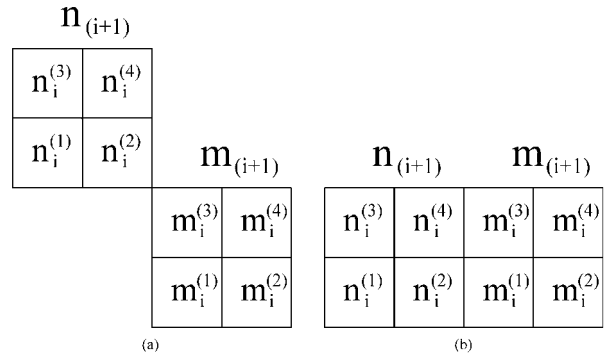


Figure 3. Illustration of block interactions at the i th level.

direction. So the total number of patches is N . We use P level various subgroup sizes to partition the whole area as follows:

[35] 1. At the P th level, we split the whole area as 4 groups. Each group has $N/4$ elements:

$$L_P = 4 = 2^{2(P-P+1)} \quad (15)$$

$$M_P = N/4 = N/L_P, \quad (16)$$

where L_P is number of groups and M_P is number of elements of each group at the P th level.

[36] 2. Split each group as 4 subgroups and continue this partitioning, at i th level, we have

$$L_i = 2^{2(P-i+1)} \quad (17)$$

$$M_i = N/L_i. \quad (18)$$

[37] 3. At the first level, we have

$$L_1 = 2^{2P} \quad (19)$$

$$M_1 = N/L_1. \quad (20)$$

6.2. Cost Function at the i th Level

[38] In $\bar{Z}^{(i)}$, we select the interactions between level $(i+1)$ groups and their nearest neighbors. However, some of the interactions have already been included in the lower level groups and must be excluded from them. There are two kinds of neighboring groups. One is that sharing only one common point and the other is that sharing one common edge.

[39] 1. Sharing only one common point (Figure 3a): For the neighboring groups of $m_{(i+1)}$ and $n_{(i+1)}$ shown in Figure 3a, only one common point is shared

between them. The impedance matrix that will be calculated is

$$\bar{\bar{Z}}'_{m_{(i+1)}n_{(i+1)}} = \begin{bmatrix} \bar{\bar{Z}}_{m_i^{(1)}n_i^{(1)}} & \bar{\bar{Z}}_{m_i^{(1)}n_i^{(2)}} & \bar{\bar{Z}}_{m_i^{(1)}n_i^{(3)}} & \bar{\bar{Z}}_{m_i^{(1)}n_i^{(4)}} \\ \bar{\bar{Z}}_{m_i^{(2)}n_i^{(1)}} & \bar{\bar{0}}_i & \bar{\bar{Z}}_{m_i^{(2)}n_i^{(3)}} & \bar{\bar{Z}}_{m_i^{(2)}n_i^{(4)}} \\ \bar{\bar{Z}}_{m_i^{(3)}n_i^{(1)}} & \bar{\bar{Z}}_{m_i^{(3)}n_i^{(2)}} & \bar{\bar{Z}}_{m_i^{(3)}n_i^{(3)}} & \bar{\bar{Z}}_{m_i^{(3)}n_i^{(4)}} \\ \bar{\bar{Z}}_{m_i^{(4)}n_i^{(1)}} & \bar{\bar{Z}}_{m_i^{(4)}n_i^{(2)}} & \bar{\bar{Z}}_{m_i^{(4)}n_i^{(3)}} & \bar{\bar{Z}}_{m_i^{(4)}n_i^{(4)}} \end{bmatrix}. \quad (21)$$

The dimension of matrix $\bar{\bar{Z}}_{m_i n_i}$ is of M_i by M_i . The computational steps for $\bar{\bar{Z}}_{m_i n_i} \cdot \bar{b}_{M_i}$ through the UV is of $2M_i r_i$, where r_i is the rank of $\bar{\bar{Z}}_{m_i n_i}$. In here we assume the rank r_i of $\bar{\bar{Z}}_{m_i n_i}$ is same for all block interactions at the same level. There are 15 nonzero matrices of $\bar{\bar{Z}}_{m_i n_i}$ in the $\bar{\bar{Z}}'_{m_{(i+1)}n_{(i+1)}}$. The total computational steps for $\bar{\bar{Z}}'_{m_{(i+1)}n_{(i+1)}} \cdot \bar{b}_{M_{(i+1)}}$ are $30M_i r_i$.

