

# MODEL PROBLEMS FOR THE MULTIGRID OPTIMIZATION OF SYSTEMS GOVERNED BY DIFFERENTIAL EQUATIONS\*

ROBERT MICHAEL LEWIS<sup>†</sup> AND STEPHEN G. NASH<sup>‡</sup>

**Abstract.** We discuss a multigrid approach to the optimization of systems governed by differential equations. Such optimization problems appear in many applications, and are of a different nature than systems of equations. Our approach uses an *optimization-based* multigrid algorithm in which the multigrid algorithm relies explicitly on nonlinear optimization models as subproblems on coarser grids. Our goal is not to argue for a particular optimization-based multigrid algorithm, but instead to demonstrate how multigrid can be used to accelerate nonlinear programming algorithms. Furthermore, using several model problems we give evidence (both theoretical and numerical) that the optimization setting is well-suited to multigrid algorithms. Some of the model problems show that the optimization problem may be more amenable to multigrid than the governing differential equation. In addition, we relate the multigrid approach to more traditional optimization methods, as further justification for the use of an optimization-based multigrid algorithm.

**Key words.** Multigrid methods, optimization of systems governed by differential equations

**1. Introduction.** We consider the optimization of systems governed by differential equations. The nonlinear optimization problems have the form

$$(1.1) \quad \underset{a}{\text{minimize}} \quad F(a) = f(a, u(a)),$$

where  $a$  is a set of design variables, and  $u = u(a)$  is a set of state variables. Given  $a$ , the state variables are defined implicitly by a system of state equations

$$(1.2) \quad S(a, u(a)) = 0$$

in  $a$  and  $u$ . We assume that  $S(a, u) = 0$  is a system of partial differential equations; in our computational tests, we actually solve  $S(a, u) = 0$  for  $u$  given  $a$ . The design variables  $a$  are parameters of the differential equation. For instance,  $a$  might parameterize certain material properties that appear as coefficients of the differential operator in (1.2), or  $a$  might represent the geometry of some object whose performance we wish to optimize. There are typically additional constraints on  $a$  and  $u(a)$ , as well.

We propose to solve (1.1)–(1.2) using an *optimization-based* multigrid algorithm. The multigrid algorithm relies explicitly on nonlinear optimization models as subproblems on coarser grids. We argue that this approach can have theoretical and practical advantages over applying more traditional multigrid algorithms to the system of nonlinear equations that results from the Karush–Kuhn–Tucker (KKT) stationarity conditions for (1.1). In particular

- The optimization perspective is broader than trying to reduce (1.1) to a system of nonlinear equations. In particular, inequality constraints are handled in a natural way.

---

\*Release: 1.1. Date: Sep 6 2003.

<sup>†</sup>Department of Mathematics, College of William & Mary, P.O. Box 8795, Williamsburg, Virginia, 23187-8795, [buckaroo@math.wm.edu](mailto:buckaroo@math.wm.edu). This research was supported by the National Science Foundation under grant DMS-0215444, and by the National Aeronautics and Space Administration under grant NCC-1-02029 and under Contract No. NAS1-97046 while the author was in residence at the ICASE, NASA Langley Research Center, Hampton, VA 23681-2199.

<sup>‡</sup>Associate Dean, School of Information Technology and Engineering, Mail Stop 5C8, George Mason University, Fairfax, VA 22030, [snash@gmu.edu](mailto:snash@gmu.edu). Stephen G. Nash was partially supported by National Science Foundation grant DMI-9800544.

- An optimization-based multigrid algorithm has better guarantees of convergence than a multigrid algorithm for a system of nonlinear equations.
- The optimization model can be better suited to multigrid than the underlying state equation (1.2). In particular, we can expect that the reduced Hessian of the optimization model will frequently be something like an elliptic operator. Thus, the multigrid algorithm can be effective even if the state equation (1.2) is not elliptic.

We view multigrid as a general means of accelerating optimization algorithms. Our goal is not so much to argue for a particular optimization-based multigrid algorithm, but instead to demonstrate that an optimization-based approach is distinct from and (in our view) superior to an equation-based multigrid approach. For this reason, we focus on model problems that illustrate the essential ideas in our arguments, rather than on elaborate computational examples.

While we assume that (1.2) is a differential equation, the approach applies to problems with other types of governing equations such as integral equations. For simplicity, we assume in our exposition that there are no other constraints on the variables  $a$  and  $u$ , although this is not essential. In fact, some of our numerical examples include additional constraints. The presence of additional constraints, including inequalities, makes the application of multigrid to optimization a true generalization of the multigrid approach.

The problem (1.1)–(1.2) represents a family of optimization problems, each corresponding to a particular discretization of the differential equation (1.2). The finer the discretization, the more effort that must be expended to solve (1.2) and thus (1.1). We would like to use computations on a coarse discretization of (1.1)–(1.2) to improve an approximate solution of a finer-resolution problem. To do this, we propose adapting a multigrid algorithm developed in [21], called MG/Opt.

MG/Opt recursively uses coarse resolution problems to generate search directions for finer resolution problems. A line search is then used to refine the solution of each finer resolution problem. The use of a line search globalization technique makes it possible to prove convergence results for the overall multigrid optimization algorithm.

MG/Opt is inspired by multigrid methods for elliptic partial differential equations [5, 11, 19]. Multigrid algorithms for elliptic PDEs are computationally efficient, and this efficiency is the basis of their popularity. Multigrid methods for other types of PDEs have required considerably more effort and the results are mixed for certain classes of problems. The overall goal of this paper is to demonstrate that multigrid methods can be successful in the more general setting of optimization, and in fact may be more successful than when applied to PDEs alone.

In this paper we consider problems (1.1)–(1.2) where the differential equation (1.2) need not be elliptic. We present evidence, both theoretical and numerical, that a multigrid algorithm can often be an effective tool in this setting. This may seem paradoxical since a multigrid algorithm might not be appropriate for the solution of the governing differential equation itself. Our results suggest that the optimization problem (1.1)–(1.2) is better suited to a multigrid approach than the differential equation might be if considered in isolation. The new work confirms and justifies some preliminary numerical results reported in [17].

It would be desirable to present general results indicating the superiority of the optimization viewpoint, but the problem (1.1)–(1.2) is too general for this to be achievable. In particular, the governing PDE is arbitrary, the problem has a general nonlinear objective function, and the constraints can be almost anything. There are

also further complications of a more technical nature (see Section 3.2).

We have chosen a middle course between general theory and particular experiments. We have selected a set of model problems whose properties are, we believe, representative of wide classes of interesting problems. The model problems are sufficiently simple to permit analysis, yet difficult enough to distinguish the behavior of a variety of optimization algorithms. They confirm that multigrid can be an effective tool for optimization, and that the optimization problem can be better suited to multigrid than the differential equation in isolation.

We rely on Fourier analysis of the nonlinearities in the model optimization problems. Similar techniques were used in [4] to examine the Hessians for certain aerodynamic optimization problems to obtain Hessian approximations in terms of differential operators. We have chosen simpler model problems that lend themselves to complete analysis, and also examine the numerical discretizations of the problems and their suitability for multigrid solution.

The multigrid algorithm MG/Opt is adapted from [21]. Unlike most prior applications of multigrid in optimization, MG/Opt uses a globalization technique (a line search on a merit function) to ensure global convergence for the resulting optimization algorithm (i.e., convergence to stationary points from arbitrary starting points). Some of this prior work is described below.

The work in [21] was motivated by variational problems, problems lacking a governing differential equation and a set of design variables  $a$ . Instead, the optimization problem only involves  $u$ . Examples include the minimization of the energy functionals associated with the minimal surface equation or obstacle problems. The multigrid approach outlined in [9] is motivated by similar problems. However, unlike MG/Opt, it does not incorporate any globalization technique.

The work in [14, 27] reduces (1.1)–(1.2) to the Karush-Kuhn-Tucker (KKT) conditions for the equivalent equality constrained problem

$$(1.3) \quad \begin{array}{ll} \underset{a,u}{\text{minimize}} & f(a, u) \\ \text{subject to} & S(a, u) = 0 \end{array}$$

and then applies multigrid to the resulting system of equations. This approach does not use a merit function, so there is a potential for non-convergence for nonconvex problems. Moreover, it is not straightforward to extend an approach based on the KKT conditions to problems that include inequality constraints. Thus, it lacks the guarantees of convergence, and the range of applicability, of MG/Opt.

The work in [7] considers a more limited set of problems where the number of design variables  $a$  is fixed and only  $u$  changes with the level of discretization. Reduced Hessians based on coarser grid calculations are used as preconditioners in a sequential quadratic programming approach. This approach proved effective in accelerating an SQP algorithm for (1.1)–(1.2). However, this approach is one of successive refinement, where the algorithm makes one pass from the coarsest level of discretization to the finest, using the solution of the optimization problem from the previous level of discretization as a starting point for the current one. The approach lacks the multigrid feature of passing back and forth between the various levels of discretization. Our computational experiments demonstrate that successive refinement may be significantly inferior to the multigrid strategy in MG/Opt.

While similar in spirit to this earlier work, the algorithm MG/Opt should be viewed as a nonlinear programming adaptation of the multigrid idea. In particular,

the multigrid subproblems are nonlinear optimization problems, not systems of linear or nonlinear equations.

MG/Opt can be related to more traditional optimization techniques. In particular, it is related to the approximation management schemes discussed in [2, 3]. MG/Opt is also related to the steepest descent method and to Newton’s method for optimization. (See Section 7 for further details.) These relationships provide further justification for the application of multigrid to optimization models.

Here is an outline of the paper. The multigrid algorithm MG/Opt is explained in Section 2. Section 3 gives an analytical motivation for the approach. Model problems are discussed in Sections 4–6. Section 7 explains the relationship of MG/Opt to model management and other optimization algorithms. Some conclusions follow in Section 8.

**2. The Multigrid Algorithm.** The algorithm MG/Opt given here is adapted from [17, 21]. A detailed discussion of more general versions of MG/Opt can be found in [15]. The MG/Opt algorithm can be considered as a “template” since components of the algorithm are left unspecified. In particular, the statement of the algorithm does not identify the optimization algorithm used, nor does it specify particular discretizations or projection operators.

