that required to integrate them exactly, the finite element equations (Equation 17) will give acceptable solutions for the velocity field. This technique of under-integrating the penalty terms is known in the literature as *reduced (order) integration*. For example, if a linear rectangular element is used to approximate the velocity field in a two-dimensional problem, the matrix coefficients $\mathbf{K}^{\mu}$ as well as $\mathbf{K}^{\rho}$ are evaluated using the 2×2 Gauss quadrature, and $\mathbf{K}^{\gamma}$ are evaluated using the one-point 1×1) Gauss quadrature. The one-point quadrature yields a singular $\mathbf{K}^{\gamma}$. Therefore, Equation 17 cannot be inverted, whereas $\mathbf{K}^{\mu} + \mathbf{K}^{\rho} + \mathbf{K}^{\gamma}$ is nonsingular and can be inverted (after assembly and imposition of boundary conditions) to obtain a good finite element solution of the original problem. When a quadratic rectangular element is used, the 3×3 Gauss quadrature is used to evaluate $\mathbf{K}^{\mu}$ and $\mathbf{K}^{\rho}$, and the 2×2 Gauss quadrature is used to evaluate $\mathbf{K}^{\gamma}$.

The choice of the penalty parameter is largely dictated by the ratio of the magnitude of penalty terms to the viscous and convective terms (or compared to the Reynolds number, *Re*), the mesh, and the word length in the computer. The following range of γ is suggested in computations

$$\gamma = 10^4 R_e \text{ to } \gamma = 10^{12} R_e.$$

Summary

Numerical simulation of geomechanical processes requires a good understanding of computational fluid mechanics, heat transfer, and solid mechanics and their couplings. The increase in computing power in both single processor and parallel environments has allowed realistic geomechanics problems of significant complexity and fidelity to be routinely solved and utilized in technological advances. Commercial software has made rapid progress in providing a broad spectrum of analysis capabilities to a variety of industries. Though software is increasingly robust, accurate simulations still require a knowledgeable user, with a background in both mechanics and numerical methods. This entry only provides an introduction to an individual who is interested in the use of the finite element method as a numerical simulation tool for the study and understanding of geomechanical phenomena. The Poisson equation and the Navier–Stokes equations visited here provide the necessary background for the study of diffusion processes and viscous flow problems. Interested readers may consult the references listed.

Acknowledgment

The author gratefully acknowledges the support provided by the Oscar S. Wyatt Endowed Chair.

Bibliography

Bathe, K. J., 1996. *Finite Element Procedures*. Englewood Cliffs: Prentice Hall.

Belytschko, T., Liu, W. K., and Moran, B., 2000. *Nonlinear Finite Elements for Continua and Structures*. Chichester: Wiley.

Bird, P., 1989. New finite element techniques for modeling deformation histories of continents with stratified temperature-dependent rheology. *Journal of Geophysical Research*, **94**(B4), 3967–3990.

Dyksterhuis, S., et al., 2005. Finite-element modelling of contemporary and palaeo-intraplate stress using ABAQUS. *Computers and Geosciences*, **31**(3), 297–307.

Parsons T., 2002. Post-1906 stress recovery of the San Andreas fault system calculated from three-dimensional finite element analysis. *Journal of Geophysical Research* (B8), doi:10.1029/2001JB001051.

Reddy, J. N., 2003. *An Introduction to Nonlinear Finite Element Analysis*. Oxford: Oxford University Press.

Reddy, J. N., 2006. *An Introduction to the Finite Element Method*, 3rd edn. New York: McGraw-Hill.

Reddy, J. N., and Gartling, D. K., 2010. *The Finite Element Method in Heat Transfer and Fluid Dynamics*, 3rd edn. Boca Raton: CRC Press.

Reddy, J. N., Stein, R. J., and Wickham, J. S., 1982. Finite-element modeling of folding and faulting. *International Journal for Numerical and Analytical Methods in Geomechanics*, **6**, 425–440.