[40] 2. Sharing one common edge (Figure 3b): For the neighboring groups of $m_{(i+1)}$ and $n_{(i+1)}$ shown in Figure 3b, one common edge is shared between them. The impedance matrix that will be calculated is

$$\bar{\bar{Z}}'_{m_{(i+1)}n_{(i+1)}} = \begin{bmatrix} \bar{\bar{Z}}_{m_i^{(1)}n_i^{(1)}} & \bar{\bar{0}}_i & \bar{\bar{Z}}_{m_i^{(1)}n_i^{(3)}} & \bar{\bar{0}}_i \\ \bar{\bar{Z}}_{m_i^{(2)}n_i^{(1)}} & \bar{\bar{Z}}_{m_i^{(2)}n_i^{(2)}} & \bar{\bar{Z}}_{m_i^{(2)}n_i^{(3)}} & \bar{\bar{Z}}_{m_i^{(2)}n_i^{(4)}} \\ \bar{\bar{Z}}_{m_i^{(3)}n_i^{(1)}} & \bar{\bar{0}}_i & \bar{\bar{Z}}_{m_i^{(3)}n_i^{(3)}} & \bar{\bar{0}}_i \\ \bar{\bar{Z}}_{m_i^{(4)}n_i^{(1)}} & \bar{\bar{Z}}_{m_i^{(4)}n_i^{(2)}} & \bar{\bar{Z}}_{m_i^{(4)}n_i^{(3)}} & \bar{\bar{Z}}_{m_i^{(4)}n_i^{(4)}} \end{bmatrix}. \quad (22)$$

There are 12 nonzero matrices of $\bar{\bar{Z}}_{m_i n_i}$ in the $\bar{\bar{Z}}'_{m_{(i+1)}n_{(i+1)}}$. The total computational steps for $\bar{\bar{Z}}'_{m_{(i+1)}n_{(i+1)}} \cdot \bar{b}_{M_{(i+1)}}$ are $24M_i r_i$.

[41] 3. Interior groups: Interior groups have 8 neighbors. 4 of them share only one common point and 4 of them share one common edge. There are $(L_{(i+1)}^{1/2} - 2)^2$ interior groups at $(i + 1)$ th level. Thus computational steps for the interior groups at the i th level are

$$\begin{aligned} & \left(L_{(i+1)}^{1/2} - 2\right)^2 (4 * 30M_i r_i + 4 * 24M_i r_i) \\ & = 216 \left(L_{(i+1)}^{1/2} - 2\right)^2 M_i r_i. \end{aligned} \quad (23)$$

[42] 4. Edge groups: Edge groups have 5 neighbors. 2 of them share only one common point and 3 of them share one common edge. There are $4(L_{(i+1)}^{1/2} - 2)$ edge groups at $(i + 1)$ th level. Thus computational steps for the edge groups at the i th level are

$$\begin{aligned} & 4 \left(L_{(i+1)}^{1/2} - 2\right) (2 * 30M_i r_i + 3 * 24M_i r_i) \\ & = 528 \left(L_{(i+1)}^{1/2} - 2\right) M_i r_i. \end{aligned} \quad (24)$$

[43] 5. Corner groups: Corner groups have 3 neighbors. One of them shares only one common point and 2 of them share one common edge. There are 4 corner groups at $(i + 1)$ th level. Thus computational steps for the corner groups at the i th level are

$$4(1 * 30M_i r_i + 2 * 24M_i r_i) = 312M_i r_i. \quad (25)$$

[44] 6. Total computational steps at the i th level are

$$\begin{aligned} & 216 \left(L_{(i+1)}^{1/2} - 2\right)^2 M_i r_i + 528 \left(L_{(i+1)}^{1/2} - 2\right) M_i r_i + 312M_i r_i \\ & = 2M_i r_i (108L_{(i+1)} - 168L_{(i+1)}^{1/2} + 60). \end{aligned} \quad (26)$$

