Progress In Electromagnetics Research, PIER 21, 173–201, 1999

ADAPTIVE MULTISCALE MOMENT METHOD FOR SOLVING TWO-DIMENSIONAL FREDHOLM INTEGRAL EQUATION OF THE FIRST KIND

C. Su

Dept. of Applied Math.
Northwestern Polytechnical University
Xian, Shaanxi, P.R. China

T. K. Sarkar

Dept of Electrical and Computer Engineering
Syracuse University
Syracuse, NY 13244, U.S.A

1. INTRODUCTION

Fredholm integral equation of the first kind is an important integral equation in many fields of engineering. Many methods have been proposed, such as expansion method [1–2], regularization method [3],

Backus-Gilbert method [4], Galerkin method [1] or the moment method [5] and etc. But the moment method has played an important role in engineering computations. In general, the matrix constructed by the conventional moment method is dense, the inversion of the matrix and the final solution of the linear equation is very time-consuming, particularly for a large number of subsections. But a conventional MoM matrix contains all the information required to solve scattering problems. To overcome the difficulties of large memory requirement and high computation time, many researchers have proposed the use of wavelet basis.

As we know, it is important to select a suitable basis function in numerical computation of integral equation and differential equations. For one dimensional case, many kinds of basis functions have been proposed , such as triangular basis function, pulse basis function, polynomial basis function, spline and B-spline basis function. Recently, the wavelet basis function [6] or wavelet-like basis function [7] has been proposed to solve the numerical solutions of Fredholm equations and differential equation in one dimension. Steinberg et al [8] used the wavelet expansions for the unknown current (function) in the moment method, which is expressed as a twofold summation of shifted and dilated forms of properly chosen basis function. Goswami et al [9] used wavelets on a bounded interval to solve the first-kind integral equations in electromagnetic scattering problems. Wang [10] proposed a hybrid method based on the wavelet expansion method and boundary element method. In his method, the unknown surface current is expanded in terms of a basis derived from periodic, orthogonal wavelet in interval [0,1].

Because of the local supports and vanishing moment properties of wavelets, many of the matrix elements are very small compared to the largest element, and hence can be dropped without significantly affecting the solution. Using moment method and subsequently a threshold procedure, the matrix constructed by these methods can be rendered sparse. Then, the linear equation with the sparse matrix is solved. This means that one can save CPU-time and reduce the storage requirements for the solution of the matrix equation.

The method of non-uniform grid and the multiscale technique which generates locally finer grid are usually used when the solutions of the integral equations or the differential equations have been known to vary widely in different domains. By non-uniform gridding one can reduce

the size of the problem and improve the accuracy. The multilevel or the multigrid technique has been widely used in solving the differential equations and integral equations [11–15]. Kalbasi and Demarest [16–17] applied the multilevel concepts to solve the integral equation by the moment method on different levels, which has been called the multilevel moment method. No matter what the multigrid technique is, the basis functions for an improved approximation have to be reconstructed again. By using the multiscale technique in one dimension, the basis functions for the new scale are reconstructed. From the point of view of reducing the size of linear equation, the authors proposed AMMM (adaptive multiscale moment method) [18] to study the numerical solutions of the integral equations of the first kind using the multiscale triangular basis, which is similar to wavelet basis. When using these basis to represent the known function, many of the coefficients will be zero if the known function is linear on some intervals. Therefore, we can compress the original data with the multiscale triangular basis. Many of the results [18–20] have shown that AMMM can automatically reduce the size of the linear equation constructed by the moment method with the multiscale triangular basis, and it is a very effective algorithm in solving the integral equations.

The objective of this paper is to develop AMMM for solving two-dimensional Fredholm integral equation of the first kind. First, two-dimensional multiscale basis will be introduced according to the tensor product of the one-dimensional multiscale triangular basis. We will discuss the geometrical meaning of the coefficients of the multiscale basis and compression technique for representing the original two-dimensional function on the multiscale basis. Second, by use of this kind of basis, the multiscale moment method for solving two-dimensional Fredholm integral equation of the first kind has been proposed. Furthermore, the adaptive algorithm of the multiscale moment method has been presented according to the characteristics of solution coefficients of the integral equation. Three examples for the numerical simulations carried out to test the feasibility of the multiscale moment method and its adaptive algorithm have been presented.

2. THE SIMPLEST CASE OF TWO-DIMENSIONAL MULTISCALE BASIS AND THE GEOMETRICAL MEANING OF THE COEFFICIENTS FOR MULTISCALE BASIS

Consider that for one scale on the internal [0,1], the original triangular basis and the multiscale basis have the following relationship:

$$\begin{pmatrix} \psi_1(x) \\ \psi_2(x) \\ \psi_3(x) \end{pmatrix} = \begin{pmatrix} 1 & 0 & -1/2 \\ 0 & 0 & 1 \\ 0 & 1 & -1/2 \end{pmatrix} \begin{pmatrix} \phi_1^{(0)}(x) \\ \phi_2^{(0)}(x) \\ \phi_1^{(1)}(x) \end{pmatrix} \tag{1}$$

The plots of $\phi_1^{(0)}(x), \phi_2^{(0)}(x), \phi_1^{(1)}(x), \psi_1(x), \psi_2(x), \psi_3(x)$ are shown in Fig. 1.

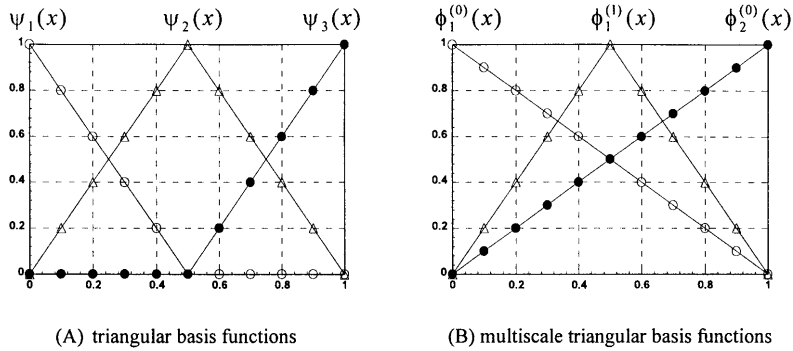


Figure 1. The triangular basis and multiscale triangular basis.

The basis functions on the region $[0,1] \times [0,1]$ are constructed by tensor product.

So the 2-D triangular basis functions are $(\psi_1(x)\psi_1(y)\,,\ \psi_2(x)\psi_1(y)\,,\ \psi_3(x)\psi_1(y)\,,\ \psi_1(x)\psi_2(y)\,,\ \psi_2(x)\psi_2(y)\,,\ \psi_3(x)\psi_2(y)\,,\ \psi_1(x)\psi_3(y)\,,\ \psi_2(x)\psi_3(y)\,,\ \psi_3(x)\psi_3(y))$. They are shown in Fig. 2(A). The 2-D multiscale triangular basis functions are $(\phi_1^{(0)}(x)\phi_1^{(0)}(y)\,,\ \phi_2^{(0)}(x)\phi_1^{(0)}(y)\,,\ \phi_1^{(0)}(x)\phi_2^{(0)}(y)\,,\ \phi_2^{(0)}(x)\phi_2^{(0)}(y)\,,\ \phi_1^{(1)}(x)\phi_1^{(0)}(y)\,,\ \phi_1^{(1)}(x)\phi_2^{(0)}(y)\,,\ \phi_1^{(0)}(x)\phi_1^{(1)}(y)\,,\ \phi_2^{(0)}(x)\phi_1^{(1)}(y)\,,\ \phi_1^{(1)}(x)\phi_1^{(1)}(y))$ and are shown in Fig. 2(B).

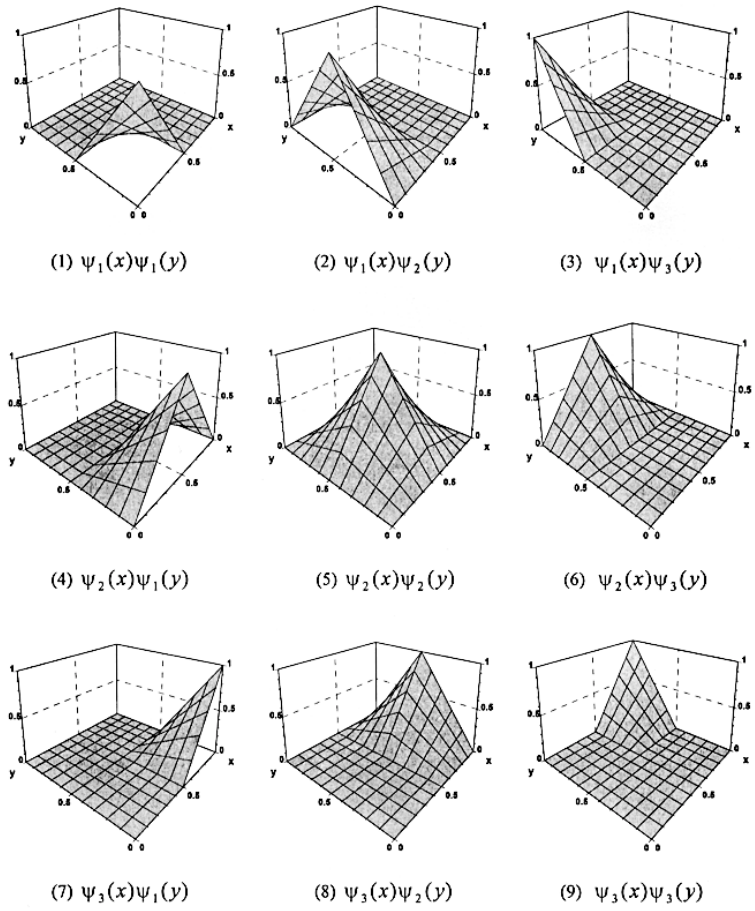


Figure 2a. 2-D triangular basis functions.