Wickham, J. S., Tapp, G. S., and Reddy, J. N., 1982. Finite-element modeling of fracture density in single layer folds. *International Journal for Numerical and Analytical Methods in Geomechanics*, **6**, 441–459.

Cross-references

Numerical Methods, Boundary Element
Numerical Methods, Finite Difference
Numerical Methods, Multigrid

NUMERICAL METHODS, MULTIGRID

Wim A. Mulder
Department of Geotechnology, Delft University of Technology, Delft, The Netherlands

Definition

A multigrid method is an algorithm for the iterative solution of partial differential equations using a sequence of discretizations on multiple scales.

Introduction

The numerical solution of a partial differential equation (PDE) requires its discretization and a method to solve the resulting large system of algebraic equations. For linear equations, the resulting system is often a sparse matrix and a direct solution method suffices if the size of the problem is modest. For problems in three space dimensions, the computational cost of a direct solver may be too large. An iterative method that improves the accuracy of an approximate solution step by step can be a good alternative.

The multigrid method has optimal complexity: the amount of work required to solve a problem with N unknowns is $O(N)$, meaning that it scales with N. It achieves its efficiency by employing several discretization grids for the same problem.

History

Fedorenko (1964) introduced the multigrid method as an iterative scheme for solving Poisson's equation on a square and showed that the number of computations required to determine a solution with a prescribed accuracy is proportional to the number of unknowns, N. Therefore, the method has an optimal computational complexity. Brandt (1973) found that the actual computational cost for a sufficiently accurate result was about 10 work units, where a work unit is the cost of evaluating the discretized equations. He connected the method to local adaptive grid refinement and introduced nonlinear multigrid. Hackbush (1976, 1985) discovered the method independently and provided a mathematical foundation. Since then, the method was developed further to handle PDEs other than Poisson's, which is elliptic, and to go beyond PDEs.

Textbooks include those by Hackbush (1985), who includes convergence proofs; by Wesseling (1991), with chapters on computational fluid dynamics; by Briggs et al. (2000), an easy to read introduction; and by Trottenberg et al. (2001), among others.

Two-grid scheme

A discrete representation of a PDE provides an algebraic system of equations with relations between solution values at neighboring grid points. It is fairly easy to make local corrections that reduce the solution error on a short range, but much harder to correct the long-range or long-wavelength components of the solution that have a more global character. By projecting the solution onto a coarser grid, the long wavelengths become shorter and can effectively be solved for. Combining the corrections to the solution from coarser and finer grids yields an efficient solver.

A simple 1D example can help to understand the fundamentals of the multigrid method. The PDE is $Lu = f$, with $u(x)$ the unknown solution as a function of position x on a finite interval $[x_{min}, x_{max}]$ on the real axis, $f(x)$ a source term or forcing function, and L a linear differential operator, for instance, minus the laplacian, which in 1D is $-\frac{d^2}{dx^2}$. Dirichlet boundary conditions let the solution be zero at the endpoints of the interval. To obtain a discrete representation of the problem, an equidistant 1D grid is defined with grid points $x_k = x_{min} + kh$, index $k = 0, \ldots, N+1$, where $N + 1 = 2^M$ and M is a positive integer. The grid spacing is $h = (x_{max} - x_{min})/(N+1)$. A standard second-order finite-difference scheme leads to

$$\frac{-u_{k-1} + 2u_k - u_{k+1}}{h^2} = f_k, \quad k = 1, \ldots, N,$$

where u_k approximates $u(x_k)$ and $f_k = f(x_k)$. At the boundaries, $u_0 = u_{N+1} = 0$. The discrete equations represent the problem $L^h u^h = f^h$, where L^h is a $N \times N$ sparse matrix and u^h and f^h are vectors of length N. The residual is defined as $r^h = f^h - L^h u^h$ and should vanish once the numerical solution has been found.