6.3. Cost Function for the Multilevel UV Method (Nonnear)

[45] The total computational steps for nonneighbor block interactions through the multilevel UV decomposition are then

$$\sum_{i=1}^{P-1} 2r_i M_i \left[108L_{(i+1)} - 168L_{(i+1)}^{1/2} + 60\right]. \quad (27)$$

[46] Generally speaking, the rank of each level block will increase with the increase of block size. For simplicity, we assume $r_i = \text{constant} = r$ in here and take this r into equation (27), we have

$$27rN \log_2 \left(\frac{N}{M_1}\right) + 60rN - 120rM_1. \quad (28)$$

6.4. Cost Function for Near-Field Interactions in $\bar{\bar{Z}}^{(0)}$

[47] All the near-field interactions are included in the matrix $\bar{\bar{Z}}^{(0)}$, which is computed directly as the original impedance matrix. The block size in $\bar{\bar{Z}}^{(0)}$ is of dimension of M_1 by M_1 . The computational steps for the block and block interaction are M_1^2 .

[48] 1. Interior group: Interior group has 9 blocks (including self-interaction). There are $(L_1^{1/2} - 2)^2$ interior groups at the first level. Thus computational steps for the interior groups at the first level are $9(L_1^{1/2} - 2)^2 M_1^2$.

[49] 2. Edge groups: Edge group has 6 blocks (including self-interaction). There are $4(L_1^{1/2} - 2)$ edge groups at the first level. Thus computational steps for the edge groups at the first level are $6(L_1^{1/2} - 2)M_1^2$.

[50] 3. Corner groups: Corner group has 4 blocks (including self). There are 4 corner groups at the first level. Thus computational steps for the corner groups at the first level are $4M_1^2$.

Table 2. Rank Table of Rough Surface for the Case of Dirichlet Boundary Condition^a

$L_x = L_y, \lambda$	Points in Square	Distance, λ	L_z, λ	Rank
1.0	64	2.00	1.402	15
2.0	256	4.00	1.824	25
4.0	1024	8.00	3.219	35

^aFrom one realization of rough surfaces with RMS height of 0.5 wavelength and correlation length of 0.707 wavelength. Surface area is 16 by 16 square wavelength and 64 points per square wavelength are used in generating the rough surface. For rank determination, 16 points per square wavelength are taken uniformly from the rough surfaces.

[51] 4. The total computational steps for near-field interaction is then

$$9\left(L_1^{1/2} - 2\right)^2 M_1^2 + 6\left(L_1^{1/2} - 2\right) M_1^2 + 4M_1^2 = \left(9NM_1 - 12N^{1/2}M_1^{3/2} + 4M_1^2\right). \quad (29)$$

6.5. Cost Function

[52] The total computational steps for multilevel partitioning UV is sum of the near and nonnear interactions and is given by

$$27rN \log_2 \left(\frac{N}{M_1}\right) + 60rN - 120rM_1 + \left(9NM_1 - 12N^{1/2}M_1^{3/2} + 4M_1^2\right). \quad (30)$$

7. Numerical Results and Discussion

[53] The numerical simulation results are presented in terms of the bistatic scattering coefficients. After the unknown $U(x, y)$ or $\psi(x, y)$ is completed, the bistatic scattering coefficient $\gamma(\hat{k}_s)$ in the direction of \hat{k}_s is calculated by a weighted integration of the surface field (as indicated in the work of *Tsang et al.* [2001, equation 6.1.16]), in which $F(\hat{k}_s)$ is expressed in the work of *Tsang et al.* [2001, equation 6.1.17] for Dirichlet boundary condition. For the case of Neumann boundary condition, $F(\hat{k}_s)$ is

$$F(\hat{k}_s) = \iint dx dy \psi(x, y) e^{-ik(\sin \theta_s \cos \phi_s x + \sin \theta_s \sin \phi_s y + \cos \theta_s f(x, y))} \\ \times ik(f_x(x, y) \sin \theta_s \cos \phi_s + f_y(x, y) \sin \theta_s \sin \phi_s - \cos \theta_s). \quad (31)$$

Simulations are based on Gaussian random rough surfaces with Gaussian correlation functions. The computer used in the simulations is a Pentium single processor of 2.66 GHz with 2 Gb of memory. We have implemented both the SVD based QR method and the

UV method proposed by this paper. First, we will illustrate the rank for the rough surface.

