HW 10.9.5 Find an approximation to the solution of the annulus problem (10.9.1)–(10.9.3) with $R_i = 0.5$, $F(r,\theta) = \exp(r)\sin 2\pi\theta$ for $(r,\theta) \in R_a$, $f_1(\theta) = \sin 4\theta$, $f_2(\theta) = \sin 3\theta$, $\theta \in [0, 2\pi]$. Use difference scheme (10.9.7)–(10.9.11), $M_r = 100$, $M_\theta = 100$, and the optimal SOR scheme with both the difference of successive iterates and the residual as a stopping criterion.

HW 10.9.6 (a) Find an approximate solution of the disk problem (10.9.4)–(10.9.5) with $F(r,\theta) = 0$ for all $(r,\theta) \in R_d$ and $f_2(\theta) = \sin 2\theta$, $\theta \in [0, 2\pi]$. Use difference scheme (10.9.12)–(10.9.16), $M_r = 20$, $M_\theta = 20$, and the Jacobi scheme with the residual as a stopping criterion.

(b) Repeat the solution in part (a) using the optimal SOR scheme with the difference of successive iterates as a stopping criterion.

HW 10.9.7 Find an approximate solution of the disk problem (10.9.4)–(10.9.5) with $F(r,\theta) = \cos\pi r\cos 2\pi\theta$ for $(r,\theta) \in R_d$ and $f_2(\theta) = \sin 4\theta$, $\theta \in [0, 2\pi]$. Use difference scheme (10.9.12)–(10.9.16), $M_r = 100$, $M_\theta = 100$, and optimal SOR with the difference between successive iterates and the residual as a stopping criterion.

HW 10.9.8 Resolve the problem given in HW10.9.7 using the Gauss-Seidel scheme directly on equation (10.9.17).

10.10 Multigrid

10.10.1 Introduction

In Section 10.5.1 we mentioned that in this section we would take advantage of the fact that the components of the error associated with the small eigenvalues were eliminated quickly, and most of the work we did with residual correction schemes was to eliminate the components of the error associated with the large eigenvalues. The basic idea behind multigrid is to eliminate the high frequency components of the error quickly on a fine grid. To accomplish this, the high frequency components of the error will have to correspond to the smallest eigenvalues of the iteration matrix. We then transfer the problem to a coarser grid where high frequency components of the error correspond to some of the lower frequency errors on the previous grid. We can then eliminate these high frequency components of the error on this coarse grid quickly. This process is repeated on yet coarser grids, and the result is finally transferred back to the fine grid. The savings in computational costs are due to both the fact that we are eliminating the errors quickly on the appropriate grid and the fact that the coarse grids

are cheaper to work on. Maybe the most amazing statement about the procedure described above is that it works, and it works well.

In this section we will give an introduction to multigrid. We will include one multigrid algorithm and try to use some computations along with some graphics to illustrate how and why the procedure described above works. For a more in-depth description of the multigrid algorithm and theory, see refs. [8], [20] and [43].

Model Problem

As a part of our discussion, we will return to model problem (10.2.3)–(10.2.7). For convenience, we will set $\Delta x = \Delta y$ and choose $M_x = M_y = M$ so that $M = 2^p$. Obviously, the latter condition restricts our choice of M. It is not a requirement, but as we will see, it is a very convenient assumption. We will consider the model problem both as given in (10.2.3)–(10.2.7) and as the matrix problem $A\mathbf{u} = \mathbf{f}$ as described in equation (10.4.1).