This ambiguity is deliberate, since we wish to focus on properties of the algorithm that are valid without regard to the details of the implementation. (In order to guarantee convergence, however, the components of the algorithm must satisfy certain assumptions [21].) Our aim is to demonstrate that a optimization-based multigrid *approach* is broadly advantageous, not that a particular optimization-based multigrid *algorithm* is superior to competitors. We hope that our results will encourage others to extend existing multigrid algorithms and results to an optimization setting.

We apply MG/Opt to the problem (1.1)–(1.2), and assume that the state equation (1.2) is solved for  $u$  given  $a$ . The statement of the algorithm is more general, however, and permits the use of other approaches, such as simultaneous analysis and design (SAND) [12]. In SAND, one views the problem (1.1)–(1.2) as being posed in the form (1.3). Our observations apply to various reduced basis SQP approaches to (1.3), such as those described in [8, 10, 13, 16]. These approaches have been widely studied and go back at least as far as [1].

The multigrid algorithm makes reference to a set of discretizations which we simply call “grids”. No assumptions are made about the choice of grids in the definition of the algorithm, although we limit ourselves to uniform grids in the analysis of the model problems. The notation for the algorithm suggests that a discretization is applied to both the design variables  $a$  and the state variables  $u$ . It is not necessary, however, to discretize the design variables; in many practical settings, there will be a small, fixed-size set of design variables. The algorithm applies without change to this case. (See [17] for a more detailed discussion and computational examples.) In the analysis of the model problems, however, the design variables  $a$  correspond to the discretization of a function.

Given coarser and finer mesh parameters  $H$  and  $h$ , respectively,  $I_H^h$  denotes a prolongation operator that transfers information from the coarser mesh to the finer mesh, while  $I_h^H$  denotes a restriction operator that transfers information from the finer mesh to the coarser mesh. We make the standard assumption that

$$I_H^h = [\text{constant}] \times (I_h^H)^T .$$

Here is the algorithm MG/Opt. One iteration of the algorithm takes a step from  $a^{(0)}$ , an initial estimate of the solution on the finest grid, to  $a^{(1)}$ , via:

- If on coarsest grid, minimize  $F_h(a_h) = f_h(a_h, u_h(a_h))$ , with initial estimate  $a_h^{(0)}$ , to obtain  $a_h^{(1)}$ .
- Otherwise:
  - Partially minimize  $F_h(a_h)$ , with initial estimate  $a_h^{(0)}$ , to get  $a_{h,1}$ . Down-date the result to obtain  $a_{H,1} = I_h^H a_{h,1}$ .
  - Compute  $v_H = \nabla F_H(a_{H,1}) - I_h^H \nabla F_h(a_{h,1})$ .
  - Recursively apply MG/Opt (with initial estimate  $a_{H,1}$ ) to solve:

$$\underset{a_H}{\text{minimize}} F_H(a_H) - v_H^T a_H$$

subject to the bound constraints

$$a_{H,low} \leq a_H \leq a_{H,up}$$

to obtain  $a_{H,2}$ . (See below for a definition of the bounds.)

- Compute the search direction  $e_h = I_H^h(a_{H,2} - a_{H,1})$ .
- Use a line search to obtain  $a_{h,2} = a_{h,1} + \alpha e_h$ .
- Partially minimize  $F_h(a)$ , with initial estimate  $a_{h,2}$ , to obtain  $a_h^{(1)}$ .

Algorithm MG/Opt is a multigrid algorithm with a V-cycle template for traversing the grids. Other templates could be used by making simple modifications to the recurrence. The algorithm is initialized with a specified estimate on the fine grid. An alternative approach, and one used in our computational tests, is to use a full multigrid initialization scheme (see, for example, [19]).

The algorithm MG/Opt is inspired from the full-approximation scheme multigrid algorithm for solving systems of nonlinear equations (see, for instance, [19]). One can apply the full-approximation scheme to the KKT optimality conditions for (1.1)–(1.2), but we believe that our approach is preferable. For example, the optimization approach used in MG/Opt has stronger guarantees of convergence, and it can be extended to solve problems with inequality constraints.

Within MG/Opt, we interpret “partially minimize” to mean that at least one iteration of a globally-convergent nonlinear optimization algorithm is applied to the optimization model. If the state equation (1.2) is solved for  $u$  given  $a$ , then, under appropriate assumptions, it is possible to prove that MG/Opt will converge to a stationary point of the optimization model [21]. Also, the search direction  $e_h$  will be a descent direction, ensuring that the estimate of the solution improves at every iteration of the multigrid algorithm.

In our model problems, the state equation (1.2) is solved for  $u$  given  $a$ . Consequently, we can use the objective function as a merit function in the line search.

If a line search is not used to update the estimate of the solution, then the algorithm is not guaranteed to converge at all, let alone to a stationary point of the optimization model. Suppose, alternatively, that multigrid (or some other nonlinear equation solver) were applied to the optimality conditions for the optimization problem, which we will denote by  $P(a, u) = 0$  for some vector-valued function  $P$ . Then the best that can be guaranteed is that the algorithm will find a stationary point of  $\|P\|$ , which need not be a stationary point of the optimization model.

The recursion requires MG/Opt to solve a shifted version of the optimization problem subject to the bound constraints

$$a_{H,low} \leq a_H \leq a_{H,up}.$$

These bounds are added to improve performance. They prevent the computation of too long a step based on the coarser grid calculations, which are only approximations

of the fine grid problem. (See also the discussion of model management in Section 7.) The bounds used are the same as those in [17], namely

$$\begin{aligned} a_{H,low} &= a_{H,1} - \gamma e \\ a_{H,up} &= a_{H,1} + \gamma e \end{aligned}$$

where

$$\begin{aligned} e &= (1, \dots, 1)^T \\ \gamma &= \max\{\|v_H\|, \|\nabla F_H(a_{H,1})\|, \|I_h^H \nabla F_h(a_{h,1})\|\}. \end{aligned}$$

The bounds are heuristic, and are related to trust-region approaches for nonlinear optimization [23].

It might appear that the inclusion of these bounds would be a potential drawback of the MG/Opt algorithm, since it adds inequality constraints to a problem (1.1)–(1.2). However, extensive research on trust-region methods has led to the development of effective techniques for solving subproblems of this type. In addition, much optimization software is capable of solving problems with bound constraints on the variables. Most significantly, the bounds only serve as a trust region globalization technique and could be replaced by some other trust region such as a sphere or an ellipse. The choice of a bound constrained region was convenient for our work here since we were working with an optimization package designed to treat bounds, and since some of our test problems were bound constrained.

Both the line search and the extra bound constraints contribute to the convergence and the performance of the algorithm, as can be seen from the results for the third model problem (see Section 6.2).

The algorithm MG/Opt does not make detailed assumptions about the underlying optimization algorithm, the discretization, the update/downdate operators, or the method used to solve the state equations (1.2). This allows considerable flexibility in implementing MG/Opt. In particular, it permits the use of specialized software for the various components of the algorithm (e.g., grid generators, PDE solvers). Of course, in our computational experiments we specify in detail a particular version of MG/Opt.

**3. Overview.** In this section we explain why an optimization-based multigrid algorithm might be appropriate for problems such as (1.1)–(1.2), and comment on some of the difficulties in analyzing such an algorithm. Our analysis is based on the analytical properties of the Hessian of the objective in (1.1). As we discuss, the analytical properties can have significant numerical consequences, and we focus on those features relevant for the general MG/Opt algorithm.

We use the quadratic Taylor’s series approximation of  $F$  to model the nonlinearity of the objective. The nonlinearity of  $F$  is modeled by its Hessian, and so we focus on the latter quantity and its qualitative behavior. Our discussion translates ideas from the solution of equations to the setting of optimization.

### 3.1. The nature of the Hessian and large scale optimization methods.

Consider the numerical solution via conjugate gradients of Poisson’s equation  $\Delta u = q$  with homogeneous boundary conditions. The linear, positive definite system of equations  $Ax = b$  that results from the discretization of the Laplacian is equivalent to the optimization problem

$$\text{minimize } \frac{1}{2} x^T A x - b^T x.$$

As is well-known, traditional iterative methods for solving linear equations on a single grid have difficulties with this problem. Methods such as the unpreconditioned conjugate gradient method will quickly determine the high-frequency components of the solution, but will then take many iterations to determine the low-frequency components of the solution. (When applied to a quadratic problem such as this, many nonlinear optimization methods are equivalent to iterative methods for solving linear equations, so these comments are relevant to MG/Opt.)

This behavior can be understood by examining the Krylov subspaces associated with the conjugate gradient method. The  $k$ -th iterate of the conjugate gradient algorithm minimizes the quadratic over the Krylov subspace

$$\text{span}\{b, Ab, A^2b, \dots, A^{k-1}b\}.$$

Since  $A$  is an approximation of the Laplacian  $\Delta$  and  $b$  is an approximation of  $q$ , this Krylov subspace is a discrete approximation to

$$\text{span}\{q, \Delta q, \Delta^2 q, \dots, \Delta^{k-1} q\}.$$

The Laplacian is a differential operator, so the basis functions for the Krylov subspace are increasingly oscillatory. In contrast, the solution is smoother than  $q$ . Thus, the Krylov vectors  $\Delta^j q$  are increasingly poor approximations to the solution. Consequently, the conjugate gradient iterations can rapidly resolve the oscillatory features of the solution, but will have difficulty making progress resolving the lower-frequency components of the solution. This is the difficulty that multigrid is designed to address.

The Laplacian is the ideal operator for multigrid. For the Laplacian, various iterative methods (e.g., Jacobi, Gauss-Seidel) have a smoothing property: they quickly resolve the components of the solution that are highly oscillatory relative to the grid size, leaving a smooth residual. Moreover, in the Fourier basis (frequency) the Laplacian is diagonal. Thus computations on each grid (i.e., the determination of particular frequency components of the solution) are largely decoupled from the computations on other grids (frequencies). Also, the solutions on the coarse grids are good approximations to the smooth (low frequency) components of the fine-grid solutions. The combination of these properties produces the exceptional performance of multigrid on the Laplacian. Our optimization-based multigrid approach is based on the observation that a similar situation may obtain for many optimization problems governed by differential equations.