7.1. Rank Determination for Random Rough Surface

[54] For the rough surface scattering problem, the difficulty is that the vertical sizes of the blocks are always changed due to the randomness of surface height. Thus we have to use the coarse-coarse area sampling to determine the rank. In the simulation, 16 points per square wavelength (instead of 100 points per square wavelength) are used for rank determination. In Table 2, we show the ranks as functions of the horizontal size, vertical size, and distance between two group centers for the case of Dirichlet boundary condition. The vertical size is defined as the maximum height of the surface minus the minimum height of the surface in the same block. The results are obtained through one realization of rough surface profiles with the given RMS height $h = 0.5\lambda$ and correlation length $l = 0.707\lambda$.

7.2. CPU for the UV

[55] In Table 3, we list the CPU time based on the UV method for different number of surface unknowns for the case of Dirichlet boundary condition. The parameters are $h = 0.5\lambda$, $l = 0.707\lambda$ for all the cases. The preprocessing time of the UV method is time used to find the rank based on coarse-coarse sampling and the time to construct U and V matrix. The CPU time for CG means the CPU time spent for solving the matrix equation after the matrix-filling and preprocessing is finished. For the case of 65,536 surface unknowns, it takes about 34 min for the total CPU.

7.3. Comparisons of the Bistatic Scattering Coefficients Simulated From Different Methods for Single Realization

[56] In Figure 4, we plot the bistatic scattering coefficients as a function of the scattering angles for the case of Dirichlet boundary condition. $h = 0.5\lambda$, $l = 0.707\lambda$ and the incidence angle $\theta_i = 20^\circ$. The surface lengths are $L_x = 8\lambda$ by $L_y = 8\lambda$ and we use 64 points per square wavelength to generate the rough surface. The results

Table 3. CPU for the UV method for the Case of Dirichlet Boundary Condition

Number of Unknowns	Preprocessing, s	CG, s	Total CPU, s
4096	6 ^a + 5 ^b	14	25
16,384	291 + 54	170	515
65,536	291 + 500	1220	2011

^aCPU for rank determination with coarse-coarse sampling.

^bCPU for building UV matrix.

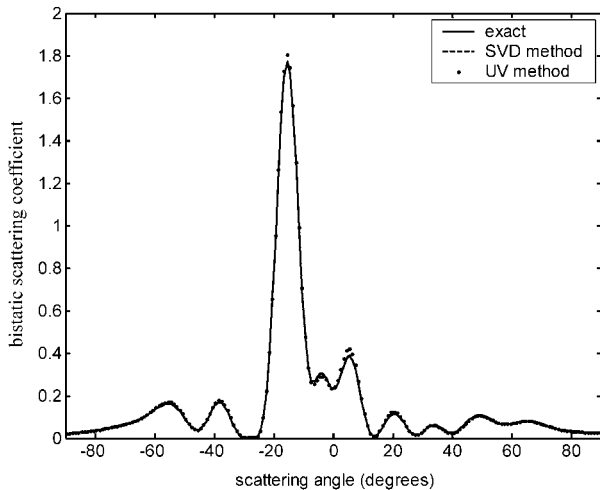


Figure 4. Comparisons of the bistatic scattering coefficients between the numerical simulations from three approaches of the exact solution, the SVD-based QR method, and the UV method for the case of Dirichlet boundary condition. Here $h = 0.5\lambda$, $l = 0.707\lambda$, $\theta_i = 20^\circ$, $L_x = 8\lambda$, and $L_y = 8\lambda$.

simulated from three methods are shown, which are exact solution of the matrix equation, solution from the SVD based QR method, and the solution of the UV method. All of them are in a good agreement.

[57] In Figure 5, we plot the same results except that the surface lengths are extended to $L_x = 16\lambda$ by $L_y = 16\lambda$. Owing to the limitation of computer memory, we cannot solve the matrix equation with the exact solution. We only show the simulation results from the SVD-based QR and the UV method. The two results match each other very well.