Suppose the function $f(x, y)$ is approximated in the region $[0, 1] \times [0, 1]$ with the 2-D triangular basis which is represented by

$$f_{approx}(x, y) = (\psi_1(x)\ \psi_2(x)\ \psi_3(x)) \begin{pmatrix} f_{1,1} & f_{1,2} & f_{1,3} \\ f_{2,1} & f_{2,2} & f_{2,3} \\ f_{3,1} & f_{3,2} & f_{3,3} \end{pmatrix} \begin{pmatrix} \psi_1(y) \\ \psi_2(y) \\ \psi_3(y) \end{pmatrix} \tag{2}$$

and the approximation of the function $f(x, y)$ in the region $[0, 1]\times[0, 1]$ with the 2-D multiscale triangular basis is represented by

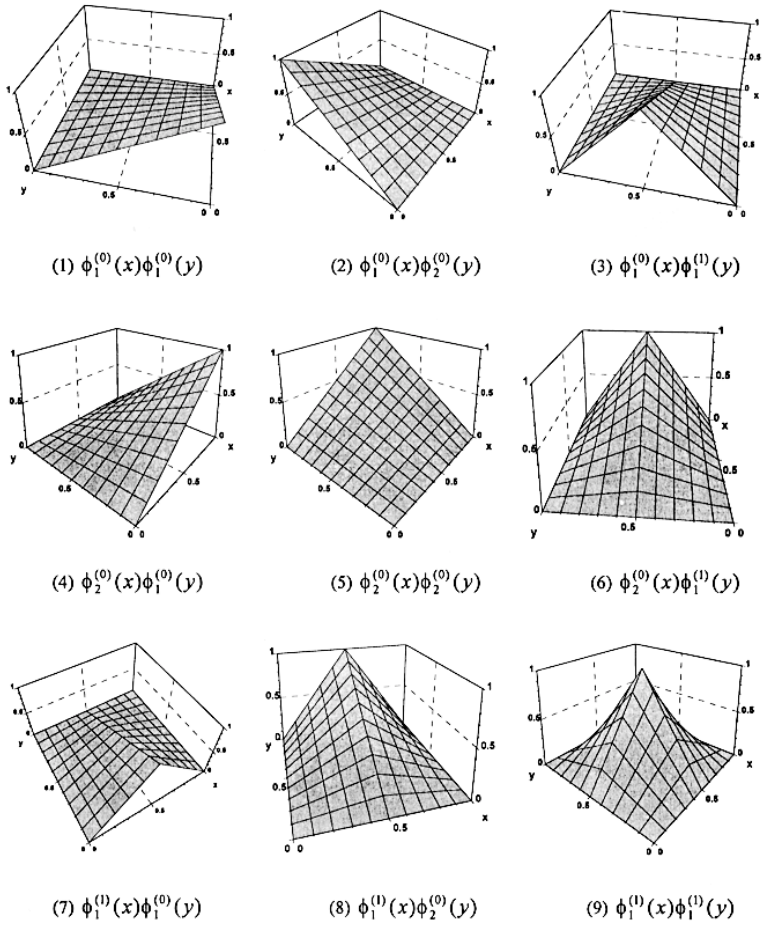


Figure 2b. 2-D multiscale triangular basis function.

$$f_{approx}(x,y) = \left(\phi_1^{(0)}(x)\ \phi_2^{(0)}(x)\ \phi_1^{(1)}(x)\right)\begin{pmatrix}\tau_{1,1} & \tau_{1,2} & \tau_{1,3}\\ \tau_{2,1} & \tau_{2,2} & \tau_{2,3}\\ \tau_{3,1} & \tau_{3,2} & \tau_{3,3}\end{pmatrix}\begin{pmatrix}\phi_1^{(0)}(y)\\ \phi_2^{(0)}(y)\\ \phi_1^{(1)}(y)\end{pmatrix}$$

$$\begin{aligned}= f_{(0)}(x,y) &+ \tau_{3,1}\phi_1^{(1)}(x)\phi_1^{(0)}(y)\\ &+ \tau_{3,2}\phi_1^{(1)}(x)\phi_2^{(0)}(y) + \tau_{1,3}\phi_1^{(0)}(x)\phi_1^{(1)}(y)\\ &+ \tau_{2,3}\phi_2^{(0)}(x)\phi_1^{(1)}(y) + \tau_{3,3}\phi_1^{(1)}(x)\phi_1^{(1)}(y)\end{aligned}$$

where

$$f_{(0)} = \tau_{1,1}\phi_1^{(0)}(x)\phi_1^{(0)}(y) + \tau_{1,2}\phi_1^{(0)}(x)\phi_2^{(0)}(y) + \tau_{2,1}\phi_2^{(0)}(x)\phi_1^{(0)}(y) + \tau_{2,2}\phi_2^{(0)}(x)\phi_2^{(0)}(y)$$

Through the relationship (1), we can obtain the following relationship between $f_{i,j}$ and $\tau_{i,j}$,

$$\begin{pmatrix} \tau_{1,1} & \tau_{1,2} & \tau_{1,3} \\ \tau_{2,1} & \tau_{2,2} & \tau_{2,3} \\ \tau_{3,1} & \tau_{3,2} & \tau_{3,3} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ -1/2 & 1 & -1/2 \end{pmatrix} \begin{pmatrix} f_{1,1} & f_{1,2} & f_{1,3} \\ f_{2,1} & f_{2,2} & f_{2,3} \\ f_{3,1} & f_{3,2} & f_{3,3} \end{pmatrix} \begin{pmatrix} 1 & 0 & -1/2 \\ 0 & 0 & 1 \\ 0 & 1 & -1/2 \end{pmatrix} \tag{3}$$

that is

$$\begin{aligned}
&\tau_{1,1} = f_{1,1}, \qquad \tau_{1,2} = f_{1,3}, \qquad \tau_{2,1} = f_{3,1}, \qquad \tau_{2,2} = f_{3,3} \\
&\tau_{1,3} = f_{1,2} - \frac{1}{2}(f_{1,1} + f_{1,3}), \qquad \tau_{2,3} = f_{3,2} - \frac{1}{2}(f_{3,1} + f_{3,3}) \\
&\tau_{3,1} = f_{2,1} - \frac{1}{2}(f_{1,1} + f_{3,1}), \qquad \tau_{3,2} = f_{2,3} - \frac{1}{2}(f_{1,3} + f_{3,3}) \\
&\tau_{3,3} = f_{2,2} - \frac{1}{2}(f_{1,2} + f_{3,2}) \\
&\qquad - \frac{1}{2}(f_{2,1} + f_{2,3}) + \frac{1}{4}(f_{1,1} + f_{1,3} + f_{3,1} + f_{3,3}) \\
&\quad = f_{2,2} - \frac{1}{2}(\tau_{1,3} + \tau_{3,1} + \tau_{2,3} + \tau_{3,2}) - \frac{1}{4}(f_{1,1} + f_{1,3} + f_{3,1} + f_{3,3}) \\
&\quad = \tau'_{3,3} - \frac{1}{2}(\tau_{1,3} + \tau_{3,1} + \tau_{2,3} + \tau_{3,2})
\end{aligned}$$

where $\tau'_{3,3} = f_{2,2} - \frac{1}{4}(f_{1,1} + f_{1,3} + f_{3,1} + f_{3,3})$. Geometrically, $\tau_{i,j}$ implies the difference between the approximate linear function in $[0,1]$ and the original function $f(x,y)$ at the middle point (x_i, y_j) (see Fig. 3).