There are a variety of iterative methods available for large-scale optimization: truncated Newton methods, limited memory quasi-Newton methods, nonlinear conjugate gradient methods [23]. They will all, however, behave like the conjugate gradient method applied to a linear system. If the methods are implemented with an exact line search, they are equivalent to applying the conjugate gradient method to the minimization of the second-order Taylor's series approximation of the objective in (1.1):

$$\frac{1}{2} (p, \nabla^2 F p) + (\nabla F, p).$$

Thus, if the Hessians of our model problems are similar in nature to the Laplacian (or some other elliptic operator), we would expect a multigrid algorithm such as MG/Opt to work well, and a traditional large-scale optimization method applied only to the fine-grid problem to have difficulty.

To say more requires that we consider specific model problems, such as those described in the subsequent sections. Roughly speaking, we will show that the reduced Hessians for both the original analytic, infinite dimensional problem and for the discretized problem are well-suited for multigrid and ill-suited for the typical large-scale optimization approach. For our first two model problems, in fact, we show that the Hessians are elliptic differential operators. Thus, we would expect that MG/Opt would be an effective algorithm for solving these problems, and would perform better than either a traditional optimization approach.

Not all optimization problems for systems governed by differential equations have Hessians that resemble elliptic differential operators. A good example is provided by certain output least-squares inverse problems for diffusive processes. The Hessian of the objective in such problems is often a compact, smoothing operator; the Hessian damps the high-frequency components. On the other hand, such problems are typically highly underdetermined and unpreconditioned conjugate direction methods can often make good progress in reducing the objective.

**3.2. The dependence of the Hessian on the discretization.** In the following sections we analyze the properties of the Hessian using Fourier analysis. This is analogous to the approach used in analyzing multigrid methods for linear, constant coefficient PDEs.

However, there is a subtlety that arises in the context of optimization that does not arise in the multigrid solution of differential equations. When we apply multigrid to the Laplacian, for instance, we have direct access to its representation  $A_h$  on a fine grid, and can thus work with the coarser grid representation given by  $A_H = I_h^H A_h I_H^h$ . The situation is not so straightforward for (1.1)–(1.2) because of the indirect way the solution operator of the PDE and its discretization enter the calculation of the Hessian. The relationship between the Hessian and the solution operator of (1.2) lies in the structure of the Hessian of (1.1) and the reduced Hessian of the equivalent equality constrained problem (1.3), as we now discuss.

Implicit differentiation of (1.2) yields the following expression for the Jacobian of  $u(a)$  with respect to  $a$ :

$$\frac{du}{da} = -S_u^{-1} S_a,$$

where  $S_a$  and  $S_u$  are the derivatives of  $S$  with respect to  $a$  and  $u$ , respectively. The operator  $S_u^{-1}$  represents the solution operator of the linearized state equation. The Lagrangian of the equality constrained formulation (1.3) of our problem is

$$L(a, u; \lambda) = f(a, u) + (\lambda, S(a, u)).$$

Solutions to (1.1)–(1.2) will be stationary points of the Lagrangian.

The Hessian of the Lagrangian is

$$\nabla_{(a,u)}^2 L(a, u; \lambda) = \nabla_{(a,u)}^2 f(a, u) + \nabla_{(a,u)}^2 S(a, u) \lambda.$$

The Hessian of  $L$  with respect to both  $a$  and  $u$  has the block structure

$$\nabla^2 L = \begin{pmatrix} L_{aa} & L_{au} \\ L_{ua} & L_{uu} \end{pmatrix}.$$

We then have the following expression for the Hessian of  $F$  with respect to  $a$ :

$$(3.1) \quad \nabla^2 F = L_{aa} + L_{au} S_u^{-1} S_a + S_a^* S_u^{-*} L_{ua} + S_a^* S_u^{-*} L_{uu} S_u^{-1} S_a.$$

In this formula,  $*$  denotes the adjoint of an operator. This is the *reduced Hessian* [23] of  $f(a, u)$  with respect to the equality constraints  $S(a, u) = 0$  in (1.3).

In Section 4 we comment further on the significance of the structure of the reduced Hessian on the qualitative behavior of a multigrid algorithm such as MG/Opt. We also give additional arguments why we believe the results we present are representative of a large class of problems.

**3.3. Choice of Comparative Algorithms.** Our overall theme is that there are good reasons to apply an optimization-based multigrid algorithm to accelerate optimization algorithms applied to (1.1)–(1.2). In our computational tests, we compare MG/Opt to related optimization algorithms, but not to state-of-the-art multigrid software. In this subsection, we explain this decision.

Established multigrid software is designed to solve a system of equations, not an optimization problem. If no inequality constraints were present, it would be possible to convert our optimization problem to a system of equations by forming the KKT optimality conditions. However, the resulting system would not necessarily be anything like an elliptic operator, and considerable effort might be required to make multigrid work well.

For this reason, in our computational tests we compare MG/Opt with related algorithms that can be applied directly to the optimization problem. This also supports our overall goal of understanding how multigrid approaches can be useful in accelerating more traditional optimization algorithms. Our choice of algorithms is made so that the algorithms have as many components in common as possible. In this way, it is possible to isolate the effect of the multigrid strategy on the performance of the algorithms.

In the results we report here, the algorithm MG/Opt is based on a Matlab implementation of the optimization software TN [20]. It uses a truncated-Newton method, and is able to solve unconstrained and bound-constrained nonlinear optimization problems. In MG/Opt, the instruction “partially minimize” is interpreted as applying one outer iteration of TN. The instruction “if on coarsest grid, minimize” is interpreted as a call to TN with an upper limit of 25 outer iterations. (TN may terminate before 25 outer iterations if its convergence criteria are satisfied.) The line search used in MG/Opt is the same as the line search used in TN, and corresponds to the backtracking line search described in [23].

We believe that this implementation of MG/Opt is appropriate for illustrating our results. Our goal is to indicate that an optimization-based multigrid algorithm will be successful for problems of the form (1.1)–(1.2), and will perform better than more traditional approaches. In order to isolate the effect of MG/Opt from the effect of the nonlinear optimization algorithm, it is important to make MG/Opt as much like the more traditional approaches as possible. Thus, we are using the same software as the basis for all the algorithms. Further, there is considerable numerical evidence (see, e.g., [22]) that TN is an effective large-scale optimization algorithm.

In one of our model problems, we include bound constraints on the design variables. There is a subtle interaction between the template for MG/Opt and the active-set technique TN uses to manage the constraints. Because of the active-set strategy, it is necessary to project the multigrid search direction  $e_h$  onto the subspace defined by the set of non-binding constraints on the current grid before performing the line search. If the constraints were handled by a different technique (for example, via an interior-point method) the projection would be unnecessary. We have not examined such an algorithm, however.

Also, we mentioned previously the possibility of applying a multigrid algorithm to the constraints (1.2) alone. While possible, we did not include such an algorithm in the computational experiments. We omitted it because it would have limited applicability, i.e., it would only be useful when the constraints corresponded to a system of equations for which there was an effective multigrid scheme available. We believe that an optimization-based multigrid approach will have much broader applicability, and in fact have analyzed a model problem where the constraints are ill-suited to multigrid.

**4. A hyperbolic model problem.** Our first model problem is the initial-value problem (IVP) for the linear advection equation,

$$(4.1) \quad \begin{aligned} u_t + cu_x &= 0 \\ u(x, 0) &= a(x). \end{aligned}$$

The design variable is the initial condition  $a(x)$ . This equation is chosen as the simplest instance of a hyperbolic equation.

The objective we wish to minimize is

$$F(a) = \frac{1}{2} \iint_0^T [\alpha(u(x, t) - \phi(x, t))^2 + \beta(u_x(x, t) - \phi_x(x, t))^2] dx dt,$$

where  $\alpha, \beta$  are non-negative weights, and  $\phi$  is a prescribed target. Since  $u$  depends linearly on  $a$ , this is a linear least-squares problem; however, the evaluation of  $F$  involves the solution of the IVP (4.1).

This model problem is interesting because the forward problem—that of solving (4.1) and evaluating  $F(a)$ —is not particularly well-suited for textbook multigrid. Nevertheless, we will see that the optimization problem *is* well-suited for the application of multigrid.

**4.1. The analytical problem.** The solution operator for (4.1) is simply

$$a \mapsto u(x, t) = a(x - ct).$$

We then have

$$F(a) = \frac{1}{2} \iint_0^T [\alpha(a(x - ct) - \phi(x, t))^2 + \beta(a'(x - ct) - \phi_x(x, t))^2] dx dt.$$

Let  $\delta a$ ,  $\delta a_1$ , and  $\delta a_2$  be arbitrary functions which vanish at  $\pm\infty$ . Then

$$\begin{aligned} F'(a)\delta a &= \\ \iint_0^T &[\alpha(a(x - ct) - \phi(x, t))\delta a(x - ct) + \beta(a'(x - ct) - \phi_x(x, t))\delta a'(x - ct)] dx dt, \end{aligned}$$

and

$$\begin{aligned} F''(a)(\delta a_1, \delta a_2) &= \iint_0^T [\alpha\delta a_1(x - ct)\delta a_2(x - ct) + \beta\delta a_1'(x - ct)\delta a_2'(x - ct)] dt dx \\ &= \int_0^T \left\{ \int [\alpha\delta a_1(x - ct) - \beta\delta a_1''(x - ct)] \delta a_2(x - ct) dx \right\} dt \\ &= \int_0^T \left\{ \int [\alpha\delta a_1(x) - \beta\delta a_1''(x)] \delta a_2(x) dx \right\} dt \\ &= T \int [\alpha\delta a_1(x) - \beta\delta a_1''(x)] \delta a_2(x) dx. \end{aligned}$$

From the latter we see that

$$\nabla^2 F(a) \delta a_1 = T [\alpha \delta a_1(x) - \beta \delta a_1''(x)],$$

or

$$(4.2) \quad \nabla^2 F(a) = T \left[ \alpha I - \beta \frac{d^2}{dx^2} \right].$$

Since the governing PDE (4.1) is linear,  $F$  is quadratic, and so  $\nabla^2 F(a)$  exactly captures the nonlinearity of the objective.