Model Computational Problem

When we use computations to illustrate different aspects of the multigrid scheme, we will use a special case of model problem (10.2.3)–(10.2.7) where we set $f_{j\,k} = 0$ and $F_{j\,k} = 0$ for all j and k, and choose $M = 2^4 = 16$. Obviously, the homogeneous difference equation has the unique zero solution and is the approximation to the homogeneous boundary–value problem (10.2.1)–(10.2.2) that will have the unique zero solution. Hence, we see that in our computations, we do not have to be concerned with truncation error. We will concentrate on the convergence of the multigrid scheme. So as to illustrate our point clearly, we will use an initial guess of

$$\mathbf{u}_0 = \sum_{s=1}^{15}\sum_{p=1}^{15}\mathbf{w}^{p\,s} \tag{10.10.1}$$

where $\mathbf{w}^{p\,s}$, $p, s = 1, \ldots, 15$ are the eigenvectors of A. We note that by using such an initial guess, all eigencomponents of the error are present at a significant level and must be eliminated. The initial error is given by $\mathbf{e}_0 = -\mathbf{u}_0$, but for convenience, when we are discussing the error, we will often plot $-\mathbf{e}_0 = \mathbf{u}_0$. Recall from (10.5.29) that the eigenvalues of A are given by

$$\begin{aligned}\mu_{p\,s} &= \frac{2}{\Delta x^2}\big(2 - \cos\frac{p\pi}{M} - \cos\frac{s\pi}{M}\big) && (10.10.2)\\ &= \frac{4}{\Delta x^2}\big(\sin^2\frac{p\pi}{2M} + \sin^2\frac{s\pi}{2M}\big), && (10.10.3)\end{aligned}$$

and the components of the associated eigenvectors are given by

$$w^{p\,s}_{j\,k} = \sin\frac{jp\pi}{M}\sin\frac{ks\pi}{M}, \quad j,k = 1,\ldots,15, \quad p,s = 1,\ldots,15. \tag{10.10.4}$$

We note that these eigenvectors are orthogonal (this makes some of our test computations a bit easier). *We emphasize that from the computational point of view, initial guess (10.10.1) is a terrible initial guess. A little thought (and/or a plot of* $\mathbf{u}_0$ *or its analytic analogue) will reveal that this function is a terrible approximation to the solution of our model problem. We use this initial guess for the convenience of having all of the modes present.*

We notice that the first and last eigenvectors are given by

$$\mathbf{w}^{1\,1} = \left[\sin\frac{\pi}{16}\sin\frac{\pi}{16}\;\sin\frac{\pi}{16}\sin\frac{2\pi}{16}\;\cdots\;\sin\frac{\pi}{16}\sin\frac{15\pi}{16}\;\sin\frac{2\pi}{16}\sin\frac{\pi}{16}\right.$$
$$\left.\cdots\;\sin\frac{15\pi}{16}\sin\frac{15\pi}{16}\right]^T \qquad (10.10.5)$$

and

$$\mathbf{w}^{15\,15} = \left[\sin\frac{15\pi}{16}\sin\frac{15\pi}{16}\;\sin\frac{15\pi}{16}\sin\frac{2\cdot15\pi}{16}\;\cdots\;\sin\frac{15\pi}{16}\sin\frac{15\cdot15\pi}{16}\right.$$
$$\left.\sin\frac{2\cdot15\pi}{16}\sin\frac{15\pi}{16}\;\cdots\;\sin\frac{15\cdot15\pi}{16}\sin\frac{15\cdot15\pi}{16}\right]^T, \qquad (10.10.6)$$

respectively. We should understand that eigenvectors (10.10.5) and (10.10.6) are analogues of the functions $\sin\pi x\sin\pi y$ and $\sin 15\pi x\sin 15\pi y$, respectively. All of the other eigenvectors can be written similarly, and they all have analytic analogues. Inspecting the vectors $\mathbf{w}^{1\,1}$ and $\mathbf{w}^{15\,15}$ carefully (it may be easier to look at the analytic analogues) shows that $\mathbf{w}^{1\,1}$ is slowly varying and $\mathbf{w}^{15\,15}$ is highly oscillatory. The pattern seen here is generally the case for all of the eigenvectors: The eigenvectors $\mathbf{w}^{p\,s}$ where p and s are small change slowly, whereas when p or q are large, the eigenvector is oscillatory. We will generally refer to the slowly varying eigenvectors as **smooth** or **low frequency** vectors and to the highly oscillatory vectors as **oscillatory** or **high frequency** vectors.