A simple iterative method is Jacobi relaxation, in which the matrix L^h is replaced by its diagonal $D^h = 2/h^2$. One step of Jacobi relaxation amounts to

$$u^h := u^h + \omega (D^h)^{-1} r^h.$$

The symbol " := " indicates that the solution values are replaced by the expression on the right-hand side. The factor ω controls the amount of damping. Convergence requires $0 < \omega \leq 1$. Fourier analysis (Hackbush, 1985) shows that the convergence rate, the factor by which the norm of the difference between the current and the exact numerical solution is reduced per iteration, is $1 - O(h^2)$ for Jacobi relaxation. Convergence slows down for increasingly finer grids.

The Fourier analysis reveals that the slow convergence is caused by the long-wave components of the solution. Because long waves can be represented on coarser grids, it makes sense to have a look at a two-grid scheme, outlined in Figure 1 and its caption. The coarse grid consists of every other grid point of the fine grid: $x_K = x_0 + KH$, with $H = 2h$ and $K = 0, \ldots, \frac{1}{2}(N+1)$. A restriction operator $\tilde{I}_h^H$ maps the current fine-grid residual r^h to the coarser grid: $r^H = \tilde{I}_h^H r^h$. The simplest restriction operator is injection: $r_K^H = r_{2K}^h$, $K = 1, \ldots, \frac{1}{2}(N-1)$. Full weighting, which lets $r_K^H = \frac{1}{4} r_{2K-1}^h + \frac{1}{2} r_{2K}^h + \frac{1}{4} r_{2K+1}^h$, applies some smoothing to the residual and is more common. The exact solution of the coarse-grid problem yields the coarse-grid correction $v^H = (L^H)^{-1} r^H$, which should be interpolated back to the fine grid and added as a correction to the latest fine-grid solution. The interpolation or prolongation operator is denoted by I_H^h and lets $u^h := u^h + I_H^h v^H$.

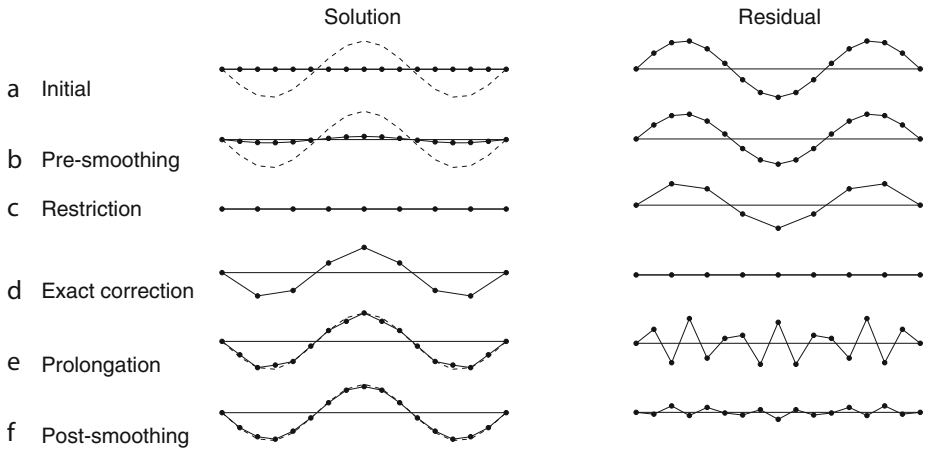
If linear interpolation is used for prolongation, this becomes

$$u_{2K}^h := u_{2K}^h + v_K^H, \quad K = 1, \ldots, \frac{1}{2}(N-1),$$

$$u_{2K-1}^h := u_{2K-1}^h + \frac{1}{2}(v_{K-1}^H + v_K^H),$$

$$K = 1, \ldots, \frac{1}{2}(N+1).$$

Here, it is assumed that $v_0^H = v_{(N+1)/2}^H = 0$. After prolongation, an additional relaxation step with damped Jacobi further removes oscillatory error components of the solution. Jacobi relaxation inside this two-grid scheme has a different purpose than when used as an iterative solver by itself. Instead of removing both the short- and long-wave components of the numerical solution error, it only has to deal with those components that cannot be represented on the coarser grid without aliasing. Therefore, a damped version with $\omega < 1$ can be more effective as it can be optimized to remove the short-wave or



