Chapter 10

Higher-Order Techniques

A sound way of constructing high-order approximations to a given differential problem $LU = F$, is first to construct a multigrid program with a low approximation order, and then convert it into a high-order program. The lower order is easier to develop and is also useful as a component in the higher-order program. Such programs are usually more efficient than programs which use high-order difference operators throughout. We mostly recommend the method of §10.2 below, especially for non-elliptic and singular perturbation problems.

10.1 Fine-grid defect corrections. Pseudo spectral methods

Given a program for solving the (linear or nonlinear) low-order (order p_0) discrete system $L_0^h u^h = f^h$, an obvious multigrid approach for raising the approximation order is by high-order "deferred" (or "defect") corrections introduced once per cycle on the currently-finest grid [Bra79b, §3.4]. That is, we add to f^h the correction

$$\sigma_{1,0}^h(x^h) := L_0^h \tilde{u}^h(x^h) - L_1^h \tilde{u}^h(x^h), \tag{10.1}$$

where L_1^h is the higher-order operator, its approximation order (consistency order) being $p_1 > p_0$, and $\tilde{u}^h$ is the current approximate solution. A similar correction is of course introduced to the discrete boundary conditions, too. To save h-cycles one should employ an FMG algorithm (§7), and use corrections like (10.1) at all the FMG stages (i.e., for every currently-finest grid). The total amount of work is then still basically given by (7.4).

Note that that work is proportional to the approximation order p_1. However, this count does not take into account the calculation of (10.1) once per cycle. For lower p_1 this extra work may be less than the other

work within the cycle (a couple of sweeps on each level), but for high p_1 it becomes dominant and makes the amount of work per cycle proportional to p_1 (assuming spectral-type methods cannot be used and the complexity of calculating L_1^h is thus proportional to p_1), hence the total work is in principle $O(p_1^2)$. Furthermore, for higher p_1 we have in principle to use higher computer precision, making the work of each arithmetic operation (in calculating (10.1)) again proportional to p_1, bringing the total work to $O(p_1^3)$. This can be reduced to just $O(p_1^2)$ by a method of *compound deferred corrections*, i.e., taking $p_0 = p_1/2$ and solving for L_0^h by deferred corrections to a system of order $p_0/2$, and so on recursively. In the normally used range of p_1, however, the work of even the uncompounded deferred correction is often still dominated by relaxation and hence still proportional to p_1.

This technique can in particular be applied to **pseudo-spectral approximations** L_1^h, i.e., approximations attaining very high order (proportional to $1/h$) through a discrete spectral (Fourier, Chebyshev, etc.) representation of the solution, using fast numerical transformers (e.g., FFT – the Fast Fourier transform) to obtain that representation and to calculate from it the approximate derivatives at gridpoints (cf. e.g., [GO77]). Using reasonably high order in L_0^h (e.g., $p_0 = 4$, itself calculated by deferred corrections to a second-order operator) one can attain the spectral approximation order with just few applications of the spectral operator. (Other spectral multigrid methods are described in [BFT83], [SZH83], [ZWH82], [ZWH83].)

The deferred correction technique (suggested by L. Fox) is a special case of the concept of defect corrections (see [Lin76], [Ste78], [AS82]). An important advantage of such a technique is that the higher-order operator L_1^h (and the corresponding higher order boundary conditions) need not be stable. This gives much freedom in the relatively difficult task of calculating L_1^h. This freedom is especially welcome in nonelliptic and singular perturbation cases, where convenient central approximations are unstable.

The reason L_1^h need not be stable is that the convergence of the defect correction iterations, to the solution corresponding to L_1^h, is fast only in the smooth components (for which L_0^h is a good approximation to L_1^h) and is very slow in the high-frequency components. Since instability is a property of high-frequencies, it can creep in only very slowly. The growth of unstable modes within the few solution cycles is negligible.

The whole purpose of defect corrections is in fact to correct lowfrequency components; only for such components higher-order approximation, such as L_1^h, are much better than lower-order approximations like L_0^h. Recognizing this and the fact that in multigrid processes low frequencies are converged via the coarse-grid corrections, we see that the main effect of the defect corrections can be obtained by applying them only at the stage of transferring residuals to the coarser grids. This would save about two work-units per cycle, and would give better approximations in case L_1^h is unstable. This idea, from a different point of view, is described in the next

section.