If we were to consider a grid with $M_x = M_y = M/2$ ($M_x = M_y = 8$ in our example), we see that the highest frequency eigenvector is given by

$$\mathbf{w}^{7\,7} = \left[\sin\frac{7\pi}{8}\sin\frac{7\pi}{8}\;\sin\frac{7\pi}{8}\sin\frac{2\cdot7\pi}{8}\;\cdots\;\sin\frac{7\pi}{8}\sin\frac{7\cdot7\pi}{8}\right.$$
$$\left.\sin\frac{2\cdot7\pi}{8}\sin\frac{7\pi}{8}\;\cdots\;\sin\frac{7\cdot7\pi}{8}\sin\frac{7\cdot7\pi}{8}\right]^T, \quad (10.10.7)$$

i.e., analogous to the analytic function $\sin 7\pi x\sin 7\pi y$. It should be clear that the frequency of the highest frequency eigenvector on the $M/2$-grid is a mid-range frequency on the M-grid. The difference between these "high frequencies" on the two grids is what we will exploit in deriving the multigrid iterative scheme.

Model One Dimensional Problem

There are some aspects of multigrid that we felt are just too gross or impossible to illustrate with two dimensional problems. Hence, we will occasionally consider the one dimensional problem analogous to our two dimensional model computational problem. We will consider the boundary–value problem

$$-v'' = 0, \quad x \in (0,1) \qquad (10.10.8)$$
$$v(0) = v(1) = 0, \qquad (10.10.9)$$

along with the difference equation approximation to boundary–value problem (10.10.8)–(10.10.9).

$$-\frac{1}{\Delta x^2}u_{k-1} + \frac{2}{\Delta x^2}u_k - \frac{1}{\Delta x^2}u_{k+1} = 0, \quad k = 1,\ldots,M-1 \quad (10.10.10)$$
$$u_0 = u_M = 0. \qquad (10.10.11)$$

When we use this example, we will either use a general value of M or set $M = 8$.

If we return to our description of the multigrid algorithm given in the first paragraph of this section, it should be reasonably clear that what we must do is

(i) develop an iterative scheme so that on any given grid, the scheme will eliminate the components of the error associated with the high frequency modes, and

(ii) develop grid transfers so that the (p,s) component of the error on the M-grid is transferred to the (p,s) component of the grid on the $M/2$-grid, and the results can be passed from the $M/2$-grid back to the M-grid without introducing new error.

As we shall see, the first step is relatively easy (because we should know quite a bit about iterative schemes by this time), and the second step is not so easy. We are able, however, to complete the second step sufficiently well to develop a very useful multigrid iterative scheme.

10.10.2 Smoothers

We begin by mentioning that we title this section "smoothers" because that is what we really want out of our iterative scheme. We want the iterative scheme to smoothen the error function, i.e., eliminate the oscillatory modes. We begin with an obvious approach by trying some of our favorite schemes. We consider our model computational problem with the initial guess given by $\mathbf{u}_0$, (10.10.1). We should realize that if we were to plot the coefficients of the initial error expanded with respect to the eigenvalues of A, by our definition of $\mathbf{u}_0$ we would get a constant "one" function (actually a minus one, but we have eliminated the minus sign for convenience).