Numerical Methods, Multigrid, Figure 1 Steps in a two-grid cycle. (**a**) The initial solution on the fine grid is set to zero and the residual equals the forcing function. The dashed line represents the exact numerical solution. (**b**) After one step of pre-smoothing with damped Jacobi relaxation, the solution error is still large and the residual has hardly changed in this example. Its restriction to the coarser grid (**c**) is solved exactly, using a coarse-grid version of the discrete differential operator. The resulting correction to the fine-grid solution (**d**) is interpolated or prolongated back to the fine grid and added to the fine-grid solution (**e**). The error in the solution is now dominated by the short wavelengths and appears as an oscillatory function, which is reflected in the corresponding residual. A post-smoothing step (**f**) removes most of the solution error. Repeating the whole cycle will further reduce the error.

oscillatory components. For Figure 1, $\omega = 2/3$ was used. The optimal choice requires a more detailed analysis (Hackbush, 1985).

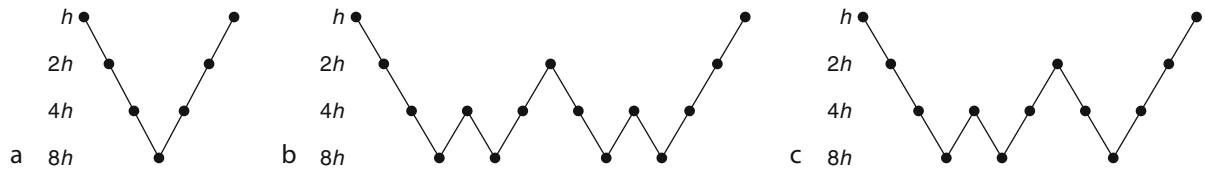
A relaxation scheme geared toward removing oscillatory components is called a smoother. Another popular choice is Gauss–Seidel relaxation. The operator is then approximation by a lower or upper triangular matrix, which is easy to invert. The implementation of the scheme is similar to Jacobi, but the residual is evaluated with the most recent solution available. The result will depend on the order in which the grid is traversed. With lexicographic Gauss–Seidel, one follows the natural index k in increasing order, or in the opposite direction. Symmetric Gauss–Seidel performs both these smoothing steps in sequence. An alternative is red-black Gauss–Seidel, where first the points with an odd and then those with an even index are updated, always using the latest solution values for the residual. In 2D, this would follow a checkerboard pattern.

The grid transfer operator, restriction and prolongation, should obey $m_p + m_r > 2m$ (Hackbush, 1985), where $2m$ is the order of the differential equation and $m_p - 1$ is the highest degree of the polynomial that is interpolated exactly. The scaled adjoint of the restriction operator is an interpolation operator for which m_r can be defined in the same way as m_p. In the example above, $2m = 2$, full weighting has $m_r = 2$, and prolongation based on linear interpolation has $m_p = 2$. More advanced grid transfer operators are based on the differential operator L^h, leading to operator-weighted restriction and prolongation.

The coarse-grid operator L^H can be based on the same discretization as the fine-grid operator L^h. An alternative is the Galerkin coarse-grid approximation $\tilde{I}_h^H L^h I_H^h$. In the current example with full weighting, these happen to be the same. Operator-weighted grid transfer operators can accelerate the convergence by using the coefficients of the differential operator in the construction of restriction and prolongation operators, which in turn will affect the Galerkin coarse-grid approximation of the differential operator.