If the function $f(x,y)$ has the two-order continuous differentiability on the region, then

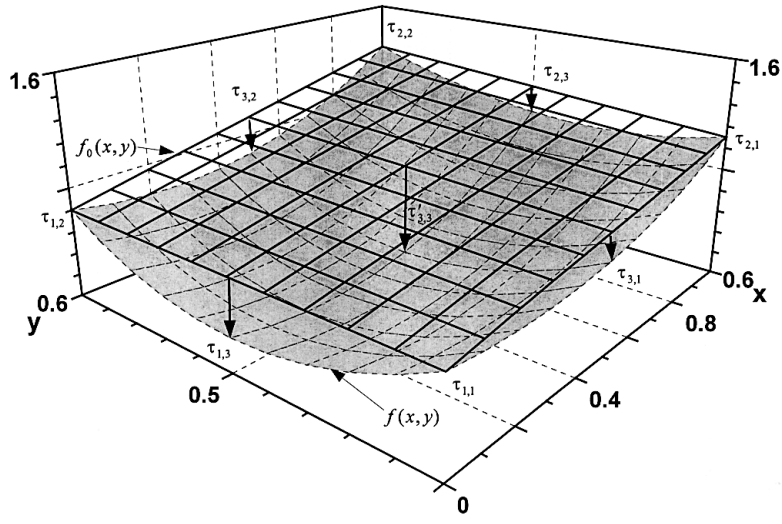


Figure 3. Geometrical meaning of the coefficients $\tau_{i,j}$.

$$\tau_{1,3} \approx -\frac{h^2}{2}\frac{\partial^2 f(0,\frac{1}{2})}{\partial y^2}, \qquad \tau_{2,3} \approx -\frac{h^2}{2}\frac{\partial^2 f(1,\frac{1}{2})}{\partial y^2}, \qquad h = 1/2$$

$$\tau_{3,1} \approx -\frac{h^2}{2}\frac{\partial^2 f(\frac{1}{2},0)}{\partial x^2}, \qquad \tau_{2,3} \approx -\frac{h^2}{2}\frac{\partial^2 f(\frac{1}{2},1)}{\partial x^2},$$

$$\tau'_{3,3} \approx -\frac{h^2}{4}\left(\frac{\partial^2 f(\frac{1}{2},\frac{1}{2})}{\partial x^2} + \frac{\partial^2 f(\frac{1}{2},\frac{1}{2})}{\partial y^2}\right)$$

$$\tau_{3,3} \approx \frac{h^2}{4}\left(\frac{\partial^2 f(\frac{1}{2},0)}{\partial x^2} + \frac{\partial^2 f(\frac{1}{2},1)}{\partial x^2} + \frac{\partial^2 f(\frac{1}{2},0)}{\partial y^2} + \frac{\partial^2 f(\frac{1}{2},1)}{\partial y^2} - \frac{\partial^2 f(\frac{1}{2},\frac{1}{2})}{\partial x^2} - \frac{\partial^2 f(\frac{1}{2},\frac{1}{2})}{\partial y^2}\right)$$

Therefore, if $f(x,y)$ is a planar function $(f(x,y) = ax + by + c)$ on the region $[0,1]\times[1,0]$, the coefficients $\tau_{1,3}$, $\tau_{2,3}$, $\tau_{3,1}$, $\tau_{3,2}$, $\tau_{3,3}$ will be zero.

3. TWO-DIMENSIONAL MULTISCALE BASIS AND FORMULA FOR APPROXIMATE FUNCTIONS BASED ON A MULTISCALE TECHNIQUETable

As we know, the multiscale basis functions in AMMM are based on the use of uniform grid and triangular basis functions. They can be obtained from the triangular basis by use of a full-rank matrix transformation, that is,

$$\Psi_{\Pi}^{N,V}(x) = T(N,V)\Phi_{\Delta}^{1+2^V N}(x) \tag{4}$$

where $\Phi_{\Delta}^{1+2^V N}(x) = (\phi_1(x), \cdots, \phi_{1+2^V N}(x))^T$ is the uniform triangular basis. $\Psi_{\Pi}^{N,V}(x) = (\psi_1(x), \cdots, \psi_{1+2^V N}(x))^T$ is the multiscale basis functions. N is the number of the initial division of the interval. V is the number of the largest scale, and the arc is divided $1 + 2^V N$ into points. $T(N,V)$ is a full-rank matrix. The coefficient vector F_M of $f(x)$ on the multiscale basis and coefficient vector F_Δ of $f(x)$ on the uniform triangular basis have the following relationship:

$$F_\Delta = T'(N,V)F_M \tag{5}$$

That means the coefficient vector F_Δ can be obtained from the coefficient vector F_M through a matrix transformation and vice versa. For the two-dimensional case, the basis functions can be constructed by tensor product.

Suppose the multiscale basis function and triangular basis function for the x- directed component are $\Psi_{\Pi}^{N,V}(x) = (\psi_1(x), \cdots, \psi_{1+2^V N}(x))^T$, $\Phi_{\Delta}^{N,V}(x) = (\phi_1(x), \cdots, \phi_{1+2^V N}(x))^T$ respectively. The multiscale basis function and triangular basis function for y-directed component of the solution are $\Psi_{\Pi}^{M,V}(y) = (\psi_1(y), \cdots, \psi_{1+2^V M}(y))^T$, $\Phi_{\Delta}^{M,V}(y) = (\phi_1(y), \cdots, \phi_{1+2^V M}(y))^T$, respectively. The basis functions in two-dimension can be written in the form of a tensor product:

$$\Phi_{\Delta}^{1+2^V N}(x) \otimes \Phi_{\Delta}^{1+2^V M}(y) = \begin{pmatrix} \phi_1(x) \\ \vdots \\ \phi_{1+2^V N}(x) \end{pmatrix} (\phi_1(y) \ \cdots \ \phi_{1+2^V M}(y))$$

for triangular form

$$\Psi_{\Pi}^{N,V}(x) \otimes \Psi_{\Pi}^{M,V}(y) = \begin{pmatrix} \psi_1(x) \\ \vdots \\ \psi_{1+2^V N}(x) \end{pmatrix} (\psi_1(y) \ \cdots \ \psi_{1+2^V M}(y))$$

for multiscale form

The relation between multiscale basis and the triangular basis is

$$\Psi_{\Pi}^{N,V}(x) \otimes \Psi_{\Pi}^{M,V}(y) = T(N,V) \begin{pmatrix} \phi_1(x) \\ \vdots \\ \phi_{1+2^V N}(x) \end{pmatrix} (\phi_1(y) \ \cdots \ \phi_{1+2^V M}(y))\, T'(M,V)$$

$$\Psi_{\Pi}^{N,V}(x) \otimes \Psi_{\Pi}^{M,V}(y) = T(N,V)\Phi_{\Delta}^{1+2^V N}(x) \otimes \Phi_{\Delta}^{1+2^V M}(y)T'(M,V) \tag{6}$$

The function $f(x,y)$ can be expressed by the basis functions in the following form

$$\begin{aligned} f(x,y) &= \sum_{i,j} f_{i,j}^{\Delta}\phi_i(x)\phi_j(y) = \left(\Phi_{\Delta}^{1+2^V N}(x)\right)' F^{\Delta}\Phi_{\Delta}^{1+2^V M}(y) \\ &= \sum_{i,j} f_{i,j}^{\Pi}\psi_i(x)\psi_j(y) = \left(\Psi_{\Pi}^{N,V}(x)\right)' F^{\Pi}\Psi_{\Pi}^{M,V}(y) \end{aligned} \tag{7}$$

F^{Π} and F^{Δ} have the following relationship:

$$F^{\Pi} = T'(N,V)F^{\Delta}T(M,V) \tag{8}$$

The approximate function of $f(x,y)$ on the two-dimensional triangular basis functions can be represented by

$$f_{approx}(x,y) = (\phi_1(x), \cdots, \phi_{1+2^V N}(x)) \begin{pmatrix} f(x_1,y_1) & \dots & f(x_1,y_{1+2^V M}) \\ \vdots & \ddots & \vdots \\ f(x_{1+2^V N},y_1) & \dots & f(x_{1+2^V N},y_{1+2^V M}) \end{pmatrix} \begin{pmatrix} \phi_1(y) \\ \vdots \\ \phi_{1+2^V M}(y) \end{pmatrix}$$

The approximate function of $f(x,y)$ can be represented by multiscale basis

$$f_{approx}(x,y) = \left(\Psi_{\Pi}^{N,V}(x)\right)' T'(N,V) \begin{pmatrix} f(x_1,y_1) & \dots & f(x_1,y_{1+2^V M}) \\ \vdots & \ddots & \vdots \\ f(x_{1+2^V N},y_1) & \dots & f(x_{1+2^V N},y_{1+2^V M}) \end{pmatrix} T(M,V)\Psi_{\Pi}^{M,V}(y)$$

The image of the function $f(x,y)$ on the two-dimensional triangular basis functions is the following matrix

$$\begin{pmatrix} f(x_1,y_1) & \cdots & f(x_1,y_{1+2^V M}) \\ \vdots & \ddots & \vdots \\ f(x_{1+2^V N},y_1) & \cdots & f(x_{1+2^V N},y_{1+2^V M}) \end{pmatrix} = \begin{pmatrix} f_{1,1} & \cdots & f_{1,1+2^V M} \\ \vdots & \ddots & \vdots \\ f_{1+2^V N,1} & \cdots & f_{1+2^V N,1+2^V M} \end{pmatrix} \tag{9}$$

The image of $f(x,y)$ on the two-dimensional multiscale basis is the following matrix

$$\begin{pmatrix} f^{\Pi}_{1,1} & \cdots & f^{\Pi}_{1,1+2^V M} \\ \vdots & \ddots & \vdots \\ f^{\Pi}_{1+2^V N,1} & \cdots & f^{\Pi}_{1+2^V N,1+2^V M} \end{pmatrix} = T'(N,V)\begin{pmatrix} f(x_1,y_1) & \cdots & f(x_1,y_{1+2^V M}) \\ \vdots & \ddots & \vdots \\ f(x_{1+2^V N},y_1) & \cdots & f(x_{1+2^V N},y_{1+2^V M}) \end{pmatrix} T(M,V) \tag{10}$$