In Figure 10.10.1 we plot the coefficients of the error for our computational model problem expanded with respect to the eigenvectors $\mathbf{w}^{p\,s}$, $p, s = 1,\ldots,15$. This plot represents the error after two Jacobi steps with

$\mathbf{u}_0$, (10.10.1), as our initial guess. It should be pretty clear that the Jacobi scheme is eliminating the middle frequency components, not the high frequency components. This should not surprise us, since the eigenvalues of the Jacobi iteration matrix R_J are

$$\lambda_s^p = \frac{1}{2}\left(\cos\frac{p\pi}{M} + \cos\frac{s\pi}{M}\right)$$

and the associated eigenvectors are $\mathbf{w}^{ps}$, $p, s = 1, \ldots, M$. See example (10.5.1). The error components associated with the smallest eigenvalues of the iteration matrix get smashed down. The smallest eigenvalues of the Jacobi iteration matrix are associated with the mid-range values of p and s. Hence, *the oscillatory components of the error do not get eliminated by the Jacobi iteration scheme.*

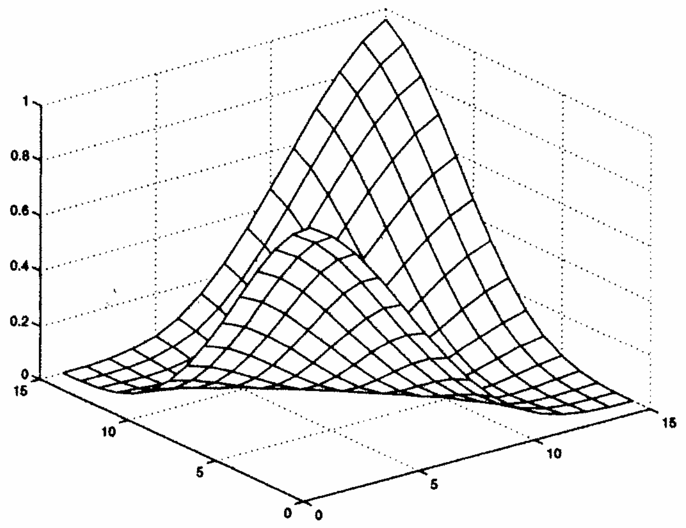

FIGURE 10.10.1. Plot of the coefficients of the error after two Jacobi steps with $\mathbf{u}_0$, (10.10.1), as the starting guess.

Before we proceed with our next candidate, it might be helpful to make it clear how we generate the plot given in Figure 10.10.1. We begin by using a Jacobi scheme where we set *uold* equal to the value defined by $\mathbf{u}_0$ in (10.10.1). After we have performed two Jacobi iterations, we have $\mathbf{u}_2$ and want a_{ps}, $p, s = 1, \ldots, 15$, such that

$$-\mathbf{e}_2 = \mathbf{u}_2 = \sum_{s=1}^{15}\sum_{p=1}^{15} a_{ps}\mathbf{w}^{ps}.$$

This is not difficult, since the vectors $\mathbf{w}^{ps}$ are orthogonal, i.e., we take the dot product of both sides with $\mathbf{w}^{p_0 s_0}$ and get

$$a_{p_0 s_0} = \frac{\mathbf{u}_2 \cdot \mathbf{w}^{p_0 s_0}}{\mathbf{w}^{p_0 s_0} \cdot \mathbf{w}^{p_0 s_0}}.$$

Figure 10.10.1 is a plot of the function a_{ps}, for $p, s = 1, \ldots, 15$.

In HW10.5.16 we introduced the weighted Jacobi scheme. We introduced the weighted Jacobi scheme as the Jacobi analogue to the SOR scheme, i.e., the weighted Jacobi is to the Jacobi scheme as the SOR scheme is to the Gauss-Seidel scheme. In HW10.5.16 we found that we could not overrelax the Jacobi scheme. For convergence, we could only underrelax the scheme, i.e., the weighted Jacobi scheme converges for $0 < \omega \leq 1$. Probably at that time, there appeared to be no redeeming characteristics of the weighted Jacobi scheme. In Figure 10.10.2 we plot the coefficients of the error expanded with respect to the eigenvectors $\mathbf{w}^{ps}$, $p, s = 1, \ldots, 15$. The error plotted is after three weighted Jacobi steps with $\omega = \frac{2}{3}$, beginning again with initial guess $\mathbf{u}_0$, (10.10.1). We see in Figure 10.10.2 that *the weighted Jacobi scheme eliminates the high frequency components of the error.*