[58] To demonstrate that the proposed algorithm can be used also for large number of surface unknowns, we plot the simulation results for the case of Dirichlet boundary condition with $h = 0.5\lambda$, $l = 0.707\lambda$ and $\theta_i = 20^\circ$ in Figure 6. $L_x = 16\lambda$ by $L_y = 16\lambda$, but we use 256 points per square wavelength to sample the rough surface. Again, very good agreements are observed at all scattering angles between the simulation results from the SVD-based QR method and the UV.

[59] In Figure 7, we plot the bistatic scattering coefficient as a function of the scattering angles for the case of Neumann boundary condition. For this case, the parameters are $h = 0.5\lambda$, $l = 0.707\lambda$, $\theta_i = 20^\circ$, $L_x = 16\lambda$, $L_y = 16\lambda$. We use 64 points per square wavelength to generate the rough surface with the number of boundary unknowns $N = 16384$. The results simulated from the SVD and the UV method are compared. They are in a good agreement. The CPU are also shown in Table 4. It

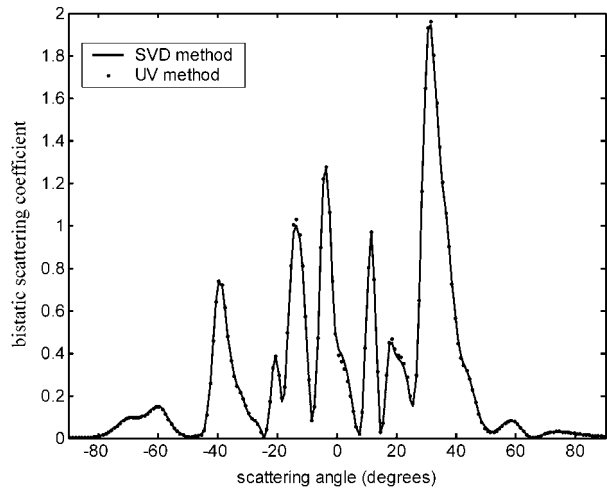


Figure 5. Comparisons of the bistatic scattering coefficients between the numerical simulations from two approaches of the SVD-based QR method and the UV method for the case of Dirichlet boundary condition. Here $h = 0.5\lambda$, $l = 0.707\lambda$, $\theta_i = 20^\circ$, $L_x = 16\lambda$, and $L_y = 16\lambda$; 64 points per square wavelength are used in the rough surface sampling.

shows that the preprocessing CPU of the SVD is much more than that of the UV method because the SVD is based on full sampling while the UV is based on coarse-coarse sampling and interpolation technique. On the

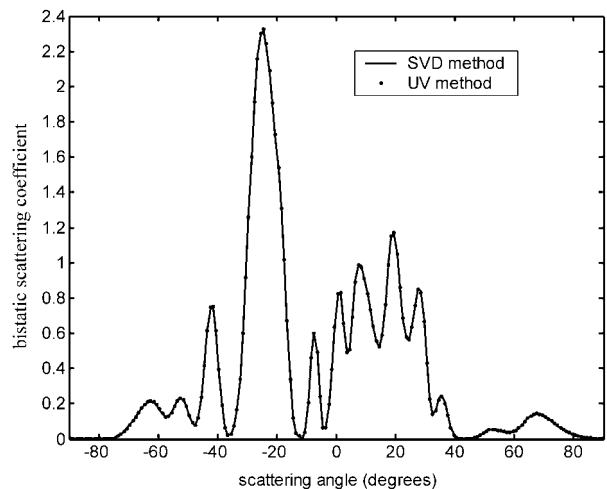


Figure 6. Comparisons of the bistatic scattering coefficients between the numerical simulations from two approaches of the SVD-based QR method and the UV method for the case of Dirichlet boundary condition. Here $h = 0.5\lambda$, $l = 0.707\lambda$, $\theta_i = 20^\circ$, $L_x = 16\lambda$, and $L_y = 16\lambda$; 256 points per square wavelength are used in the rough surface sampling.