After filtering of some relatively small elements, we can get the new image

$$\begin{pmatrix} f^{C}_{1,1} & \cdots & f^{C}_{1,1+2^V N} \\ \vdots & \ddots & \vdots \\ f^{C}_{1+2^V N,1} & \cdots & f^{C}_{1+2^V N,1+2^V N} \end{pmatrix}$$

where

$$f^{C}_{i,j} = \begin{cases} f^{\Pi}_{i,j} & \text{if } |f^{\Pi}_{i,j}| > \epsilon \\ 0 & \text{if } |f^{\Pi}_{i,j}| \le \epsilon \end{cases}$$

The approximate function of $f(x,y)$ can be obtained after filtering some relatively small elements as

$$f^{C}_{approx}(x,y) = \left(\Psi^{N,V}_{\Pi}(x)\right)' \begin{pmatrix} f^{C}_{1,1} & \cdots & f^{C}_{1,1+2^V M} \\ \vdots & \ddots & \vdots \\ f^{C}_{1+2^V N,1} & \cdots & f^{C}_{1+2^V N,1+2^V M} \end{pmatrix} \Psi^{M,V}_{\Pi}(y) \tag{11}$$

Compression ratio is defined as the number of elements $f^C_{i,j}$ which is zero over the total number of elements $(1+2^V N)^2$. The average error and the maximum error between $\{f_{approx}(x_i, y_j)\}$ and $\{f^C_{approx}(x_i, y_j)\}$ are defined as follows:

$$AverErr(N,V) = \sqrt{\frac{\sum_i \sum_j \left(f_{approx}(x_i,y_j) - f^C_{approx}(x_i,y_j)\right)^2}{\sum_i \sum_j \left(f_{approx}(x_i,y_j)\right)^2}}$$

$$MaxErr(N,V) = \max\left|f_{approx}(x_i,y_j) - f^C_{approx}(x_i,y_j)\right|$$

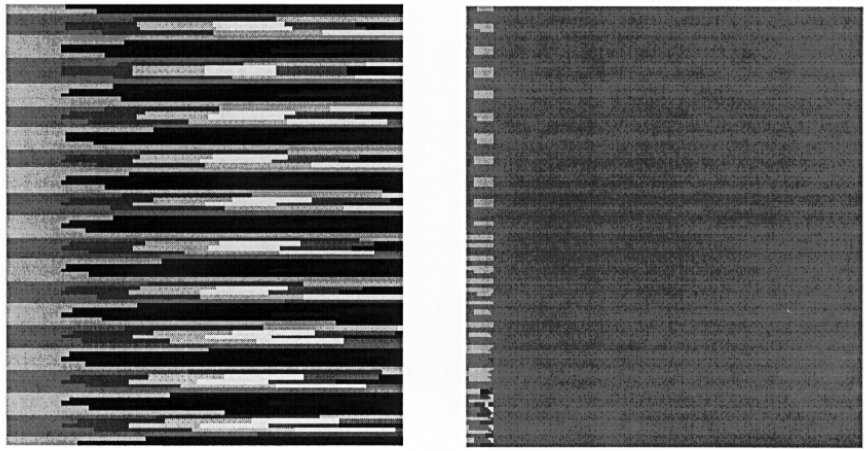
In the following, we give some examples of two-dimensional functions $f(x,y)$ and the compressed image F^{Π} using the multiscale basis.

Example one: The original function $f(x,y)$ is $(10\pi x+1)(\cos 20\pi y)+1$. Suppose the threshold is $\epsilon = 0.01$. The largest scale is $V = 4$. The initial division of the interval is $N = M = 16$. Total number of the elements is 66049. The original image $\{f_{i,j}\}$ of the function $f(x,y)$ and the image of $\{f^C_{i,j}\}$ are plotted in Fig. 4(a) and Fig. 4(b), respectively. The number of zero and non-zero elements of $\{f^C_{i,j}\}$ is 61756 and 4293, respectively.

Example two: The original function $f(x,y)$ is defined as $\sin(10\pi(x-0.5)^2 + 20\pi(y-0.5)^2) + 1$. Suppose the threshold is $\epsilon = 0.01$. The largest scale is $V = 4$. The initial division of the interval is $N = M = 16$. The original image of $\{f_{i,j}\}$ the function $f(x,y)$ and the image of $\{f^C_{i,j}\}$ are plotted in Fig. 5(a) and Fig. 5(b), respectively. The number of zero and non-zero elements of $\{f^C_{i,j}\}$ is 62152 and 3897, respectively.
Example three: The original function $f(x,y)$ is defined as follows

$$f(x,y) = \begin{cases} 20\left((x-0.5)^2 + (y-0.5)^2\right) & (x-0.5)^2 + (y-0.5)^2 \le 0.3^2 \\ 3x - 2y + 1 & (x-0.5)^2 + (y-0.5)^2 > 0.3^2 \end{cases}$$

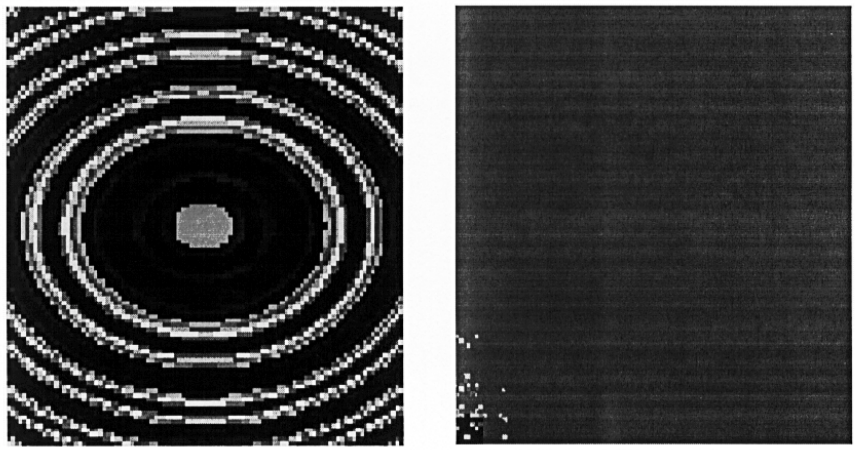
Suppose the threshold is $\epsilon = 0.01$. The largest scale is $V = 4$. The initial division of the interval is $N = M = 16$. The original image $\{f_{i,j}\}$ of the function $f(x,y)$ and the image of $\{f^C_{i,j}\}$ are plotted in Fig. 6(a) and Fig. 6(b), respectively. The number of zero and non-zero elements of $\{f^C_{i,j}\}$ is 65124 and 925, respectively.



(a) The image of $\{f_{i,j}\}$ (b) The image of $\{f^C_{i,j}\}$

Compression ratio =93.5%, AverErr = 4.7E-5, MaxErr=1.22E-2

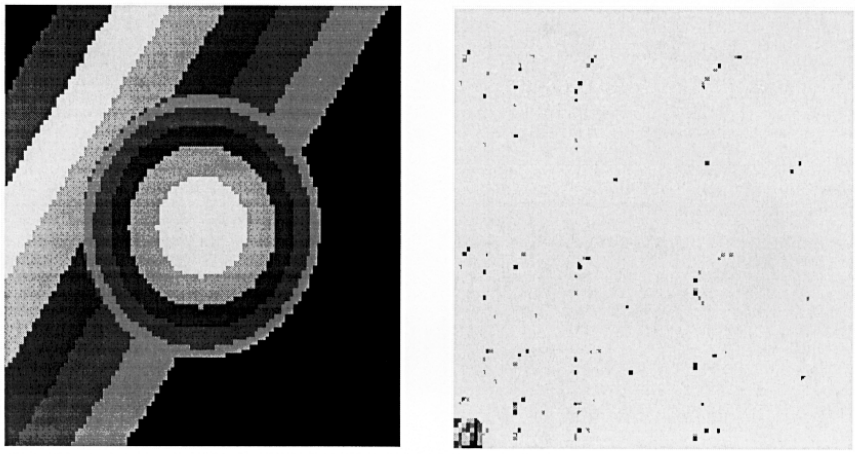
Figure 4(a-b). The image of the original function $f(x, y)$ and the function $f^C_{approx}(x, y)$ on the two-dimensional multiscale basis.



(a) The image of $\{f_{i,j}\}$ (b) The image of $\{f^C_{i,j}\}$

Compression ratio =94.1%, AverErr = 6.49E-3, MaxErr=2.83E-2

Figure 5(a-b). The image of the original function $f(x, y)$ and the function $f^C_{approx}(x, y)$ on the two-dimensional multiscale basis.



(a) The image of $\{f_{i,j}\}$ (b) The image of $\{f^C_{i,j}\}$

Compression ratio =98.6%, AverErr = 3.09E-4, MaxErr=7.78E-3

Figure 6(a-b). The image of the original function $f(x,y)$ and the function $f^C_{approx}(x,y)$ on the two-dimensional multiscale basis.