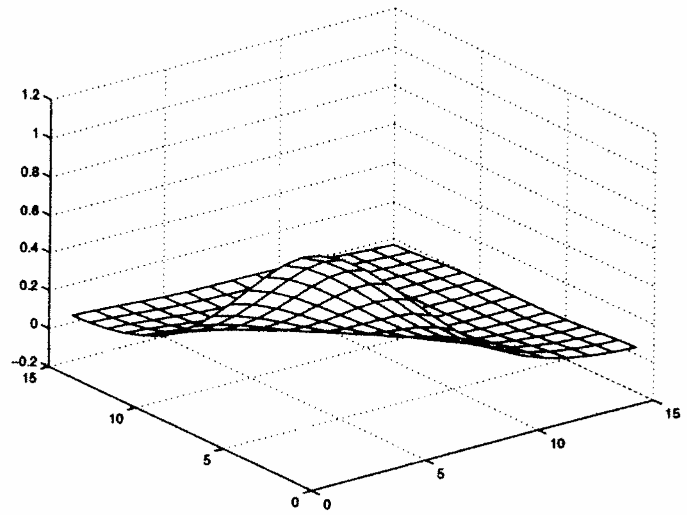

FIGURE 10.10.2. Plot of the coefficients of the error after three weighted Jacobi steps with $\mathbf{u}_0$, (10.10.1), as the starting guess.

It is not difficult to see why the weighted Jacobi scheme eliminates the high frequency components of the error. Using the results of HW10.5.16, we see that the eigenvalues of the iteration matrix for the weighted Jacobi scheme are given by

$$\begin{aligned}\lambda_s^p &= 1-\omega+\omega\lambda_{J_s}^p \quad (\lambda_{J_s}^p \text{ is the eigenvalue of the Jacobi iteration matrix})\\ &= 1-\omega+\frac{\omega}{2}\Big(\cos\frac{p\pi}{M}+\cos\frac{s\pi}{M}\Big)\\ &= 1-\omega\Big(\sin^2\frac{p\pi}{2M}+\sin^2\frac{s\pi}{2M}\Big), \quad p,s=1,\dots,M-1. \qquad (10.10.12)\end{aligned}$$

The eigenvalues get smaller as p and s get larger.

When we choose an iterative scheme to use with multigrid, we really want more than the scheme to eliminate some of the high frequency error modes. During the second step of multigrid, we will work on a grid with $M_x = M_y = M/2$. On this grid we will have only the modes analogous to the analytic modes $\sin\pi x\sin\pi y$, $\sin 2\pi x\sin\pi y, \dots$, $\sin(M/2)\pi x\sin(M/2)\pi y$. For this reason, on the M-grid we would really like to eliminate all of the error associated with the modes from $M/2$ to $M-1$ (or at least eliminate most of the error associated with these modes). It is not difficult to see that the eigenvalues associated with the weighted Jacobi scheme get smaller as ω gets larger. Also, the eigenvalues decrease monotonically with respect to p and s, and for sufficiently large values of ω, the eigenvalues become negative for the larger values of p and s. Hence, we see that if we choose ω so that $\lambda_{M/2}^{M/2} = -\lambda_M^M$, we will damp the modes between $M/2$ and $M-1$ as much as is possible by using the weighted Jacobi scheme. In HW10.10.2 we see that $\lambda_{M/2}^{M/2} = -\lambda_M^M$ if $\omega = \frac{2}{3}$ (which, conveniently, is the value of ω that we used in the calculation given in Figure 10.10.2). We also see from HW10.10.2 that when $\omega = \frac{2}{3}$, then $|\lambda_s^p| \le \frac{1}{3}$ for $p,s = M/2,\dots,M-1$. Hence, when we use $\omega = \frac{2}{3}$, each iteration reduces each high frequency component of the error (the frequencies between $M/2$ and $M-1$) by at least a multiple of $\frac{1}{3}$. In the above discussion and in HW10.10.2 we use λ_M^M. We should be clear that λ_M^M is not an eigenvalue. We use the logical extension of the notation and definition for the eigenvalues to give λ_M^M, which is a convenient bound.