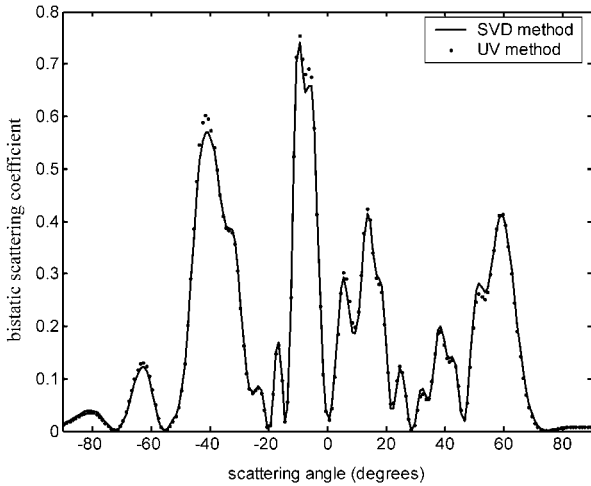


Figure 7. Comparisons of the bistatic scattering coefficients between the numerical simulations from two approaches of the SVD-based QR method and the UV method for the case of Neumann boundary condition. Here $h = 0.5 \lambda$, $l = 0.707\lambda$, $\theta_i = 20^\circ$, $L_x = 16\lambda$, and $L_y = 16\lambda$; 64 points per square wavelength are used in the rough surface sampling.

other hand, the CPU per iteration in CG of the UV method is more than the SVD's because the rank in the UV is 10–20% above the rank in the SVD. However, the total CPU using the UV is much less than the SVD's.

7.4. Comparisons of the Bistatic Scattering Coefficients Averaged Over Many Realizations

[60] In Figure 8, the bistatic scattering coefficients that averaged over 310 realizations are shown for the case of Dirichlet boundary condition using the UV method. For this case, the parameters are $h = 0.5\lambda$, $l = 0.707\lambda$, $\theta_i = 20^\circ$, $L_x = 8\lambda$ and $L_y = 8\lambda$. We also show the results of the exact solutions. It is clear that the two results are in a good agreement.

[61] In Figure 9, the bistatic scattering coefficients that averaged over 100 realizations are shown using the UV method for the case of Neumann boundary condition. The parameters are the same as Figure 7. Comparing the

Table 4. Comparison of CPU Between the UV Method and the SVD Method for the Case of Neumann Boundary Condition

Method	CG				
	Preprocessing, s	Time per Iteration, s	Number of Iterations	Time in CG, s	Total CPU, s
UV	354	8.11	19	154.1	508
SVD	26,575	3.12	19	59.3	26,634

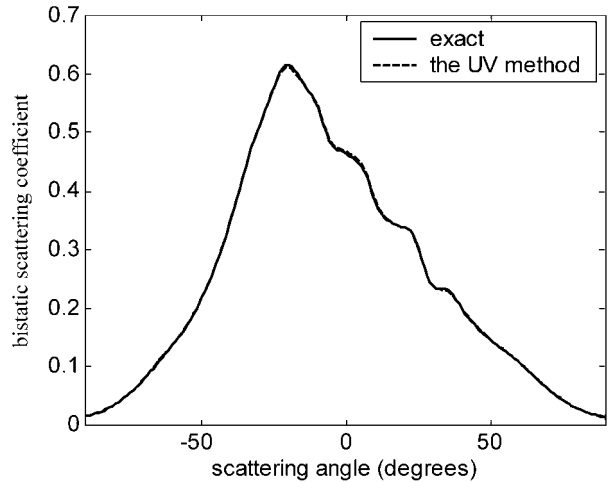


Figure 8. Comparisons of the bistatic scattering coefficients averaged over 310 realizations between the exact solution and the UV method for the case of Dirichlet boundary condition. Here $h = 0.5\lambda$, $l = 0.707\lambda$, $\theta_i = 20^\circ$, $L_x = 8\lambda$, and $L_y = 8\lambda$.

results of Figure 9 with the results of Figure 8, we see that there are more fluctuations in Figure 9 because of the smaller number of realizations used in Figure 9 than in Figure 8.