Multigrid

Instead of using the exact numerical solution on the coarser of the two grids, a two-grid scheme can be applied to the coarse-grid problem. Extending this recursively to a coarsest grid with three points and one unknown, a multigrid solver is obtained. All these grids can be visited in different orders. The standard approach is to start on the finest grid, perform a number of pre-smoothing steps, then move to the coarser grid, again perform pre-smoothing, and continue this until the coarsest grid is reached. There, the exact numerical solution is computed and the resulting coarse-grid correction is prolongated back to the finer level, followed by a number of post-smoothing steps. This is repeated up to the finest grid. Such a sequence is called a V-cycle. A single V-cycle may be insufficient to obtain a convergence rate on the coarser grid that is similar to the smoothing rate on the finer grid. Convergence on coarser grids can be improved by performing more than one cycle. In the W-cycle, the number of cycles doubles on each coarser grid. Figure 2 illustrates the order in which grids are visited for a V-, W-, and F-cycle. The last one is less expensive than a W-cycle, as the number of cycles increases by only one for increasingly coarser grids. Another option, adaptive



Numerical Methods, Multigrid, Figure 2 (**a**) V-cycle, (**b**) W-cycle, and (**c**) F-cycle.

cycling, decides on the sequence of grids by monitoring the decrease of the residuals.

With a proper choice of relaxation scheme or smoother and grid transfer operators, the multigrid method can reach a grid-independent or h-independent convergence rate. This does not, however, mean that the amount of work required to reach a sufficiently accurate solution is proportional to the number of grid points N. The exact numerical solution differs from the true solution of the PDE because of the numerical error due to the discretization, typically with some power p of h. In the earlier example, the discretization is second-order accurate, so $p = 2$. Given this numerical error, convergence to machine precision is more than needed. An iteration error somewhat smaller than the discretization error should suffice. Therefore, more iterations should be carried out if a more accurate solution on a finer grid has to be computed. This leads to an overall complexity of $O(N\log N)$ rather than $O(N)$. Successive grid refinement, in which one first computes a solution on a coarse grid and uses that as an initial guess for the next finer grid and so on, enables removal of the factor $\log N$, assuming that a fixed number of multigrid cycles are used at each level. The combination of successive grid refinement with a multigrid solver is called Full Multigrid (FMG).

Nonlinear multigrid

One way to apply the multigrid method to nonlinear PDEs is the use of Newton's method. A multigrid solver is then applied to a linearization of the discretized nonlinear problem. An example is the computation of galactic gas flow (Mulder, 1986). Another approach is the use of the discretized nonlinear PDE inside the multigrid algorithm. This requires the full solution to be available on the coarser grids. In the example above, only corrections to the solution were represented on coarser grids. The Full Approximation Scheme (FAS) is a reformulation of the multigrid method that includes the full solution (Brandt, 1982). It requires a restriction of the full solution, $u^H = I_h^H u^h$, and of the coarse-grid forcing function, $f^H = L^H(I_h^H u^h) + \tilde{I}_h^H r^h$. The prolongation needs to be changed to $u^h := u^h + I_H^h(u^H - I_h^H u^h)$, where u^H is the sufficiently converged solution on the coarser grid and $I_h^H u^h$ the restriction of the latest fine-grid solution. The restriction operators for the residuals, $\tilde{I}_h^H$, and for the solution, I_h^H, do not have to be the same. Note that the coarse-grid problem, $L^H u^H = f^H$, can be interpreted in a different way: $L^H u^H = f^H = \tilde{I}_h^H f^h + \tau_h^H$. The fine-to-coarse defect correction, $\tau_h^H = L^H(I_h^H u^h) - \tilde{I}_h^H (L^h u^h)$, ensures that the coarse-grid equations maintain the accuracy of the fine-grid equations.

Nonlinear equations of the form $L^h(u^h) = f^h$ fit into this scheme. For $L^H(u^H)$, the same discretization scheme as on the fine grid can be adopted. Smoothing operators can be based on a local linearization of the residuals.