10.2 Double discretization: High-order residual transfers

On any given grid participating in multigrid interactions, discrete approximations to the continuous operator L are used in two different processes: in relaxation sweeps, and in calculating residuals transferred to coarser grids. The two discretization schemes need not be the same [BD79, §3.11]. The discretization L_0^h employed in the relaxation sweeps must be stable (see §12), but its accuracy may be lower than the one we wish to generate. The discretization L_1^h used in calculating the transferred residuals determines the accuracy of our numerical solution, but it need not be stable. This "double discretization" scheme is especially useful in dealing with non-elliptic and singular perturbation problems: One can use the most convenient (but sometimes unstable) central differencing for L_1^h, and add artificial viscosities (see §2.1) only to L_0^h. This will ensure stable solutions which still have the accuracy of the central differencing.

Note that such a multigrid process will not converge to zero residuals, since it uses two conflicting difference schemes. The very point is, indeed, that the solution produced may be a better approximation to the *differential* solution than can be produced by any of the two schemes. More importantly, during running a double discretization solver, instead of checking algebraic convergence, *one should directly check convergence to the differential solution* through the sequence of solutions produced at different stages of the FMG algorithm (see §1.6).

The lack of algebraic convergence makes the usual two-level mode analysis irrelevant for double discretization schemes. Instead they can be analyzed by the two-level FMG mode analysis (§7.4).

Double discretization schemes can of course similarly be applied to **boundary conditions**; e.g., to Neumann conditions: Simple first-order schemes can be used in relaxation, while second-order Neumann conditions (which are sometimes complicated and may sometimes be unstable) can be used to transfer boundary-condition residuals to coarser grids.

The double discretization scheme need not be confined to the currently finest level; it can **also be used on coarser levels**. This will give better coarse-grid corrections, and hence faster algebraic convergence. (In non-elliptic and singular perturbation cases the algebraic convergence is usually determined by the quality of the coarse-grid correction [Bra81a, §5.1].) It is also more convenient to program, since the same residual transfer routine, based on L_1^h, is used on all levels.

Moreover, if only L_0^h is used on coarser levels, the gain in approximation order per cycle cannot be more than p_0; hence the final approximation order cannot exceed $2p_0 + r_0$, where r_0 is the convergence order of relax-

ation [Bra81a, §2.2]. Such a restriction does not exist if L_1^h is used for residual transfers on all levels. The approximation order p_1 can then be attained, perhaps even in one cycle, no matter how high p_1 is. In particular, pseudo-spectral approximations can be used in L_1^h, yielding very high approximation orders in few cycles.

In order to obtain the high approximation orders several rules should be observed: Suitable interpolation orders and residual transfer orders should be employed. The right orders can be derived by crude mode analysis, as in §4.3, but with particular attention to boundary (see in particular rule (C) in §4.3). FMG algorithms with $W(\nu, 0)$ cycles should be used (see §6.2), to ensure accurate enough solution of the course-grid equations and to avoid degradation of the approximation by terminal relaxation. Also, when double discretization is used on all levels together with the Full Approximation Scheme (see §8), notice that two different right-hand sides should be used on coarser grids, one for relaxation and a different one for residual transfers [Bra81a, §2.1].

In case L_1^h is a better approximation than L_0^h not only for smooth components but also in the high-frequency range, the method of fine-grid defect corrections (§10.1) will eventually give smaller errors than the coarse-grid defect correction described here. But the gain will hardly justify the extra work involved in calculating (10.1) separately from the calculation of residuals. In problems where L_1^h is unstable, the present method is both faster and more accurate.

Double discretization schemes have already been used successfully in various cases, including fourth and sixth order approximations to Poisson equation [Sch82]; second-order approximations to simple singular perturbation problems [Bra77a, §7], [Bör81, §7]; and second-order approximations to incompressible Navier-Stokes equations with high Reynolds numbers. Also, the λ extrapolation (§8.4) can be viewed as a special case, where $L_1^h = (2^{p_0} L^h - L^{2h})/(2^{p_0} - 1)$.

10.3 Relaxation with only subprincipal terms

A particularly useful application of the above techniques is to employ a simple relaxation operator L_0^h where non-principal terms are neglected; more precisely, to employ the simplest stable L_0^h which approximate the subprincipal terms of the differential operator (see §2.1). Other terms need to be approximated only in L_1^h. For some fluid-dynamics systems this procedure can save a substantial amount of work. The techniques of either §10.1 or §10.2 can be used with this relaxation; more work is saved by the latter, but the former is safer. On very coarse grids, as the non-principal terms become more important, this type of relaxation may give worse performance. In such cases use more sweeps or reintroduce the neglected non-principal terms.