As noted previously, we can see from the action of the Hessian why conjugate gradients will run into trouble. The Krylov vectors will be

$$\{-\nabla F(a), -(\nabla^2 F(a))\nabla F(a), -(\nabla^2 F(a))^2 \nabla F(a), \dots\}.$$

Since  $\nabla^2 F(a)$  is a differentiation operator, the Krylov vectors are increasingly oscillatory functions. This means conjugate gradients will struggle to make low-frequency corrections.

The fact that  $\nabla^2 F(a)$  looks like the textbook one-dimensional operator used to illustrate multigrid immediately suggests that a multigrid technique would work well. Moreover, conjugate gradients should do a good job correcting the high-frequency components on a given length scale.

It is not an accident that the Hessian turns out to be an elliptic operator, even though the governing equation is hyperbolic. This outcome follows from the structure of the Hessian, and the rules for composing pseudodifferential and Fourier integral operators. In terms of the operators of Section 3.2, the Hessian has the form  $S_a^* S_u^{-*} f_{uu} S_u^{-1} S_a$ . This has the form a pseudodifferential operator ( $f_{uu}$ ) conjugated by a Fourier operator ( $S_u^{-1} S_a$ ), which takes the initial condition to the solution of the hyperbolic IVP. One can apply Egorov's theorem [6, 29] to compute the symbol of the Hessian. Roughly speaking, the hyperbolic effect of the forward solution of the IVP, which propagates information along characteristics, is undone by the adjoint solution, which propagates information backward along the characteristics. The result is that the hyperbolic nature of the governing PDE is absent from the Hessian of the objective of the optimization problem. The Hessian ends up being elliptic, and thus amenable to multigrid.

We believe this to be a not uncommon situation. In a number of situations of interest, for instance, when  $a$  is a source term, the dependence on  $a$  is such that the Hessian simplifies to  $S_a^* S_u^{-*} f_{uu} S_u^{-1} S_a$ . In this case, the aforementioned effect of forward and back-propagation occurs, and the Hessian lacks the hyperbolic nature of the governing equation.

**4.2. The discretized problem.** The analysis of the discretized problem is more complicated than for the analytical problem. We examine the effect of the reduced Hessian in the frequency domain for the spatial variable, as in the von Neumann analysis of stability of difference schemes.

Let  $\Delta x$  be the (uniform) discretization in space, and  $\Delta t$  be the discretization in time. For a grid function  $v = \{v_m\}$  defined for all integers  $m$ , the Fourier transform is

$$\hat{v}(\omega) = \frac{1}{\sqrt{2\pi}} \sum_{m=-\infty}^{\infty} e^{-im\omega\Delta x} v_m \Delta x.$$

The associated Fourier inversion formula is

$$v_m = \frac{1}{\sqrt{2\pi}} \int_{-\pi/\Delta x}^{\pi/\Delta x} e^{im\omega\Delta x} \hat{v}(\omega) d\omega.$$

We denote by  $u_m^n$  the approximation to  $u(x_m, t_n)$ , where  $u$  is the solution of (4.1).

Suppose we apply a one-step finite difference scheme to solve the IVP (4.1), for which the amplification factor is  $g(\omega\Delta x)$ . Then the solution operator for the IVP is given by  $a \mapsto \{u_m^n\}$ , where

$$(4.3) \quad \hat{u}(\omega, t) = [g(\omega\Delta x)]^n \hat{a}(\omega).$$

In Section 4.3, we discuss the amplification factor for a specific finite-difference scheme.

Because the IVP is linear, the solution operator  $-S_u^{-1}S_a$  is just that of (4.1), the Jacobian of  $u$  with respect to  $a$ . We use Plancherel's theorem to compute its adjoint:

$$\begin{aligned} \left\langle \frac{du}{da} \delta a, w \right\rangle &= \sum_m \sum_n [g(\omega_m \Delta x)]^n \widehat{\delta a}(\omega_m) \overline{w_m^n} \Delta x \Delta t \\ &= \sum_m \widehat{\delta a}(\omega_m) \left( \sum_n [g(\omega_m \Delta x)]^n \overline{w_m^n} \Delta t \right) \Delta x \\ &= \left\langle \delta a, \frac{du^*}{da} w \right\rangle, \end{aligned}$$

from which we see that

$$\frac{du^*}{da} w = \sum_n \overline{[g(\omega_m \Delta x)]^n} w_m^n \Delta t.$$

In the objective function (1.1), we use forward differences to approximate  $\partial_x$ ; the associated symbol  $\theta$  is given by

$$\frac{v_{m+1} - v_m}{\Delta x}(\omega_m) = ie^{i\Delta x \frac{\omega}{2}} \frac{2 \sin \Delta x \frac{\omega}{2}}{\Delta x} \hat{v}(\omega_m) = \theta(\omega) \hat{v}(\omega_m).$$

The objective can then be written as

$$F(a) = f(a, u(a)) = \sum_m \sum_n \frac{1}{2} \sigma(\omega) \left| \hat{u}(\omega_m, t_n) - \hat{\phi}(\omega_m, t_n) \right|^2 \Delta x \Delta t,$$

where  $\sigma(\omega) = \alpha + \beta |\theta(\omega)|^2$ .

The action of the discretized Hessian  $\nabla^2 F(a)$  of the least-squares functional is then given by

$$\nabla^2 F(a) \delta a = \frac{du^*}{da} f_{uu} \frac{du}{da} \delta a.$$

From (4.3) we see that

$$\widehat{\delta u}(\omega_m, t_n) = [g(\omega_m \Delta x)]^n \hat{a}(\omega_m).$$

Since

$$f_{uu} \widehat{\delta u} = \sigma(\omega) \widehat{\delta u}(\omega_m, t_n),$$

we obtain

$$f_{uu} \frac{du}{da} \delta a = \sigma(\omega_m) [g(\omega_m \Delta x)]^n \hat{a}(\omega_m)$$

and

$$\frac{du^*}{da} f_{uu} \frac{du}{da} \delta a = \sum_n \overline{[g(\omega_m \Delta x)]^n} \sigma(\omega_m) [g(\omega_m \Delta x)]^n \hat{a}(\omega_m) \Delta t.$$

The symbol of the Hessian is thus

$$(4.4) \quad H(\omega) = \left( \sum_n |g(\omega \Delta x)|^{2n} \Delta t \right) \sigma(\omega).$$

From its symbol we would expect  $H$  to behave like an elliptic operator, and so multigrid would perform well. On the other hand, if a general-purpose optimization algorithm based on conjugate directions were applied to solve (1.1)–(1.2), then we would expect the behavior of the algorithm to be qualitatively similar to the unpreconditioned conjugate gradient method.

When solving the IVP (4.1) it is usually good practice to choose the discretization parameters so that  $|g(\omega \Delta x)|$  is close to one, to minimize dissipation and dispersion [25]. If this is done, the Hessian of the discretized problem, like the analytical Hessian (4.2), amplifies higher-frequency components more than lower-frequency ones. Thus, as for the analytic problem, we would expect methods related to the unpreconditioned conjugate gradient method to perform poorly.

It might be possible to accelerate a general-purpose optimization algorithm with a good preconditioner. In general, it will be difficult to identify *a priori* a good preconditioner for the reduced Hessian, because of its complicated structure (3.1). Separate preconditioners may be available for the Hessian of the objective (1.1), and for the Jacobian of the state constraints (1.2). These could be combined to form a preconditioner for the reduced Hessian [24], but we would not expect this preconditioner to perform anywhere near as well as multigrid applied directly to (1.1)–(1.2).

**4.3. Numerical results.** We apply the forward-time, backward-space scheme to (4.1):

$$(4.5) \quad \frac{u_m^{n+1} - u_m^n}{\Delta t} + c \frac{u_m^n - u_{m-1}^n}{\Delta x} = 0.$$

The associated amplification factor for this scheme is [25]

$$g(\omega \Delta x) = (1 - c\lambda) + c\lambda e^{-i\omega \Delta x},$$

where  $\lambda = \Delta t / \Delta x$ , and

$$|g(\omega \Delta x)|^2 = 1 - 4c\lambda(1 - c\lambda) \sin^2 \Delta x \frac{\omega}{2}.$$

This very simple scheme is only first-order accurate. However, its simplicity serves to make the analysis of the optimization problem clearer, and accuracy is not a concern here. Moreover, if we run this scheme at its stability limit  $c\lambda = 1$ , then the scheme is non-dissipative and non-dispersive, so we can ignore these effects in our numerical experiments.

For this choice, (4.4) becomes

$$H(\omega) = \sigma(\omega).$$

If  $\Delta x \ll \omega$  then  $H(\omega) \approx \alpha + \beta\omega^2$ , while if  $\omega$  is near the highest frequencies  $\pm\pi/\Delta x$ , then  $H(\omega) \approx \alpha + \beta\frac{2}{\pi}\omega^2$ . Thus, the Hessian amplifies higher frequency components.

This is illustrated in Figure 4.1, where we plot the magnitude of the FFTs of two Krylov vectors from our numerical tests. The figures show the FFTs of the first and fifth Krylov vectors from the first iteration of the optimization algorithm (we could have chosen successive Krylov vectors, but then the effect would have been less obvious). Both vectors have been normalized to have unit length. The first vector is plotted on the left and the fifth on the right. The plots make clear that the Hessian amplifies the high-frequency content of the vector it acts on. Thus, the successive Krylov subspaces become dominated by high-frequency (“roughening”) behavior, whereas the solution operator is a smoothing operator. This is counter-productive.

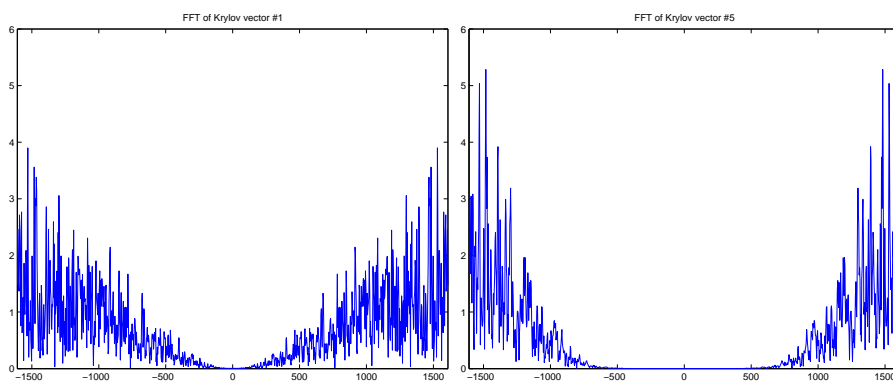


FIG. 4.1. *The magnitude of the FFTs of successive Krylov vectors.*

In the numerical example presented here, we choose  $c(x) \equiv 1$  and use (4.5). The space variable  $x$  is discretized uniformly in the interval  $[0, 2]$ , with  $x_0 = 0$  and  $x_n = 2$ . For the time step we choose  $\Delta t = \Delta x$  (at the stability limit of the difference scheme) and  $T = 1$  in the definition of the objective.