4.A MULTISCALE MOMENT METHOD FOR SOLVING 2D FREDHOLM INTEGRAL EQUATION OF THE FIRST KIND

For simplicity, We consider the following Fredholm integral equation in two dimension

$$\int_0^1 dy' \int_0^1 k(x,y;x',y')f(x',y')dx' = g(x,y) \qquad (x,y) \in [0,1] \times [1,0] \tag{12}$$

we choose the triangular basis functions on the uniform grid with the node $\{x_m = y_m = mh, m = 0,1,2,\ldots,2^V N, h = \frac{1}{2^V N}\}$ for x and y axis. Suppose the unknown function $f(x,y)$ be represented by

$$f(x,y) = \sum_{i=0}^{1+2^V N} \sum_{j=0}^{1+2^V N} f(x_i,y_j)\phi_i(x)\phi_j(y) = \Phi'_\Delta F_\Delta \tag{13}$$

where

$$F_\Delta = (f(x_0,y_0),\ldots,f(x_{1+2^V N},y_0),\ldots$$
$$\ldots,f(x_0,y_{1+2^V N}),\ldots,f(x_{1+2^V N},y_{1+2^V N}))'$$

Discritizing the integral equation by use of the point-matching method and triangular basis functions in two dimensions, we obtain the following matrix equation

$$A_\Delta F_\Delta = G_\Delta \tag{14}$$

where

$$G_\Delta = (g(x_0, y_0), \ldots, g(x_{1+2^V N}, y_0), \ldots$$
$$\ldots, g(x_0, y_{1+2^V N}), \ldots, g(x_{1+2^V N}, y_{1+2^V N}))'$$

$$A_\Delta = (a_{i,j})_{(1+2^V N)^2 \times (1+2^V N)^2}$$

$$a_{0,0} = \iint K(x_0, y_0, x', y')\phi_0(x')\phi_0(y')dx'dy'$$

$$a_{0,1} = \iint K(x_0, y_0, x', y')\phi_1(x')\phi_0(y')dx'dy'$$

$$a_{0,1+2^V N} = \iint K(x_0, y_0, x', y')\phi_{1+2^V N}(x')\phi_0(y')dx'dy'$$

$$a_{0,(1+2^V N)\times(1+2^V N)} = \iint K(x_0, y_0, x', y')\phi_{1+2^V N}(x')\phi_{1+2^V N}(y')dx'dy'$$

$$a_{1,0} = \iint K(x_1, y_0, x', y')\phi_0(x')\phi_0(y')dx'dy'$$

$$a_{1,1} = \iint K(x_1, y_0, x', y')\phi_1(x')\phi_0(y')dx'dy'$$

$$a_{1,1+2^V N} = \iint K(x_1, y_0, x', y')\phi_{1+2^V N}(x')\phi_0(y')dx'dy'$$

$$a_{1,(1+2^V N)\times(1+2^V N)} = \iint K(x_1, y_0, x', y')\phi_{1+2^V N}(x')\phi_{1+2^V N}(y')dx'dy'$$

$$a_{1+2^V N,0} = \iint K(x_{1+2^V N}, y_0, x', y')\phi_0(x')\phi_0(y')dx'dy'$$

$$a_{1+2^V N,1} = \iint K(x_{1+2^V N}, y_0, x', y')\phi_1(x')\phi_0(y')dx'dy'$$

$$a_{1+2^V N,1+2^V N} = \iint K(x_{1+2^V N}, y_0, x', y')\phi_{1+2^V N}(x')\phi_0(y')dx'dy'$$

$$a_{1+2^V N,(1+2^V N)\times(1+2^V N)} = \iint K(x_{1+2^V N}, y_0, x', y')\phi_{1+2^V N}(x')\phi_{1+2^V N}(y')dx'dy'$$

$$a_{(1+2^V N)\times(1+2^V N),0} = \iint K(x_{1+2^V N}, y_{1+2^V N}, x', y')\phi_0(x')\phi_0(y')dx'dy'$$

$$a_{(1+2^V N)\times(1+2^V N),1} = \iint K(x_{1+2^V N}, y_{1+2^V N}, x', y')\phi_1(x')\phi_0(y')dx'dy'$$

$$a_{(1+2^V N)\times(1+2^V N),1+2^V N} = \iint K(x_{1+2^V N}, y_{1+2^V N}, x', y')\phi_{1+2^V N}(x')\phi_0(y')dx'dy'$$

$$a_{(1+2^V N)\times(1+2^V N),(1+2^V N)\times(1+2^V N)} = \iint K(x_{1+2^V N}, y_{1+2^V N}, x', y')\phi_{1+2^V N}(x')\phi_{1+2^V N}(y')dx'dy'$$

Now, we consider the form of the above matrix equation in the two-dimensional multiscale basis. Suppose the two-dimensional triangular basis $\vec{\Phi}_\Delta(x, y)$ be arranged as follows:

$$\vec{\Phi}_\Delta(x, y) = \begin{pmatrix} \phi_1(y)\phi_1(x) \\ \vdots \\ \phi_1(y)\phi_{1+2^V N}(x) \\ \vdots \\ \vdots \\ \phi_{1+2^V N}(y)\phi_1(x) \\ \vdots \\ \phi_{1+2^V N}(y)\phi_{1+2^V N}(x) \end{pmatrix}$$

and the two-dimensional multiscale basis $\vec{\Psi}^V_\Pi(x, y)$ be arranged as fol-

lows

$$\begin{aligned}\vec{\Psi}_{\Pi} \equiv \Big(&\vec{\Psi}_0(y)\otimes\vec{\Psi}_0(x), \vec{\Psi}_0(y)\otimes\vec{\Psi}_1(x), \vec{\Psi}_1(y)\otimes\vec{\Psi}_0(x),\\ &\vec{\Psi}_1(y)\otimes\vec{\Psi}_1(x), \vec{\Psi}_0(y)\otimes\vec{\Psi}_2(x), \vec{\Psi}_2(y)\otimes\vec{\Psi}_0(x),\\ &\vec{\Psi}_1(y)\otimes\vec{\Psi}_2(x), \vec{\Psi}_2(y)\otimes\vec{\Psi}_1(x), \vec{\Psi}_2(y)\otimes\vec{\Psi}_2(x), \ldots\ldots,\\ &\vec{\Psi}_0(y)\otimes\vec{\Psi}_V(x), \vec{\Psi}_V(y)\otimes\vec{\Psi}_0(x), \vec{\Psi}_1(y)\otimes\vec{\Psi}_V(x), \vec{\Psi}_V(y)\otimes\vec{\Psi}_1(x),\\ &\ldots\ldots, \vec{\Psi}_V(y)\otimes\vec{\Psi}_V(x), \vec{\Psi}_V(y)\otimes\vec{\Psi}_V(x)\Big)^T\end{aligned}$$

where $\vec{\Psi}_V(x)$ and $\vec{\Psi}_V(y)$ is the basis functions on v th scale along the x axis and y axis, respectively.

Therefore, from 0-scale to 5-scale, the number of multiscale basis functions is $(1+N)^2, 3N^2+2N, 12N^2+4N, 48N^2+8N, 192N^2+16N, 768N^2+32N$ respectively.

From equation (6), we can evaluate the transformation matrix $W(N,V)$ from the basis functions $\vec{\Phi}_{\Delta}(x,y)$ to the multiscale basis function in two dimension $\vec{\Psi}^V_{\Pi}(x,y)$, that is

$$\vec{\Psi}^V_{\Pi}(x,y) = W(N,V)\vec{\Phi}_{\Delta}(x,y) \tag{15}$$

The matrix equation (13) can be written in the form of the multiscale basis

$$A_M F_M = G_M \tag{16}$$

where $F_{\Delta} = W'(N,V)F_M$, $G_M = W(N,V)G_{\Delta}$, $A_M = W(N,V)A_{\Delta}$ $W'(N,V)$. The coefficient matrix A_M, the unknown F_M and the array F_M are arranged as the scaled-block form. (see Fig. 7)

This scaled-block form in two-dimensional integral equation is similar to the scaled-block form in one-dimensional case. The only different place is that the size of the block matrix of the next scale is not the same as of the matrix of the previous scale.