We must realize that there are other iterative schemes that are effective smoothers. One very obvious choice to try is the Gauss-Seidel scheme. In Figure 10.10.3 we give the plot of the coefficients of the error after two Gauss-Seidel iterations. It is clear that *Gauss-Seidel also eliminates the high frequency components of the error*. If we were to look at the eigenvalues of the iteration matrix associated with the Gauss-Seidel scheme, it would not be clear that the Gauss-Seidel scheme would eliminate the high frequency components of the error. We saw in Example 10.5.2 that the eigenvalues of the Gauss-Seidel iteration matrix are given by

$$\lambda_s^p = \Big(\cos\frac{s\pi}{M}+\cos\frac{p\pi}{M}\Big)^2, \quad p,s=1,\dots,M-1$$

(the squares of the eigenvalues of the Jacobi iteration matrix), so it should be clear that the mid-range eigenvalues are the smallest. The difference here is that the eigenvectors of the Gauss-Seidel iteration matrix are not the same as the eigenvectors of the matrix A. In fact, in Section 10.5.7 we saw that the Gauss-Seidel iteration matrix does not have a full set of eigenvectors. It is just the case that the mid-range eigenvalues of the Gauss-Seidel iteration matrix (the smallest eigenvalues of the Gauss-Seidel iteration matrix) correspond to the high frequency components given with respect to the eigenvectors of A.

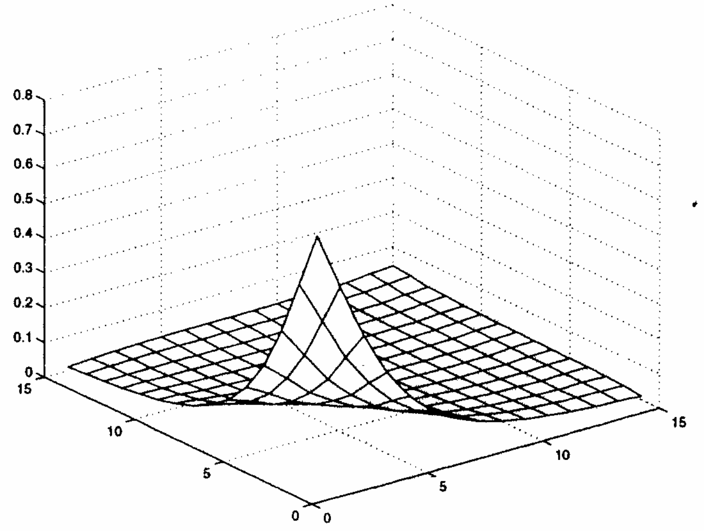

FIGURE 10.10.3. Plot of the coefficients of the error after two Gauss-Seidel steps with $\mathbf{u}_0$, (10.10.1), as the starting guess.

In most of our work on multigrid, we will use the weighted Jacobi scheme. It is not that the weighted Jacobi scheme is the best. We will use it because of the fact that it does work well and because it is very convenient that the eigenvectors of the iteration matrix of the weighted Jacobi scheme and the matrix A are the same.

HW 10.10.1 Consider the one dimensional model problem

$$-v'' = 0, \quad x\in(0,1) \qquad (10.10.13)$$
$$v(0) = v(1) = 0 \qquad (10.10.14)$$

and the finite difference approximation

$$-\frac{1}{\Delta x^2}u_{k-1}+\frac{2}{\Delta x^2}u_k-\frac{1}{\Delta x^2}u_{k+1}=0, \quad k=1,\dots,M-1 \qquad (10.10.15)$$
$$u_0 = u_M = 0. \qquad (10.10.16)$$

(a) Show that the eigenvalues and eigenvectors associated with the matrix