The target  $\phi(x, t)$  was chosen as the true solution of the IVP with initial condition  $a(x) = v_*(x)$  where

$$v_*(x) = v_0(x) \times x \times (1 - x)$$

and  $v_0$  was obtained from the Matlab commands

```
rand('state',0)
v0 = 5 + 0.10*rand(n,1)
I = find(x >= 0.95)
v0(I) = 0
```

The addition of random “noise” introduces significant high-frequency content to the problem, which makes the optimization problem more challenging. The solution  $v_*(x)$  was also cut off beyond  $x = 0.95$  to insure that its nonzero portion would remain inside

the computational domain for all  $t$ , so we would not need to worry about the effects of any artificial boundary.

In the objective function,  $\alpha = \beta = 1$ . The discretized objective has the form

$$f(a, u) = \frac{\Delta t}{2} \left[ \alpha \sum_{i=1}^n \sum_{j=1}^m r_{i,j}^2 + \beta \sum_{i=2}^n \sum_{j=1}^m d_{i,j}^2 \right]$$

where

$$r_{i,j} = u(x_i, t_j) - \phi(x_i, t_j), \quad d_{i,j} = \frac{r_{i,j} - r_{i-1,j}}{\Delta x}.$$

The gradient of the objective was computed via an adjoint computation.

The update operator  $I_H^h$  and the downdate operator  $I_h^H$  are chosen so that they are transposes of each other (up to a constant). It is perhaps easiest to define these operators by way of an example. Let

$$v_H = (V_1 \ V_2 \ V_3), \quad v_h = (v_1 \ v_2 \ v_3 \ v_4 \ v_5).$$

Then

$$I_H^h v_H = (V_1 \ \frac{1}{2}(V_1 + V_2) \ V_2 \ \frac{1}{2}(V_2 + V_3) \ V_3)$$

and

$$I_h^H v_h = \frac{1}{2} (v_1 + \frac{1}{2}v_2 \ \frac{1}{2}v_2 + v_3 + \frac{1}{2}v_4 \ \frac{1}{2}v_4 + v_5).$$

In Figure 4.2 and Table 4.1, we compare three algorithms. MG/Opt is as described above. “Optimization” uses TN (without the multigrid strategy) on the finest grid only. “Successive Refinement” is a simple multigrid-like strategy. It begins by applying TN on the coarsest grid. The solution is then updated to the next finer grid, and TN is called again. This continues until TN is called on the finest grid. Except on the finest grid, an upper limit of 25 outer iterations is imposed within TN.

		$n = 1025$	$n = 512$	$n = 257$	$n = 129$	$n = 65$	$n = 33$
Optimization	it	99					
	ls	100					
	cg	967					
Successive Refinement	it	99	25	25	25	25	19
	ls	100	26	26	26	26	20
	cg	956	216	225	214	220	145
MG/Opt	it	10	12	14	16	18	232
	ls	20	24	28	32	36	242
	cg	57	65	79	89	112	1974

TABLE 4.1  
Detailed results for the advection problem.

Figure 4.2 shows the improvement in the objective versus the computational cost; Table 4.1 summarizes the total amount of work depicted in the figure. The results show clearly that the multigrid algorithm MG/Opt is superior to pure optimization, as well as to the refinement technique. The multigrid algorithm shifts much of the

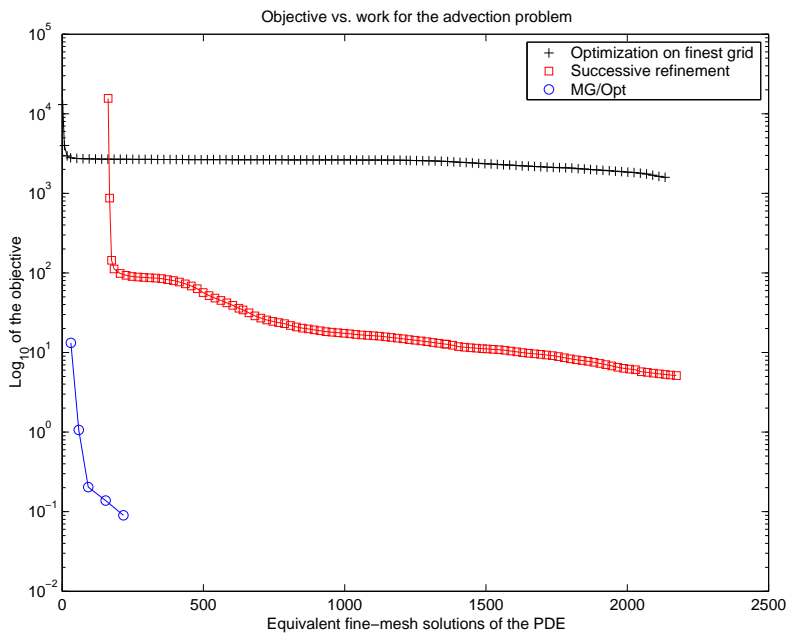


FIG. 4.2. Comparison of approaches for the advection problem.

computational effort to the coarsest grid. The other two algorithms expend most of their effort on the finest grid. The results provide strong confirmation for the earlier theoretical analysis of this model problem.

In Figure 4.2, the graphs for the three algorithms start at different places in the figure, suggesting that the three algorithms used three different starting points. This is not the case. All three algorithms were initialized with the same initial point on the finest grid. However, only Optimization uses this initial point directly. Both Successive Refinement and MG/Opt begin by downdating the initial point to the coarsest grid and use additional computations on this and other grids to eventually produce an estimate of the solution on the finest grid. Since it only makes sense to compare estimates of the solution on a single grid (the finest), of necessity the *graphs* for the three algorithms start at different points on the figure.

One other aspect of the figure deserves comment. It is traditional when reporting computational results for multigrid algorithms to report the norm of the residual (for a system of equations). In an optimization setting, however, it is more sensible to report the value of the objective function (or a merit function, if the constraints in the optimization problem are not satisfied by the estimates of the solution). As was discussed in Section 3.3, the comparison we are making is with other optimization algorithms, not with multigrid algorithms for systems of equations.

**5. An elliptic model problem.** The next model problem is the Dirichlet-to-Neumann map for the Laplacian on a rectangle in two dimensions. Let

$$\Omega = \{ (x_1, x_2) \mid 0 \leq x_1 \leq \pi, 0 \leq x_2 \leq 1 \}$$

and let  $\Gamma = \{ (x_1, 0) \mid 0 \leq x_1 \leq \pi \}$ ;  $\Gamma$  is the bottom portion of the boundary of  $\Omega$ . The governing equation is

$$\begin{aligned}\Delta u(x_1, x_2) &= 0 && \text{in } \Omega \\ u(x_1, x_2) &= 0 && \text{on } \partial\Omega \setminus \Gamma \\ u(x_1, 0) &= a(x_1)\end{aligned}$$

The objective we wish to minimize is

$$F(a) = \frac{1}{2} \int_0^\pi \left( \frac{\partial u}{\partial x_2}(x_1, 0) - \phi(x_1) \right)^2 dx_1,$$

where  $\phi$  is a prescribed target for the normal derivative along  $\Gamma$ . In steady-state heat conduction,  $a$  would represent the temperature distribution along the lower edge of the square. The objective measures the  $L^2$  discrepancy between the resulting heat flux and  $\phi$ .

**5.1. The analytical problem.** The Dirichlet-to-Neumann map and its Fourier analysis have been widely studied (e.g., see [26] in connection with electrical impedance tomography). The analysis given here relies on Fourier series. We compute  $F(a)$  directly by constructing  $u$  by separation of variables. Suppose  $a(x_1)$  has the Fourier sine series

$$a(x_1) = \sum_{k=1}^{\infty} a_k \sin kx_1.$$

Then

$$\begin{aligned}u(x_1, x_2) &= \sum_{k=1}^{\infty} a_k \sin kx_1 \left( -\frac{e^{-k}}{e^k - e^{-k}} e^{kx_2} + \frac{e^k}{e^k - e^{-k}} e^{-kx_2} \right) \\ \partial_{x_2} u(x_1, x_2) &= \sum_{k=1}^{\infty} a_k \sin kx_1 \left( -k \frac{e^{-k}}{e^k - e^{-k}} e^{kx_2} - k \frac{e^k}{e^k - e^{-k}} e^{-kx_2} \right) \\ \partial_{x_2} u(x_1, 0) &= \sum_{k=1}^{\infty} -k \coth k a_k \sin kx_1.\end{aligned}$$

From this we can compute the action of the Hessian of  $F$ . Let  $v(x_1) = \sum_{k=1}^{\infty} v_k \sin kx_1$ . Then the Jacobian  $J$  of the map  $a \mapsto \partial_{x_2} u(x_1, 0)$  is given by

$$Jv = \sum_{k=1}^{\infty} -k \coth k v_k \sin kx_1.$$

We compute the adjoint of  $J$  via Plancherel's theorem. If  $w(x_1) = \sum_{k=1}^{\infty} w_k \sin kx_1$ , then

$$(Jv, w)_{L^2} = \sum_{k=1}^{\infty} -k \coth k v_k w_k = (v, J^* w)_{L^2},$$

so

$$J^* w = \sum_{k=1}^{\infty} -k \coth k w_k \sin kx_1 = Jw.$$

$J$  is self-adjoint, which makes sense since it is a real-valued Fourier multiplier.