5. AN ADAPTIVE ALGORITHM OF MULTISCALE MOMENT METHOD

In the discussion of the Section (3), it is shown that if the solution function $f(x,y)$ of the integral equation is linear in some region, many of the coefficients $\{f^{\Pi}_{i,j}\}$ (that is, $\{F^M(v)\}$) will be zero. So we can

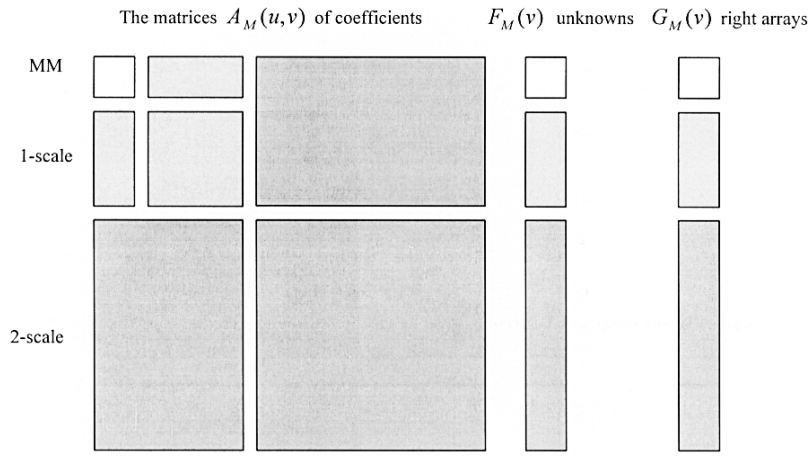


Figure 7. The illustration of coefficient matrices, the right arrays and the unknowns

reduce the size of the linear equation in the moment method. For $(v+1)$ th scale, the size of the linear equation formed by the moment method will be reduced according to the solution $\{F^M(0), F^M(1), \ldots, F^M(v)\}$ of the previous scale and the predicted initial guess $\{F^M(v+1)\}$, as some of them approach zero. This method is called as the adaptive multiscale moment method. The scheme of the adaptive multiscale moment method to solve the two- dimensional Fredholm integral equation of the first kind from the (v) th scale to the $(v+1)$ th scale is given as follows:

Step 1: start with the original solution $\{F_\Delta(v)\}$ on the coarse grid by the use of the following formula

$$F_\Delta = T'(N, v) \begin{pmatrix} F_M(0) \\ F_M(1) \\ \vdots \\ F_M(v) \end{pmatrix}$$

Step 2: estimate the solution $\{F_\Delta(v+1)\}$ on the thinner grid by use of the two-dimensional interpolant formula, such as a tensor product spline interpolant.

Step 3: get the initial guess for the solution $\{F^M(0), F^M(1), \ldots, F^M(v)\}$

on the thinner grid by use of the following formula

$$\begin{pmatrix} \hat{F}_M(0) \\ \hat{F}_M(1) \\ \vdots \\ \hat{F}_M(v+1) \end{pmatrix} = \left(T'(N, v+1)\right)^{-1} F_\Delta(v+1)$$

Step 4: If the elements of $\{\hat{F}_M(k)\}(k = 1, 2, 3, \ldots, v+1)$ are less than ϵ (the given threshold parameter), we set these elements to zero and delete the corresponding row and column of the coefficient matrix A_M and the corresponding left element of the array F_M on the $(v+1)$ th scale. Then after reducing the size of the original linear equation, the modified linear equation and the initial guess are obtained.
Step 5: solve the modified linear equation by use of the iterative method or LU method. Adding these elements which are set to be zero in Step 4, the solution $\{F^M(0), F^M(1), \ldots, F^M(v)\}$ on the multiscale triangular basis along the thinner grid are obtained. Using the step one, we can obtain the original solution on the sparse grid. This procedure continue until the largest scale is reached.

In the numerical computation of interpolation to estimate the coefficients, we adapt the tensor product spline interpolant. The tensor product spline interpolant function to data $\{f(x_i, y_j)\}$, where $\{1 \le i \le N_x\}$ and $\{1 \le j \le N_j\}$, has the form

$$\sum_{m=1}^{N_y} \sum_{n=1}^{N_x} c_{nm} B_{n,k_x,t_x}(x) B_{m,k_y,t_y}(y)$$

where $B_{i,k,t}(s)$ is v-th (normalized) B-spline of order k for the knot sequence t. The coefficients $c_{n,m}$ are necessary to solve the system of equations

$$\sum_{m=1}^{N_y} \sum_{n=1}^{N_x} c_{nm} B_{n,k_x,t_x}(x_i) B_{m,k_y,t_y}(y_j) = f(x_i, y_j)$$
$$\{1 \le i \le N_x, 1 \le j \le N_y\}$$

This problem can be computed quite efficiently by repeatedly solving two univariate interpolation problems as described in de Boor [21] (1978, p.347).

Some notes:
(1) This new method is different from the method used to the 2-D integral equations by use of the multilevel moment method. There are four differences between the multilevel moment method and the adaptive multiscale moment method (abbreviated as AMMM). First, both of the methods are different in order to form the matrix of the linear equation. For the multilevel moment method, the matrix needs to be computed for all of levels. For AMMM, the new matrix needs to be computed once and it is desired to increase the scale. Second, although the initial guess of adding terms is determined by interpolating method, the meaning of the data are different. For the multilevel moment method, the estimated initial guesses are the approximated value of the solution $f(x, y)$ on the fine level. For AMMM, the estimated initial guesses are the value relate to the function $f''_{xx}(x, y)$, $f''_{yy}(x, y)$ and the scale V. If the solution function is almost linear, most of unknowns coefficients will be zero, which can help us to omit these terms and reduce the size of the linear equation. Third, from the coarse grid to the fine grid, all of the basis functions need to be constructed again for the multilevel moment method. For AMMM, the basis functions on the newly added nodes need to be constructed, which are the same as the basis functions of the multilevel moment method. Fourth, the multilevel moment method can not reduce the size of the linear equation. For different levels, there are the fixed order number of linear equations no matter how the solution function is. For AMMM, the size of the linear equation can be reduced according to the characteristics of the solution function and the suitable threshold. So the new method is an adaptive algorithm. If the threshold is taken to be zero, the size of the linear equation will not be reduced.
(2) The AMMM is almost like the wavelet method for solving the Fredholm equation. But there are also some differences between the two methods. For the wavelet moment method, the basis functions are constructed by shifted and dilated forms of the mother wavelet, which has vanishing moment properties. Many of the matrix elements are very small compared to the largest elements and can be dropped without significantly affecting the solution. The matrix is rendered sparsely populated. So it can improve computational efficiency. But it is difficult (almost impossible) to solve the integral equation over an arbitrary domain in two or three dimensions by use of the wavelet moment method because it is difficult to find the wavelet on an arbitrary

domain.

For 2-D AMMM, the basis functions are also constructed by a tensor product of 1-D multiscale triangular basis which is shifted and dilated forms of a function which has no vanishing moment properties. The matrix formed by the moment method on the different scales will not be sparse. 2-D AMMM can improve the computational efficiency from reducing the size of the linear equation on the basis of the characteristics of the solution function.

(3) From the formula (16), it is seen that 2-D AMMM is very similar to 1-D AMMM. The only difference is that the transformation matrices are constructed in different ways. So 2-D AMMM is the extended version of 1-D AMMM.

6. NUMERICAL RESULTS

In order to test the feasibility of the adaptive multiscale moment method for solving two-dimensional Fredholm integral equation of the first kind, two kinds of kernel functions of the integral are considered. All of numerical simulations have been performed on DELL OptiPlex Gxi 166 MHz personal computer in a multi-tasking environment. The basic method of solving the linear equation is chosen as the Gauss method.

Example one: consider the following kernel function

$$k(x, y, x', y') = -\ln\sqrt{(x - x')^2 + (y - y')^2}$$

The region $[0, 1] \times [0, 1]$ is uniformly subdivided into $N = N_x \times N_y = 33 \times 33$. The source function $g(x, y) = 1$. The error between the solution functions $f_\Delta(x_i, y_i)$ and $f_\Pi(x_i, y_i)$ by use of the conventional moment method and the adaptive multiscale moment method is defined as

$$Err(f_\Delta, f_\Pi) = \sqrt{\frac{\sum\limits_i \sum\limits_j \left(f_\Delta(x_i, y_j) - f_\Pi(x_i, y_j)\right)^2}{\sum\limits_i \sum\limits_j \left(f_\Delta(x_i, y_j)\right)^2}}$$

For the different thresholds, the actual size and the condition number of the linear equation on the largest scale $J = 3$, the error $Err(f_\Delta, f_\Pi)$ and the CPU time are given in TABLE (1).

threshold	size	CN	Err	CPU (sec)
0.0	1089	8763167	2.676E-5	56.46
0.001	677	5242464	0.1665	40.54
0.01	303	1973944	0.4810	27.02
0.1	73	142653	0.7277	25.38
MM	1089	6294	0.	56.19

Table I. Error and CPU time for example one.

where CPU time is the time required to solve the linear equation by use of the conventional moment method or the adaptive multiscale moment method. This does not include the time spent to compute the coefficient matrix and the source term. CN is the condition number.

The solution functions $f_\Delta(x_i, y_j)$ and $f_\Pi(x_i, y_j)$ for the different threshold on the largest scale $V = 3$ are plotted in Fig. 8.

From the computed results, it is shown that the AMMM can reproduce well the singularity of the function at the four corners although it can reduce about 38% , 72% , 93% the size of the linear equation for different thresholds, respectively.

Example two: consider the same kernel function as in example one. The solution is defined as

$$f(x,y) = \begin{cases} 20\left((x-0.5)^2 + (y-0.5)^2\right) & (x-0.5)^2 + (y-0.5)^2 \le 0.3^2 \\ 3x - 2y + 1 & (x-0.5)^2 + (y-0.5)^2 > 0.3^2 \end{cases}$$

The number of the totally uniform division on $[0,1] \times [0,1]$ is $N = N_x \times N_y = 65 \times 65$. Total number of unknowns are 4225.

For the different scales, the actual size and the condition number of the linear equation of the threshold parameter $\epsilon = 0.01$, the error $Err(f_\Delta, f_\Pi)$ and CPU time are given in TABLE (2).

size V	size	CN	Err	CPU (sec)
0 (MM)	4225	24490	0	4456.98
1	2086	177237	0.015867	710.29
2	1377	416272	0.029397	425.73
3	1235	2329504	0.045076	455.88
4	1351	15130424	0.051114	606.82

Table 2. Error and CPU time for example two.