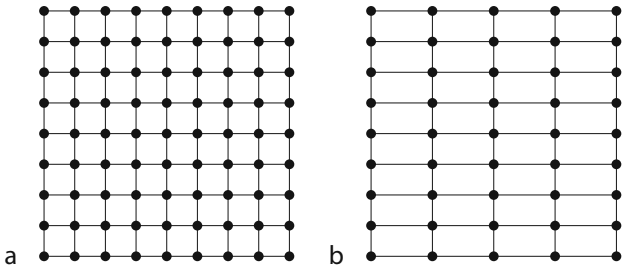
Generalizations

Multigrid is highly efficient for the solution of Poisson's equation on the square, discretized by the finite-difference method on a cartesian grid. For other PDEs, describing convection, diffusion, waves, flow, the method may be less easy to apply. The same is true for unstructured grids, for instance based on triangles or tetrahedra. With suitable modifications and at the expense of additional code complexity, the multigrid method can still be the optimal approach. With unstructured grids, the coarser and finer grid are generally not nested, leading to grid transfer operations that are less easy to code than on a cartesian grid.

Anisotropy in the PDE can degrade performance. Consider the equation

$$-\frac{\partial^2 u}{\partial x^2} - a\frac{\partial^2 u}{\partial y^2} = f(x,y).$$

For $0 < a \ll 1$, smoothing operators may have difficulty with the y-direction where there is only weak coupling between neighboring unknowns. Effective anisotropy in the discretized equation can also happen with grids that have widely different spacings in the various coordinate directions. A more powerful relaxation scheme may repair the poor performance of a simple smoother. For 2D problems, line relaxation will smooth in one direction and solve in the other. An example is line Jacobi relaxation, which approximates the matrix L^h by dropping the off-diagonals in one coordinate direction, while keeping them in the other. The dropped off-diagonals can be subtracted from the main diagonal with a suitable weight factor to obtain better smoothing properties. For 3D problems, plane relaxation is required, which may be too expensive. An alternative is semi-coarsening, where the coarse grid has the same number of points in the direction of weak coupling and is coarsened only in the direction in which the smoothing effectively removes the shorter wavelengths. Figure 3 shows



Numerical Methods, Multigrid, Figure 3 Semi-coarsening of grid (**a**) in the horizontal direction provides grid (**b**).

an example of a grid before and after semi-coarsening in the horizontal direction. In the more general case where semi-coarsening is required in all coordinates, it can be applied in alternating directions. A more costly but powerful approach is simultaneous semi-coarsening in all coordinates (Mulder, 1989), which still retains optimal complexity.

Beyond partial differential equations

The application of the multigrid method is not restricted to the solution of PDEs. There are generalizations to eigenvalue problems (Brandt et al., 1983), integral equations, optimization and optimal control (Borzì and Schulz, 2009), statistical physics (Kandel et al., 1988), image processing (Terzopoulos, 1986), image segmentation, and edge detection. In all cases, the simultaneous use of different scales overcomes the problem of slowly converging solution components.

Algebraic multigrid (AMG), or, better, algebraic multilevel method, refers to a class of solvers that construct sets of coarse-level equations without referring to grids or geometrical properties (Trottenberg et al., 2001; Shapira, 2008). Coarsening is applied to subsets of unknowns for which smoothing is effective. If the problem is specified by a large matrix, these subsets can be determined by examining the coefficients of the matrix. Strongly coupled unknowns, say u_i and u_j, are related by a matrix coefficient a_{ij} that is relatively large. In that case, smoothing is effective and one of the unknowns can be removed from the equations on the coarser level. The coarser level will then be mainly populated by weakly coupled variables. Operator-weighted grid transfer operators are natural in this context, as well as the Galerkin approach for the construction of the coarse-level operator. Algebraic multilevel methods are a popular choice for finite-element discretizations on unstructured grids, as they can be used as a black-box solver and do not require the tedious coding of grid transfer operations between non-nested elements.