The action of the Hessian is thus given by the normal operator

$$\nabla^2 F(a) v = J^* J v = \sum_{k=1}^{\infty} k^2 \coth^2 k v_k \sin kx_1.$$

For large  $k$  we have  $k^2 \coth^2 k \approx k^2$ . The amplification by  $k^2$  means that the Hessian resembles a second derivative. Thus,  $\nabla^2 F(a)$  amplifies high-frequency components, so we would expect conjugate gradients applied purely to the fine-grid problem to converge slowly, as discussed in Section 3.1. On the other hand, the Hessian does not mix frequencies. This is the same behavior that we have seen in our earlier example, and (as before) it suggests that a multigrid approach will be successful.

**5.2. The discretized problem.** We apply the standard five-point finite difference stencil for the Laplacian with the same grid size  $h$  in both the  $x_1$  and  $x_2$  directions:

$$(5.1) \quad \frac{-u_{m+1,n} + 2u_{m,n} - u_{m-1,n}}{h^2} + \frac{-u_{m,n+1} + 2u_{m,n} - u_{m,n-1}}{h^2} = 0.$$

We assume that  $h = \pi/N$  for some integer  $N$ . Again we construct a solution via separation of variables, starting with the following identities for central differences:

$$(5.2) \quad -\sin k(m+1)h + 2\sin kmh - \sin k(m-1)h = 4\sin^2 k \frac{h}{2} \sin kmh$$

$$(5.3) \quad -e^{\alpha(n+1)h} + 2e^{\alpha nh} - e^{\alpha(n-1)h} = -4\sinh^2 \alpha \frac{h}{2} e^{\alpha nh}.$$

For the boundary condition  $a(x_1) = \sin kx_1$ , we seek solutions of the form

$$u_{m,n} = \sin kmh (b_k e^{\alpha_k nh} + c_k e^{-\alpha_k nh}).$$

Applying (5.1) to  $u_{m,n}$  and using (5.2)–(5.3), we obtain

$$\sin^2 k \frac{h}{2} \sin kmh (b_k e^{\alpha_k nh} + c_k e^{-\alpha_k nh}) - \sinh^2 \alpha_k \frac{h}{2} \sin kmh (b_k e^{\alpha_k nh} + c_k e^{-\alpha_k nh}) = 0,$$

or

$$\sinh^2 \alpha_k \frac{h}{2} = \sin^2 k \frac{h}{2}.$$

This yields

$$\alpha_k = \frac{2}{h} \sinh^{-1} \left( \sin k \frac{h}{2} \right).$$

From the boundary conditions along  $x_2 = 0$  and  $x_2 = 1$  we obtain

$$\begin{aligned} x_2 = 0 : \quad b_k + c_k &= 1 \\ x_2 = 1 : \quad b_k e^{\alpha_k} + c_k e^{-\alpha_k} &= 0, \end{aligned}$$

so

$$b_k = -\frac{e^{-\alpha_k}}{e^{\alpha_k} - e^{-\alpha_k}}, \quad c_k = \frac{e^{\alpha_k}}{e^{\alpha_k} - e^{-\alpha_k}}.$$

If  $a(x_1) = \sum_{k=1}^N a_k \sin kx_1$ , then the solution of the discretized BVP is

$$u_{m,n} = \sum_{k=1}^N a_k \sin kmh (b_k e^{\alpha_k nh} + c_k e^{-\alpha_k nh}).$$

Then  $\delta_{x_2} u$ , the discrete approximation to  $\partial_{x_2} u(x_1, 0)$ , is

$$(\delta_{x_2} u)_{m,0} = \frac{u_{m,1} - u_{m,0}}{h} = \sum_{k=1}^N a_k \sin kmh \frac{b_k (e^{\alpha_k h} - 1) + c_k (e^{-\alpha_k h} - 1)}{h}.$$

We write this as

$$(\delta_{x_2} u)_{m,0} = \sum_{k=1}^N \sigma_k a_k \sin kmh.$$

To compute the adjoint of this map, let  $w = \sum_{k=1}^N w_k \sin kmh$ . Then

$$\langle w, Jv \rangle = \sum_{k=1}^N w_k \sigma_k v_k,$$

so

$$(J^* w)_m = \sum_{k=1}^N \sigma_k w_k \sin kmh.$$

The Hessian of the discretized problem is then

$$(Hv)_m = (J^* Jv)_m = \sum_{k=1}^N \sigma_k^2 v_k \sin kmh.$$

A representative plot of  $\sigma_k^2$  against wavenumber  $k$  is given in Figure 5.1. From the series expansions

$$\begin{aligned} \sinh^{-1} u &= u - \frac{1}{6} u^3 + \text{h.o.t.} \\ \sin k \frac{h}{2} &= k \frac{h}{2} - \frac{1}{6} \left( k \frac{h}{2} \right)^3 + \text{h.o.t.}, \end{aligned}$$

where h.o.t. represents higher order terms, we obtain

$$\begin{aligned} \alpha_k(h) &= \frac{2}{h} \left[ \left( k \frac{h}{2} - \frac{1}{6} \left( k \frac{h}{2} \right)^3 + \text{h.o.t.} \right) - \frac{1}{6} \left( k \frac{h}{2} - \frac{1}{6} \left( k \frac{h}{2} \right)^3 + \text{h.o.t.} \right)^3 \right] \\ &= \frac{2}{h} \left[ \left( k \frac{h}{2} - \frac{1}{3} \left( k \frac{h}{2} \right)^3 + \text{h.o.t.} \right) \right] \\ &= k - \frac{2}{3} k^3 h^2 + \text{h.o.t.} \end{aligned}$$

It follows that for  $h \ll k$ ,  $\sigma_k^2 \approx k^2 \coth^2 k$ . If  $h$  is larger relative to  $k$ ,  $\sigma_k^2$  still grows roughly like  $k^2$ . As in the continuous case, the Hessian of the discretized problem

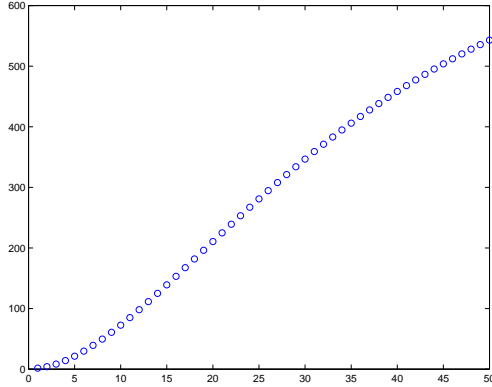


FIG. 5.1. A plot of  $\sigma_k^2$  against  $k$  for  $h = \pi/100$ .

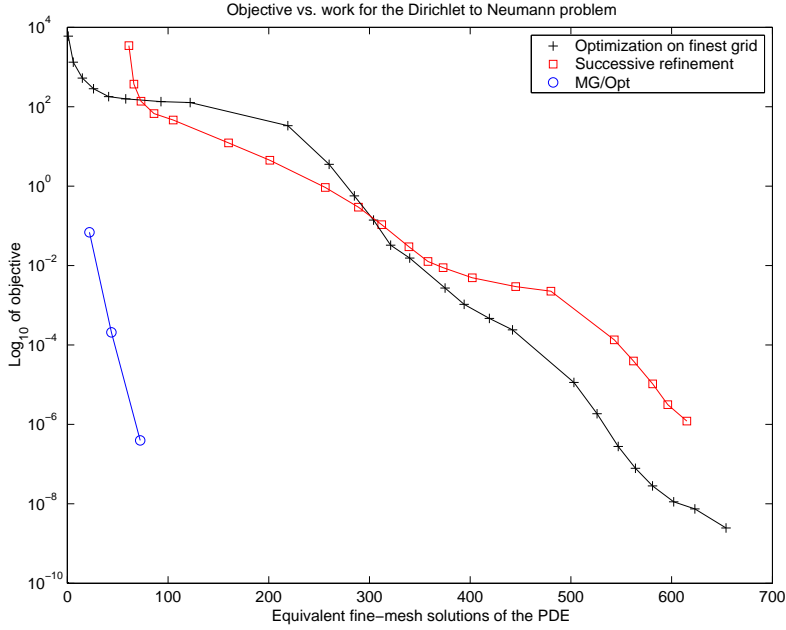


FIG. 5.2. Comparison of approaches for the Dirichlet-to-Neumann map.

amplifies high-frequency components. We would therefore expect conjugate gradients applied purely to the fine-grid problem to converge slowly. On the other hand, the Hessian looks like an elliptic operator, so we would expect multigrid to be effective.

As for the advection model problem, we compare MG/Opt with a truncated Newton algorithm applied only to the finest-mesh problem and with successive refinement. The results are summarized in Figure 5.2 and Table 5.1. Of the three, MG/Opt is again the most efficient, confirming our theoretical analyses.

**6. Nonlinear advection–diffusion.** The governing equations for the previous two model problems are linear, and the problems can be fully analyzed at both the analytical and numerical levels. We describe here a nonlinear model problem that represents a more realistic test of our optimization-based multigrid approach.

		$n = 129$	$n = 65$	$n = 33$	$n = 17$
Optimization	it	20			
	ls	21			
	cg	263			
Successive Refinement	it	20	22	16	10
	ls	21	26	17	11
	cg	267	186	87	28
MG/Opt	it	6	8	10	17
	ls	12	16	20	23
	cg	19	22	35	44

TABLE 5.1  
Detailed results for the Dirichlet-to-Neumann map.

The governing equation for the model problem is the viscous Burgers equation in one spatial dimension

$$\begin{aligned} u_t - \left(\frac{1}{2}u^2\right)_x - \nu u_{xx} &= 0 & (x, t) \in [0, L] \times [0, T] \\ u(x, 0) &= a(x) \\ u(0, t) = u(L, t) &= 0 \end{aligned}$$

The design variable is the initial condition  $a(x)$ . The objective we wish to minimize is

$$\frac{1}{2} \int_0^L (u(x, T) - \phi(x))^2 dx,$$

where  $\phi$  is a prescribed target.