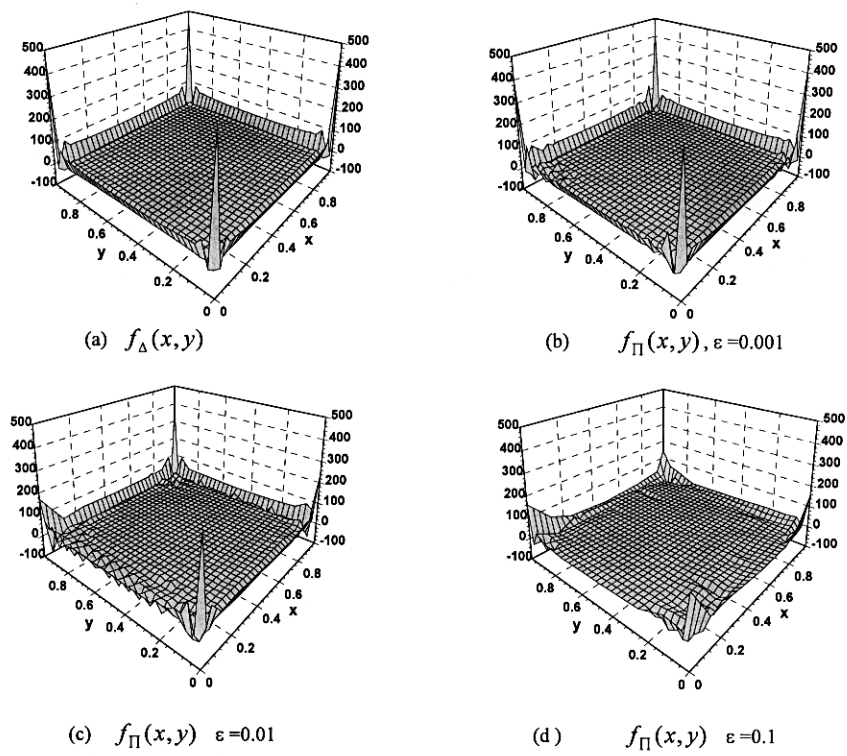


Figure 8. The solution functions of $f_{\Delta}(x,y)$ and $f_{\Pi}(x,y)$ for the different thresholds on the 3-th scale.

The solution functions $f_{\Delta}(x_i,y_j)$ and $f_{\Pi}(x_i,y_j)$ from 1-th to 4-th level of decimation for the threshold $\epsilon = 0.01$ are plotted in Fig. 9.

From the computed results, it is shown that the 2-D AMMM can reproduce well the discontinuity of the function near the circle $(x-0.5)^2+(y-0.5)^2 = 0.3^2$ although it can reduce about 51%, 67%, 71%, 68% the size of the system of linear equations for the different levels of decimations.

Example three: consider the following kernel function

$$k(x,y,x',y') = \frac{1}{\sqrt{(x-x')^2+(y-y')^2}}$$

The region $[0,1]\times[0,1]$ is uniformly subdivided into $N = N_x \times N_y = 33\times 33$. The exact solution is $f(x,y) = 1+(x-\frac{1}{2})x(y+\frac{1}{2})\cos(\frac{\pi}{3}y) - y(1-y)$.

threshold	size	CN	Err	CPU (sec)
MM	1089	138	0	56.19
0.0	1089	248837	1.766E-5	79.48
0.001	384	17295	4.289E-4	34.71
0.01	282	9927	5.605E-4	26.49
0.1	147	6588	7.441E-3	25.05

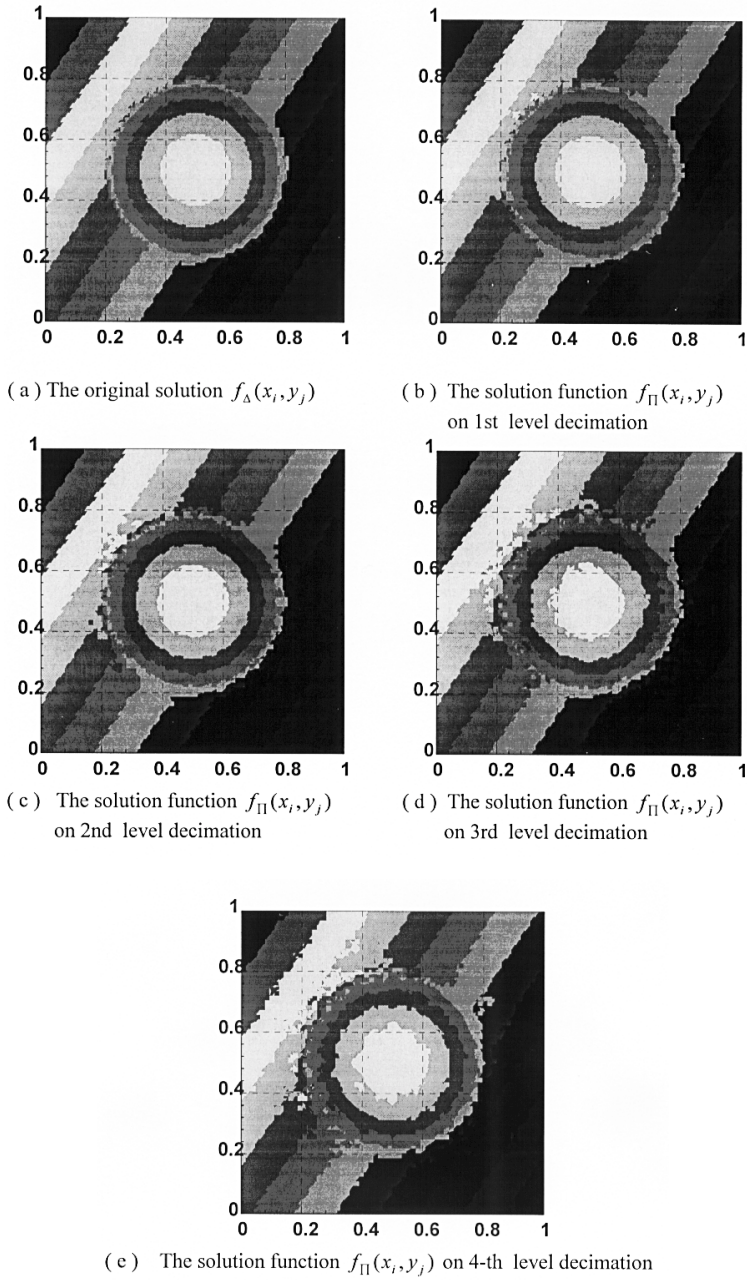
Table 3. Error and CPU time for example three.

For the different thresholds, the actual size and the condition number of the linear equation on the largest scale $J = 3$, the error $Err(f_\Delta, f_\Pi)$ and the CPU time are given in TABLE (3). CN represents the condition number.

When one dimensional AMMM is used to find the solution of the linear equation, for the different thresholds, the actual size and the condition number of the linear equation on the largest scale $J = 3$, the error $Err(f_\Delta, f_\Pi)$ and the CPU time are given in TABLE (4).

The solution $f_\Delta(x_i, y_j)$ and the solution $f_\Pi(x_i, y_j)$ for the different thresholds on the largest scale V=3 obtained through the 2-D AMMM and 1-D AMMM method are plotted in Fig. 10.

From these results, it is shown that 2-D AMMM method can reduce about 65%, 74%, 87% of the size of the linear equation for the different scales $V = 1, 2, 3$. And the solution $f_\Pi(x_i, y_j)$ approximates very well the original function $f_\Delta(x_i, y_j)$. However, although 1-D AMMM also can reduce about 6%, 75%, 87% of the size of the linear equation for the different scales $V = 1, 2, 3$, the solution $f_\Pi(x_i, y_j)$ are not approximated well of the original solution $f_\Delta(x_i, y_j)$. When the approximate solution $f_\Delta(x_i, y_j)$ is plotted in one dimension (see Fig. 11), We can see that the curve of the solution has many discontinuous points and large oscillations. Therefore, 1-D AMMM can not be efficiently used to solve the linear equation.



(a) The original solution $f_{\Delta}(x_i, y_j)$

(b) The solution function $f_{\Pi}(x_i, y_j)$ on 1st level decimation

(c) The solution function $f_{\Pi}(x_i, y_j)$ on 2nd level decimation

(d) The solution function $f_{\Pi}(x_i, y_j)$ on 3rd level decimation

(e) The solution function $f_{\Pi}(x_i, y_j)$ on 4-th level decimation

Figure 9. The solution functions $f_{\Delta}(x_i, y_j)$ and $f_{\Pi}(x_i, y_j)$ from 1-th to 4-th level decimation on the threshold $\epsilon = 0.01$.

threshold	size	CN	Err	CPU (sec)
0.0	1089	5954	4.971E-5	76.57
0.001	1023	5600	9.306E-4	70.23
0.01	277	1915	9.513E-3	18.29
0.1	137	201	2.135E-2	15.55

Table 4. Error and CPU time for example three.