8. Conclusion

[62] In this paper, we have shown that the UV-MLP method can be used for rapid solution of integral

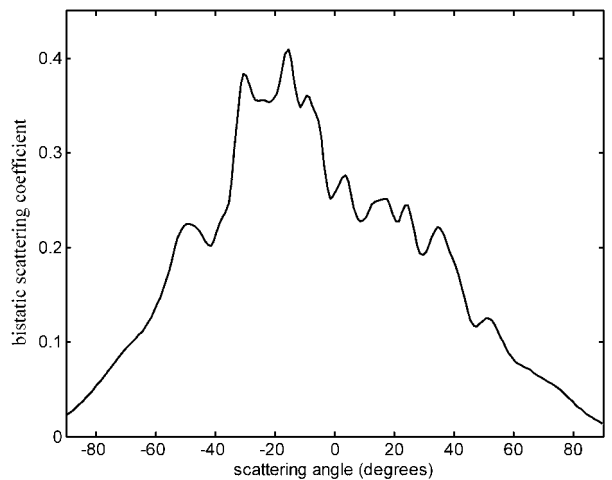


Figure 9. The bistatic scattering coefficients averaged over 100 realizations by the UV method for the case of Neumann boundary condition. Here $h = 0.5\lambda$, $l = 0.707\lambda$, $\theta_i = 20^\circ$, $L_x = 16\lambda$, and $L_y = 16\lambda$.

equation in 3-D rough surface scattering. The method can be applied to general Green's function and has been applied to volume scattering of moderate size particles using higher-order spherical waves Green's functions [Tsang and Li, 2004]. Presently we are studying the case of vector electromagnetic wave scattering by lossy dielectric random rough surfaces.

[63] **Acknowledgment.** The research in this paper was supported by the City University of Hong Kong research grant 9380034 and Hong Kong RGC Competitive Earmarked Research Grant (CERG) 9040715 and RGC central allocation grant 8730017.

References

- Jandhyala, V., E. Michielssen, S. Balasubramaniam, and W. C. Chew (1998), A combined steepest descent-fast multipole algorithm for the fast analysis of three-dimensional scattering by rough surfaces, *IEEE Trans. Geosci. Remote Sensing*, 36, 738–748.
- Kapur, S., and D. E. Long (1997), IES3: A fast integral equation solver for efficient 3-dimensional extraction, in *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, pp. 448–455, Inst. of Electr. and Electron. Eng., London.
- Maradudin, A. A., E. R. Mendez, and T. Michel (1990), Backscattering effects in the elastic scattering of p-polarized light from a large-amplitude random metallic grating, *Opt. Lett.*, 14(3), 151–153.
- Nieto-Vesperinas, M., and J. M. Soto-Crespo (1987), Monte Carlo simulations for scattering of electromagnetic waves from perfectly conducting random rough surfaces, *Opt. Lett.*, 12, 979–981.
- Pak, K., L. Tsang, C. H. Chan, and J. Johnson (1995), Backscattering enhancement of vector electromagnetic waves from two-dimensional perfectly conducting random rough surfaces based on Monte Carlo simulations, *J. Opt. Soc. Am. A*, 12(11), 2491–2499.
- Tsang, L., and Q. Li (2004), Wave scattering with UV multilevel partitioning method: Volume scattering by discrete scatterers, *Microwave Opt. Tech. Lett.*, 41(5), 354–361.
- Tsang, L., C. H. Chan, and H. Sangani (1993), Banded matrix iterative approach to Monte-Carlo simulations of scattering of waves by large-scale random rough surface problems: TM case, *Electron. Lett.*, 29(2), 166–167.
- Tsang, L., C. H. Chan, and K. Pak (1994), Backscattering enhancement of a two-dimensional random rough surface (three-dimensional scattering) based on Monte Carlo simulations, *J. Opt. Soc. Am. A*, 11(2), 711–715.
- Tsang, L., J. A. Kong, K. H. Ding, and C. O. Ao (2001), *Scattering of Electromagnetic Waves*, vol. 2, *Numerical Simulations*, Wiley-Interscience, Hoboken, N. J.
- Tsang, L., D. Chen, P. Xu, Q. Li, and V. Jandhyala (2004), Wave scattering with the UV multilevel partitioning method: 1. Two-dimensional problem of perfect electric conductor surface scattering, *Radio Sci.*, 39, doi:10.1029/2003RS003009, in press.

D. Chen, L. Tsang, and P. Xu, Department of Electrical Engineering, City University of Hong Kong, 83 Tat Chee Avenue, Kowloon, Hong Kong, China. (eeltsang@cityu.edu.hk)

V. Jandhyala and Q. Li, Department of Electrical Engineering, Box 352500, University of Washington, Seattle, WA 98195-2500, USA.