**6.1. Numerical results.** We solve (6.1) using the original MacCormack method [18, 28]. Let  $r = \nu\Delta t/(\Delta x)^2$ ,  $\psi(u) = \frac{1}{2}u^2$ , and denote  $u_j^n = u(x_j, t_n)$ , etc. Then the predictor step is

$$u_j^{\overline{n+1}} = u_j^n + \frac{\Delta t}{\Delta x} (\psi_{j+1}^n - \psi_j^n) + \nu(u_{j+1}^n - 2u_j^n + u_{j-1}^n)$$

and the corrector step is

$$u_j^{n+1} = \frac{1}{2} \left[ u_j^n + u_j^{\overline{n+1}} - \frac{\Delta t}{\Delta x} (\psi_j^{\overline{n+1}} - \psi_{j-1}^{\overline{n+1}}) + \nu (u_{j+1}^{\overline{n+1}} - 2u_j^{\overline{n+1}} + u_{j-1}^{\overline{n+1}}) \right].$$

We use a uniform discretization in both time and space. The time discretization is determined by the stability condition

$$\Delta t \leq \frac{(\Delta x)^2}{A\Delta x + 2\nu}$$

where

$$A \geq \max_{x,t} |u(x, t)|.$$

In our numerical tests, we use the parameters  $L = \pi$ ,  $T = 1.4$ , and  $\nu = 5 \times 10^{-4}$ . The initial estimate of the solution is  $a(x) \equiv 0$ . The target in the objective is  $\phi(x) =$

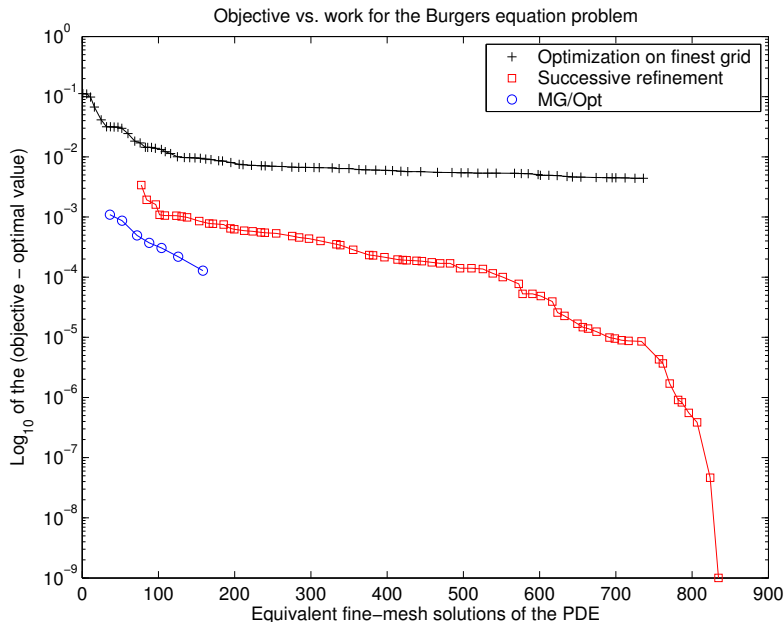


FIG. 6.1. Comparison of approaches for Burgers equation.

$5(x - 1.5)^2(2.5 - x)$  for  $1.5 \leq x \leq 2.5$  and 0 otherwise. This choice of  $\phi$  leads to a least-squares problem with a non-zero residual at the solution. We also choose  $L = 3$ ,  $T = 1.4$ , and  $\nu = 5 \times 10^{-4}$ .

As for the other model problems, we compare MG/Opt with a truncated Newton algorithm applied to the finest-mesh problem and with successive refinement. The choice of numerical schemes means that the cost of solving the nonlinear viscous Burgers' equation (in order to compute  $F(a)$  and its derivatives) and the cost of solving the linearized viscous Burgers' equation (to compute adjoint states and apply the Hessian) are approximately the same.

The results are summarized in Figure 6.1 and Table 6.1. Figure 6.1 plots the difference between the computed objective values and an “optimal” objective value. The latter was obtained by running the finest mesh stage of successive refinement until it satisfied a stringent stopping tolerance indicating that at least a local minimizer was found. In line with our expectations, MG/Opt gives the most cost-effective improvement in the objective. Visual comparison of the least-squares residual obtained by MG/Opt, and the “optimal” solution revealed no material difference in the computed solutions.

**6.2. Effect of the line search and auxiliary bound constraints.** As discussed in Section 2, the MG/Opt algorithm includes two safeguards to guarantee convergence, and to ensure that the algorithm makes progress toward the solution at every iteration. One is the line search applied to the multigrid search direction in the computation of  $a_{h,2} = a_{h,1} + \alpha e_h$ . The other is the set of bounds  $a_{H,low} \leq a_H \leq a_{H,up}$  that prevent the computation of too long a step based on the coarser grid calculations. Both safeguards are invoked in the solution of this model problem.

The line search took a non-default step (i.e., a step  $\alpha \neq 1$ ) 11 times, and occasionally with a small step (e.g.,  $\alpha = .0039$ ), indicating that the search direction from the

		$n = 513$	$n = 257$	$n = 129$	$n = 65$	$n = 33$
Optimization	it	76				
	ls	100				
	cg	318				
Successive Refinement	it	66	17	25	13	12
	ls	74	25	30	19	17
	cg	342	100	123	75	44
MG/Opt	it	8	10	12	14	134
	ls	22	24	27	31	146
	cg	48	55	41	48	369

TABLE 6.1  
Detailed results for Burgers equation.

multigrid recursion produced a poorly-scaled search direction at some iterations and for some inter-grid transitions. In the remaining 11 uses of the multigrid line search, the default step ( $\alpha = 1$ ) was taken. In addition, the auxiliary bound constraints were active at the solution of 18 of the 52 sub-problems.

On the other hand, when the line search and the move-limiting bound constraints were turned off in the algorithm, the algorithm did not converge to a solution. At the majority of the iterations, the value of the objective function *increased* when a step was taken. This is in sharp contrast to the behavior of the safeguarded algorithm.

**7. Connections to other optimization methods.** Our discussion so far has placed MG/Opt in the context of multigrid algorithms. MG/Opt also has connections with more traditional optimization methods [23], and with the approaches used in model management [2]. These connections can be used to explain some of the good computational properties of MG/Opt.

We begin by making some further connections between the fine-grid and coarse-grid problems. In particular, the coarse-grid subproblems are first-order approximations to the fine-grid subproblems. This statement is made more precise in the following lemma.

LEMMA 7.1. *In algorithm MG/Opt, define*

$$\begin{aligned} g_{h,1} &\equiv \nabla F_h(a_{h,1}) \\ g_{H,1} &\equiv \nabla F_H(a_{H,1}) - v_H. \end{aligned}$$

*These are the gradients of the fine-grid and coarse-grid problems at  $a_{h,1}$  and  $a_{H,1}$ , respectively. Then*

$$I_h^H g_h(a_{h,1}) = g_H(a_{H,1});$$

*i.e., the gradient of the coarse-grid subproblem at the initial point  $a_{H,1}$  matches the downdated fine-grid gradient. Assume that the update and downdate operators in algorithm MG/Opt satisfy*

$$I_H^h = \gamma(I_h^H)^T$$

*for some constant  $\gamma$ , and consider a pair of directions related by*

$$p_H = I_h^H p_h.$$

Then

$$\begin{aligned} F_h(a_{h,1} + p_h) &= f_h(a_{h,1}) + p_h^T g_h(a_{h,1}) + O(\|p_h\|^2) \\ &= f_h(a_{h,1}) + \gamma p_H^T g_H(a_{h,1}) + O(\|p_H\|^2) \end{aligned}$$

*Proof.* These results are direct consequences of the formulas for MG/Opt.  $\square$

The final result in the lemma shows that the coarse-grid subproblem is (essentially) a first-order approximation to the fine-grid problem, and that it corresponds (essentially) to a Taylor series approximation to the fine-grid problem.

It would be possible to adjust MG/Opt so that the coarse-grid subproblem is precisely a first-order approximation to the fine-grid problem. This would require adding a constant to the objective function in the multigrid subproblem, and scaling the search direction  $e_h$  by  $\gamma$ . These are trivial adjustments, and would not change the behavior of MG/Opt.

When the number of design variables  $a$  is fixed, and only the discretization of the governing differential equations changes, MG/Opt is an instance of the model management approach [3]. Model management uses models of varying physical or numerical fidelity, rather than relying on a single accurate but presumably expensive model. Lower fidelity models are corrected to agree with higher fidelity models to first order. These corrections ensure that descent directions computed using the corrected lower fidelity models are descent directions for the higher fidelity models. The result is an approach based on non-quadratic approximations to the high fidelity model.

MG/Opt is also related to other optimization methods. The lemma shows that the coarse-grid subproblem is a first-order approximation to the fine-grid problem. This relates MG/Opt to the steepest descent method, where the search direction is derived from a first-order approximation.

In the steepest descent method, the search direction is a descent direction, but it is not typically “well scaled” in the sense that a step length of  $\alpha = 1$  is not guaranteed to be successful. Asymptotically, MG/Opt will produce well scaled search directions (see below) but this is not guaranteed at all iterations. Because of this concern, we include bound constraints in the subproblems in MG/Opt, since the first-order approximation might only be reliable in a small region surrounding the initial point  $a_{H,1}$ . This is closely related to trust-region approaches for optimization [23].

The following lemma can be used to show that, as MG/Opt converges, the search directions  $e_h$  in MG/Opt will be well scaled, i.e., a step of 1 along the search direction will be an approximate minimizer along the search direction.

LEMMA 7.2. *Define*

$$s(\alpha) \equiv F_h(a_{h,1} + \alpha e_h).$$

Then

$$\begin{aligned} s'(1) &= \gamma \left\{ e_H^T g_{H,1} + e_H^T \nabla^2 F_H(a_{H,1}) e_H \right. \\ &\quad \left. + e_H^T [I_h^H \nabla^2 F_h(a_{h,1}) I_h^h - \nabla^2 F_H(a_{H,1})] e_H \right\} + O(\|e_H\|^3), \end{aligned}$$

where  $I_h^h = \gamma(I_h^H)^T$ .