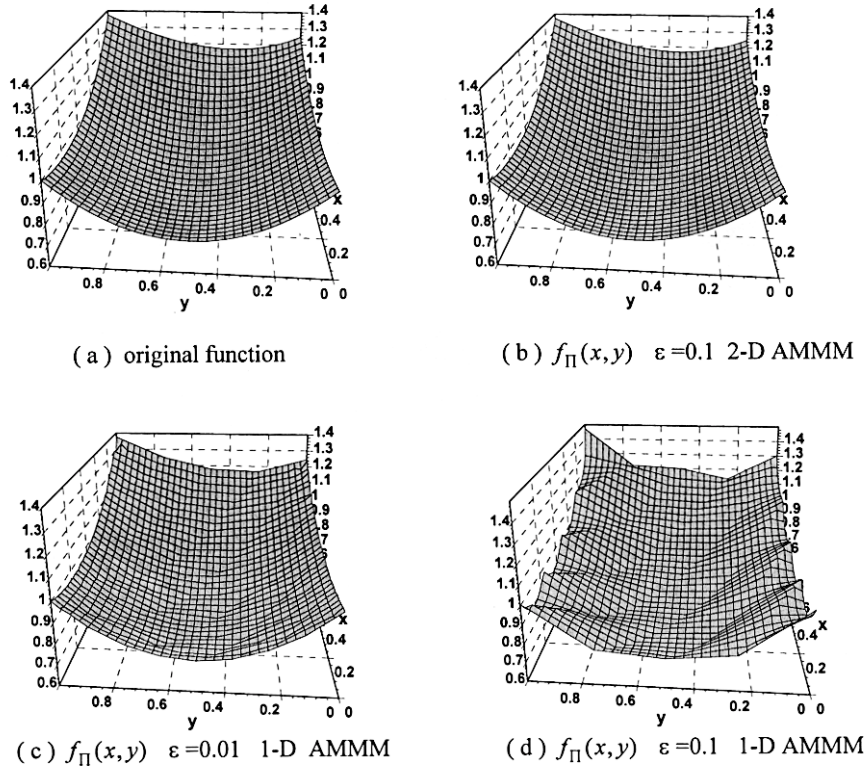


Figure 10. The original function $f_{\Delta}(x_i, y_j)$ and the solution $f_{\Pi}(x_i, y_j)$ obtained by 1-D and 2-D AMMM.

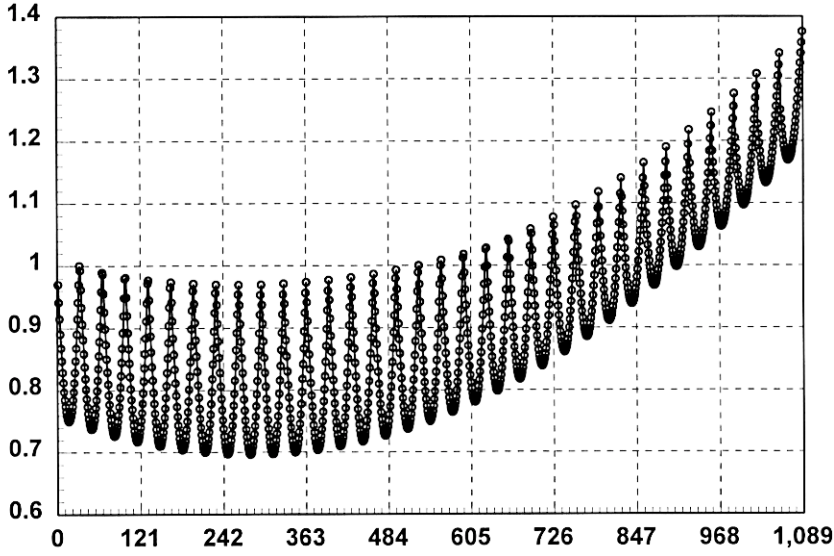


Figure 11. The figure of the original function on one dimension.

7. DISCUSSIONS

From the three examples of numerical simulations for solving the two-dimensional Fredholm integral equation of the first kind by two-dimensional adaptive multiscale moment moment, one can reach the following conclusions:

(1) It is seen that the greater is the number of multiscale, the poorer is the condition number of the linear equations. Because the condition number of the linear equation will be poor as the scale increases, the scale cannot be taken too large. The solution of the linear equation has large oscillations for the larger scales. From 1-scale to 5-scale, the number of unknown of the linear equations is almost 3, 12, 48, 192, 768 times the number of unknown on the 0-scale, respectively. In general, the largest scale should not be taken greater than 5.

(2) The size of linear equation constructed by the adaptive multiscale moment method at the v th scale can be reduced according to the solution of linear equation at the v -1th scale after some small unknowns are filtered out.

(3) The points of discontinuity in the solution can be obtained from the

solution of the linear equation, without getting blurred too much by the Gibb's phenomenon associated with a Fourier series.

(4) If the solution of the integral equation is almost a linear function, the size of the linear equation can be reduced dramatically from the original size of the system of linear equations. This property shows that the method can save computation time and storage in order to find the numerical solution of the integral equation.

(5) The adaptive multiscale moment method can realize automatically the mesh-refinement procedure in the local regions on which the solution function is not smooth .

(6) 2-D AMMM is more efficient than 1-D AMMM when the 2-D integral equation is to be solved because the function can be approximated very well by the linear combination of two-dimensional triangular basis. Efforts in studying Fredholm integral equation for the arbitrary domains in two or three dimension by use of this new technique are under way.

REFERENCES

1. Delves, L. M., *Numerical Solution of Integral Equations,* Clarendon Press, Oxford, 1974.
2. Hackbusch, W., "Integral equations theory and numerical treatment," *ISNM,* Vol. 120, Birkhauser Verlag, Switzerland, 1995.
3. Tikhonov, A. N., and V. Y. Arsenin, *On the Solution of Ill-posed Problems,* John Wiley and Sons, New York, 1977.
4. Backus, G., and F. Gilbert, "Numerical applications of a formalism for geophysical inverse problems," *Geophys. J. Roy. Astron. Soc.,* Vol. 13, 247–276, 1967.
5. Harrington, R. F., *Field Computation by Moment Method,* Hacmillan Press, New York, 1968.
6. Beylkin, G., R. Coifman, and V. Rokhlin, "Fast wavelet transform and numerical algorithm I," *Comm. Pure Appl. Math.,* Vol. 44, 141–183, 1991.
7. Alpert, B. K., G. Beylkin, R. Coifman, and V. Rokhlin, "Wavelet-like bases for the fast solution of second-kind integral equation," *SIAM J. Sci. Comp.,* Vol. 14, 159–184, January, 1993.
8. Steinberg, B. Z., and Y. Leviatan, "On the use of wavelet expansions in the method of moments," *IEEE Trans Antennas Propagat,* Vol. AP-41, No. 5, 610–619, 1993.

9. Goswami, J. C., A. K. Chan, and C. K. Chui,"On solving first-kind integral equations using wavelets on a bounded interval," *IEEE Trans Antennas Propagat,* Vol. AP-43, No. 6, 614–622, June, 1995.
10. Wang, G. F., "A hybrid wavelet expansion and boundary element analysis of electromagnetic scattering from conducting objects," *IEEE Trans Antennas Propagat.,* Vol. AP-43, No. 2, 170–178, February, 1995.
11. Brandt, A., "Multi-level adaptive solutions to boundary value problems," *Mathematics of Computation,* Vol. 31, 330–390, 1977.
12. Hackbusch, W., *Multigrid Methods and Applications,* Springer-Verlag, New York, 1985.
13. McCormick, S. F., *Multigrid Methods: Theory, Applications and Super- computing,* Marcel Dekker, New York, 1988.
14. Mandel, J., "On multilevel iterative methods for integral equations of the second kind and related problems," *Numer. Math.,* Vol. 46, 147–157, 1985.
15. Hemker, P. W., and H. Schippers, "Multiple grid methods for the solution of Fredholm integral equations of the second kind," *Mathematics of Computation,* Vol. 36, No. 153, 1981.
16. Kalbasi, K., and K. R. Demarest, "A multilevel enhancement of the method of moments," *7th Ann. Rev. Progress Appl. Computat. Electromagn.,* Naval, Monterey, CA, 254–263, March, 1991.
17. Kalbasi, K., and K. R. Demarest, "A multilevel formulation of the method of moments," *IEEE Trans. Antennas Propagat.,* Vol. AP-41, No. 5, 589–599, May 1993.
18. Su, C., and T. K. Sarkar, "A multiscale moment method for solving Fredholm integral equation of the first kind," *J. Electromag. Waves Appl.,* Vol. 12, 97–101, 1998.
19. Su, C., and T. K. Sarkar, "Scattering from perfectly conducting strips by utilizing an adaptive multiscale moment method," *Progress In Electromagnetics Research, PIES 19,* 173–197, 1998.
20. Su, C., and T. K. Sarkar, "Electromagnetic scattering from two-dimensional electrically large perfectly conducting objects with small cavities and humps by use of adaptive multiscale moment methods (AMMM)," *J. Electromag. Waves Appl.,* Vol. 12, 885–906, 1998.
21. de Boor, *A Practical Guide to Splines,* Springer-Verlag, New York, 1978.