Deflation (Vuik et al., 1999) is a technique that accelerates convergence of an iterative procedure if that is slowed down by only a few solution components. Their projection to a subspace that singles them out can be readily solved if the subspace is small.

The multigrid method bears some resemblance to other techniques such as wavelets, hierarchical-basis finite elements, multi-scale techniques, cyclic reduction, fast multipole methods, and other divide-and-conquer methods.

Geophysical applications

Multigrid has been applied successfully to a wide range of partial differential equations, describing static and dynamics problems, diffusion, convection, and flow problems. Examples in geophysics include elliptic problems in potential theory, such as gravity (Boulanger and Chouteau, 2001; Kusche, 2002), magnetostatics (De Gersem and Hameyer, 2001), electrostatics (Bailey and Cheesman, 1996), and Darcy's law in porous media flow (Schmidt, 1995). Controlled-source electromagnetics (Aruliah and Ascher, 2002; Mulder, 2006) and mantle convection (Trompert and Hansen, 1996) are examples of parabolic problems. The wave equation for seismic applications is a hyperbolic problem. Iterative solution of its frequency-domain formulation, the Helmholtz equation, was a notoriously difficult numerical problem but can nowadays be accomplished by using a damped version as preconditioner in a conjugate-gradient-type iterative scheme. Multigrid efficiently deals with the approximate inversion of the preconditioner (Erlangga et al., 2006; Riyanti et al., 2006).

Summary

The multigrid method provides an optimal iterative solution method for a wide class of problems governed by partial differential equations. Its application to a finite-difference discretization of Poisson's equation on a square is the easiest. For other problems, a bit more effort may be required.

Bibliography

Aruliah, D. A., and Ascher, U. M., 2002. Multigrid preconditioning for Krylov methods for time-harmonic Maxwell's equations in three dimensions. *SIAM Journal on Scientific Computing*, **24**(2), 702–718, doi:10.1137/S1064827501387358.

Bailey, R. C., and Cheesman, S., 1996. A multigrid solver for inductive limit EM responses in orebody delineation. *SEG, Expanded Abstracts*, **15**, 265–268, doi:10.1190/1.1826615.

Boulanger, O., and Chouteau, M., 2001. Constraints in 3D gravity inversion. *Geophysical Prospecting*, **49**(2), 265–280, doi:10.1046/j.1365-2478.2001.00254.x

Borzi, A., and Schulz, V., 2009. Multigrid methods for PDE optimization. *SIAM Review*, **51**(2), 361–395, doi:10.1137/060671590.

Brandt, A., 1973. Multi-level adaptive technique (MLAT) for fast numerical solution to boundary value problems. In *Proceedings of the Third International Conference on Numerical Methods in Fluid Mechanics*, Lecture Notes in Physics, Vol. 18, pp 82–89. Springer-Verlag. ISBN 978-3-540-06170-0, doi:10.1007/BFb0118663.

Brandt, A., 1982. Guide to multigrid development. Multigrid Methods. In *Proceedings of the Conference Held at Köln-Porz*, November 23–27, 1981. Hackbush, W., and Trottenberg, U. (eds.), Lecture Notes in Mathematics, Vol. 960: 220–312. Springer-Verlag, ISBN 978-3-540-11955-5, doi:10.1007/BFb0069930.

Brandt, A., McCormick, S., and Ruge, J., 1983. Multigrid methods for differential eigenproblems. *SIAM Journal on Scientific and Statistical Computing*, **4**(2), 244–260, doi:10.1137/0904019.

Briggs, W. L., Henson, V. E., and McCormick, S. F., 2000. *A Multigrid Tutorial*, 2nd edn. Philadelphia: Society for Industrial and Applied Mathematics.