*Proof.* We have

$$s'(\alpha) = e_h^T \nabla F_h(a_{h,1}) + \alpha e_h^T \nabla^2 F_h(a_{h,1}) + \alpha^2 O(\|e_h\|^3).$$

If we set  $\alpha = 1$  and use the results of the previous lemma,

$$\begin{aligned}
s'(1) &= e_h^T \nabla F_h(a_{h,1}) + e_h^T \nabla^2 F_h(a_{h,1}) + O(\|e_h\|^3) \\
&= e_H^T (I_H^h)^T \nabla F_h(a_{h,1}) + e_H^T (I_H^h)^T \nabla^2 F_h(a_{h,1}) I_H^h e_H + O(\|I_H^h e_H\|^3) \\
&= \gamma \left\{ e_H^T g_{H,1} + e_H^T \nabla^2 F_H(a_{H,1}) e_H \right. \\
&\quad \left. + e_H^T [I_H^H \nabla^2 F_h(a_{h,1}) I_H^h - \nabla^2 F_H(a_{H,1})] e_H \right\} + O(\|e_H\|^3).
\end{aligned}$$

This is the desired result.  $\square$

If the step length is chosen as a solution  $\alpha^*$  of the one-dimensional problem

$$\underset{\alpha}{\text{minimize}} F_h(a_{h,1} + \alpha e_h),$$

then  $s'(\alpha^*) = 0$ . We can expect that the search direction will be well scaled if  $|s'(1)|$  is small.

In the formula for  $s'(1)$ , the first term is

$$e_H^T g_{H,1} + e_H^T \nabla^2 F_H(a_{H,1}) e_H.$$

If  $e_H$  is a Newton direction or a truncated-Newton direction for the coarse-grid problem, then this term will be zero [20]. If the coarse-grid problem is solved exactly, then this term will be  $O(\|e_H\|^3)$ . Thus, in a number of practical cases,

$$s'(1) = \gamma e_H^T [I_H^H \nabla^2 F_h(a_{h,1}) I_H^h - \nabla^2 F_H(a_{H,1})] e_H + O(\|e_H\|^3).$$

As MG/Opt converges, the norms of the search directions will go to zero, and the term  $O(\|e_H\|^3)$  will become negligible. Our analytical results for the model problems suggest that it is reasonable to assume that

$$[I_H^H \nabla^2 F_h(a_{h,1}) I_H^h] \approx \nabla^2 F_H(a_{H,1}).$$

and so  $s'(1)$  will be close to zero.

Thus, asymptotically, we can expect that  $s'(1)$  will be small, and that a step length of 1 will be accepted in the line search. This is confirmed by our computational tests. In both model problems, in every multigrid line search the step length of 1 was accepted.

In summary, the coarse-grid subproblems in MG/Opt are first-order approximations to the fine-grid problem. Hence, MG/Opt is related to the steepest descent method. However, unlike the steepest descent method, we can expect that MG/Opt will produce well scaled search directions as it converges. This is a property that is shared with Newton's method. This suggests that MG/Opt, in addition to having the strong theoretical guarantees of convergence associated with an optimization algorithm, will also have the rapid convergence properties associated with Newton's method.

**8. Conclusion.** We have presented a multigrid algorithm for solving optimization problems governed by differential equations. The algorithm is a nonlinear programming adaptation of the multigrid idea. In particular, the multigrid subproblems are nonlinear optimization problems, not systems of linear or nonlinear equations. This gives additional flexibility in modeling; in particular, it makes it possible to include inequality constraints in the optimization problem. It also makes it possible

to guarantee convergence of the algorithm to a stationary point of the optimization problem.

Multigrid algorithms have proven very successful in solving elliptic PDEs, but application to other classes of PDEs requires more work, and the results are sometimes mixed. Our results suggest that multigrid can be successful in the more general setting of optimization, even if the governing PDEs are not elliptic.

We have presented several forms of evidence to justify our claims. First, we have analyzed several model problems, and shown that the reduced Hessians of these model problems are well-suited for multigrid. Second, we have applied the multigrid algorithm to these model problems, and shown numerically that multigrid performs well, but a traditional optimization algorithm performs poorly. Third, we have demonstrated that the multigrid algorithm is closely related to model management techniques, and to both the steepest-descent method and Newton's method for optimization. These relationships give further explanation of the good performance of multigrid in an optimization setting.

Because optimization models are so general, it would be difficult to demonstrate that multigrid will always be suitable in an optimization setting. We believe, though, that the model problems described here are representative of broad categories of interesting problems. Nevertheless, additional research, both theoretical and computational, will be necessary to clarify the behavior and performance of multigrid techniques in an optimization setting.

#### REFERENCES

- [1] J. ABADIE, *Application of the GRG algorithm to optimal control problems*, in Integer and Nonlinear Programming, J. Abadie, ed., North-Holland Elsevier, 1970.
- [2] N. M. ALEXANDROV AND R. M. LEWIS, *A trust region framework for managing approximation models in engineering optimization*, in Proceedings of the Sixth AIAA/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, September 1996. AIAA papers 96-4101 and 96-4102.
- [3] N. M. ALEXANDROV, R. M. LEWIS, C. R. GUMBERT, L. L. GREEN, AND P. A. NEWMAN, *Approximation and model management in aerodynamic optimization with variable-fidelity models*, Journal of Aircraft, 38 (2001), pp. 1093-1101.
- [4] E. ARIAN AND S. TA'ASAN, *Analysis of the hessian for aerodynamics optimization: Inviscid flows*, Computers and Fluids, 28 (1999), pp. 853-877.
- [5] A. BRANDT, *Multi-level adaptive solutions to boundary-value problems*, Mathematics of Computation, 31 (1977), pp. 333-390.
- [6] Y. V. EGOROV, *Linear differential equations of principal type*, Kluwer Academic/Plenum Publishers, New York, 1986.
- [7] D. FENG AND T. H. PULLIAM, *Aerodynamic design optimization via reduced Hessian SQP with solution refining*, Tech. Report 95-24, Research Institute for Advanced Computer Science (RIACS), NASA Ames Research Center, 1995.
- [8] ———, *An all-at-once reduced Hessian SQP scheme for aerodynamic design optimization*, Tech. Report 95-19, Research Institute for Advanced Computer Science (RIACS), NASA Ames Research Center, 1995.
- [9] E. GELMAN AND J. MANDEL, *On multilevel iterative methods for optimization problems*, Mathematical Programming, 48 (1990), pp. 19-39.
- [10] O. GHATTAS AND C. OROZCO, *A parallel reduced Hessian SQP method for shape optimization*, in Multidisciplinary Design Optimization: State of the Art, N. Alexandrov and M. Y. Hussaini, eds., Philadelphia, 1997, SIAM, pp. 133-152.
- [11] W. HACKBUSCH, *Multi-grid Methods and Applications*, vol. 4 of Springer series in Computational Mathematics, Springer-Verlag, Berlin, 1985.
- [12] R. T. HAFTKA, Z. GÜRDAL, AND M. P. KAMAT, *Elements of structural optimization*, Kluwer Academic Publishers, Dordrecht, 1990.
- [13] W. H. JOU, W. P. HUFFMAN, D. P. YOUNG, R. G. MELVIN, M. B. BIETERMAN, C. L. HILMES, AND F. T. JOHNSON, *Practical considerations in aerodynamic design optimization*, in 12th

- AIAA Computational Fluid Dynamics Conference, American Institute of Aeronautics and Astronautics, 1995, pp. 950–960. AIAA paper 95-1730-CP, presented at the 12th AIAA Computational Fluid Dynamics Conference, June 19–22, 1995, San Diego, California.
- [14] G. KURUVILA, S. TA'ASAN, AND M. D. SALAS, *Airfoil design and optimization by the one-shot method*. AIAA paper 95-0478, 1995.
  - [15] N. KYDES, *A Multigrid Solution of the Continuous Dynamic Disequilibrium Network Design Problem*, PhD thesis, School of Information Technology and Engineering, George Mason University, Fairfax, Virginia, 2002.
  - [16] R. M. LEWIS, *Practical aspects of variable reduction formulations and reduced basis algorithms in multidisciplinary design optimization*, in *Multidisciplinary Design Optimization: State of the Art*, N. Alexandrov and M. Y. Hussaini, eds., Philadelphia, 1997, SIAM, pp. 172–188.
  - [17] R. M. LEWIS AND S. G. NASH, *A multigrid approach to the optimization of systems governed by differential equations*. AIAA Paper 2000-4890, 2000.
  - [18] R. W. MACCORMACK, *The effect of viscosity in hypervelocity impact cratering*. AIAA Paper 69-354, 1969.
  - [19] S. F. MCCORMICK, *Multilevel Adaptive Methods for Partial Differential Equations*, Society for Industrial and Applied Mathematics, 1989.
  - [20] S. G. NASH, *Preconditioning of truncated-Newton methods*, *SIAM Journal on Scientific and Statistical Computing*, 6 (1985), pp. 599–616.
  - [21] ———, *A multigrid approach to discretized optimization problems*, *Journal of Computational and Applied Mathematics*, 14 (2000), pp. 99–116.
  - [22] S. G. NASH AND J. NOCEDAL, *A numerical study of the limited memory BFGS method and the truncated-Newton method for large scale optimization*, *SIAM Journal on Optimization*, 1 (1991), pp. 358–372.
  - [23] S. G. NASH AND A. SOFER, *Linear and Nonlinear Programming*, McGraw–Hill, New York, 1996.
  - [24] S. G. NASH AND A. SOFER, *Perconditioning reduced matrices*, *SIAM Journal on Matrix Analysis and Applications*, 17 (1996), pp. 47–68.
  - [25] J. C. STRIKWERDA, *Finite Difference Schemes and Partial Differential Equations*, Wadsworth & Brooks/Cole, 1989.
  - [26] J. SYLVESTER AND G. UHLMANN, *A global uniqueness theorem for an inverse boundary value problem*, *Annals of Mathematics*, 125 (1987), pp. 153–169.
  - [27] S. TA'ASAN, *“One Shot” methods for optimal control of distributed parameter systems I: Finite dimensional control*, Tech. Report 91-2, Institute for Computer Applications in Science and Engineering, January 1991.
  - [28] J. C. TANNEHILL, D. A. ANDERSON, AND R. H. PLETCHER, *Computational Fluid Dynamics and Heat Transfer*, Taylor & Francis, 1997.
  - [29] M. E. TAYLOR, *Pseudodifferential operators*, Princeton University Press, Princeton, New Jersey, 1981.