Erlangga, Y. A., Oosterlee, C. W., and Vuik, C., 2006. A novel multigrid based preconditioner for heterogeneous Helmholtz problems. *SIAM Journal on Scientific Computing*, **27**(4), 1471–1492, doi:10.1137/040615195.

Fedorenko, R. P., 1964. The speed of convergence of one iterative process. *USSR Computational Mathematics and Mathematical Physics*, **4**(3), 227–235, doi:10.1016/0041-5553(64)90253-8.

De Gersem, H., and Hameyer, K., 2001. Full multigrid for magnetostatics using unstructured and non-nested meshes. *IEEE Transactions on Magnetics*, **37**(5), 3460–3464, doi:10.1109/20.952637.

Hackbush, W., 1976. Ein iteratives Verfahren zur schnellen Auflösung elliptischer Randwertprobleme, Mathematisches Institut, Universität zu Köln, Report 76–12.

Hackbush, W., 1985. Multi-Grid Methods and Applications, Springer Verlag, Berlin (2nd printing, 2003). http://www.springer.com/math/cse/book/978-3-540-12761-1.

Kandel, D., Domany, E., Ron, D., Brandt, A., and Loh, E., 1988. Simulations without critical slowing down. *Physical Review Letters*, **60**(16), 1591–1594, doi:10.1103/PhysRevLett.60.1591.

Kusche, J., 2002. On fast multigrid iteration techniques for the solution of normal equations in satellite gravity recovery. *Journal of Geodynamics*, **33**(1–2), 173–186, doi:10.1016/S0264-3707(01)00062-X.

Mulder, W. A., 1986. Computation of the quasi-steady gas flow in a spiral galaxy by means of a multigrid method. *Astronomy and Astrophysics*, **156**, 354–380.

Mulder, W. A., 1989. A new multigrid approach to convection problems. *Journal of Computational Physics*, **83**(2), 303–323, doi:10.1016/0021-9991(89)90121-6.

Mulder, W. A., 2006. A multigrid solver for 3D electromagnetic diffusion. *Geophysical Prospecting*, **54**(5), 663–649, doi:10.1111/j.1365-2478.2006.00558.x.

Riyanti, C. D., Erlangga, Y. A., Plessix, R.-E., Mulder, W. A., Vuik, C., and Oosterlee, C. W., 2006. A new iterative solver for the time-harmonic wave equation. *Geophysics*, **71**(5), E57–E63, doi:10.1190/1.2231109.

Schmidt, G., 1995. Operator weighing in a multigrid method for locally refined grids. *Journal of Engineering Mathematics*, **29**(5), 473–496, doi:10.1007/BF00043979.

Shapira, Y., 2008. *Matrix-Based Multigrid: Theory and Applications Numerical Methods and Algorithms* Vol. 2, p. 322, 2nd edn. Springer Verlag, Berlin. ISBN: 978-0-387-49764-8.

Terzopoulos, D., 1986. Image analysis using multigrid relaxation methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **8**(2), 129–139, doi:10.1109/TPAMI.1986.4767767.

Trompert, R. A., and Hansen, U., 1996. The application of a finite volume multigrid method to threedimensional flow problems in a highly viscous fluid with a variable viscosity. *Geophysical and Astrophysical Fluid Dynamics*, **83**(3–4), 261–291, doi:10.1080/03091929608208968.

Trottenberg, U., Oosterlee, C. W., and Schuller, A., 2001. *Multigrid*. London: Academic. ISBN 012701070X.

Vuik, C., Segal, A., and Meijerink, J. A., 1999. An efficient preconditioned CG method for the solution of a class of layered problems with extreme contrasts in the coefficients. *Journal of Computational Physics*, **152**(1), 385–403, doi:10.1006/jcph.1999.6255.

Wesseling, P., 1991. *An Introduction to Multigrid Methods*. Chichester: Wiley. ISBN 0471930830.

Cross-references

Numerical Methods, Finite Difference
Numerical Methods, Finite Element