AUTOMATICALLY ASSESSING THE PERFORMANCE OF AN OPTIMIZATION-BASED MULTIGRID METHOD

STEPHEN G. NASH* AND ROBERT MICHAEL LEWIS†

Abstract. Many large nonlinear optimization problems are based upon discretizations of underlying function spaces. Optimization-based multigrid methods—that is, multigrid methods based on solving coarser versions of an optimization problem—are designed to solve such discretized problems efficiently by taking explicit advantage of the family of discretizations. The methods are generalizations of more traditional multigrid methods for solving partial differential equations. These multigrid methods are a powerful tool, but they are not appropriate for all optimization problems. We discuss techniques whereby the multigrid method can assess the properties of the optimization problem, with the goal of automatically determining whether the optimization problem is well suited for the multigrid-type algorithm.

Key words. Multigrid methods, optimization of systems governed by differential equations, PDE-constrained optimization

1. Introduction. A multigrid algorithm can be a powerful tool for solving discretized optimization problems, i.e., problems where the variables represent the discretization of some underlying function [1, 14]. However, multigrid algorithms are not well suited to all optimization problems, and it can be difficult to predict whether multigrid will work effectively on a particular optimization problem. The goal of this paper is to develop diagnostic tests that can assess the performance of multigrid as the optimization problem is being solved, and identify whether the optimization problem has the necessary properties.

The multigrid algorithm uses coarse versions of the optimization problem to improve the estimates of the solution on finer grids. Since the computations on the coarse problems are usually much less costly, the multigrid algorithm may be able to compute a solution far more rapidly than a traditional optimization algorithm applied directly to the fine-grid problem.

Multigrid algorithms were first developed to solve elliptic partial differential equations. In that setting, the differential operator is stated explicitly and its properties can be studied directly. In contrast, we will be solving optimization problems with an *optimization-based multigrid method*, where on each grid an optimization problem is solved, not a system of equations. For the constrained optimization problems that we consider, the corresponding properties of the relevant operator will be unknowable except in the simplest of cases. It may be difficult or impossible to deduce even qualitative information about the operator in advance of solving the problem. Since multigrid algorithms are not general-purpose optimization methods, they are not guaranteed to work effectively on all problems. So the question arises: How do you determine if it is appropriate to use a multigrid algorithm?

A simplistic approach would be to apply the multigrid algorithm and see if it works. This might be appropriate if a single property of the optimization problem would be sufficient to guarantee good performance. But there are at least four such properties:

- degree of nonlinearity of the optimization problems

*Systems Engineering and Operations Research Department, Mail Stop 4A6, George Mason University, Fairfax, VA 22030, snash@gmu.edu.

†Department of Mathematics, College of William & Mary, P.O. Box 8795, Williamsburg, Virginia, 23187-8795, buckaroo@math.wm.edu.

- consistency of the optimization problems across grids
- complementarity of the grids
- separability of the reduced Hessians of the optimization problems along frequency scales

These properties can be intertwined. For example, the reduced Hessian of an optimization problem might share some of the spectral characteristics of the Laplacian (an ideal operator for multigrid), but convergence could be slow because of nonlinearities. In addition, the optimization problems could change character depending on the values of the design variables.

Our aim is to demonstrate that a particular optimization-based multigrid algorithm [14] is capable of assessing many relevant properties of the optimization problem, either as a by-product of finding the solution or with low-cost auxiliary computations. Moreover, these diagnostic tests rely on information that can be readily extracted during the normal course of the algorithm at little additional cost. A multigrid algorithm including these tests would be capable of solving challenging discretized optimization problems or of suggesting the reasons why it failed to do so. It would not be perfect—there would be problems that would confound tests as simple as those we describe—but it would greatly expand the class of problems that could be handled automatically and provide a level of "self awareness" rare among optimization algorithms. In addition, the diagnostic tests would provide guidance if the user wished to investigate further, using more sophisticated tests fine-tuned to the specific optimization problem.

The multigrid algorithm would not always work well, and would not always be appropriate. But if the algorithm had the sort of "self awareness" we describe, it might be possible to identify the reasons why multigrid were not appropriate, and to identify an alternative algorithm as necessary. If this could be automated then it would be possible to develop an adaptive optimization environment, where the adaptation would be based on using a variety of optimization algorithms, and not on having a single algorithm adapt to a particular problem (as in adaptive quadrature or mesh refinement). With this strategy, there would be occasions when the optimization took a long time (such as cases where only an algorithm as general as pattern search [11] could be used) but for most problems more effective algorithms would be identified. Hence, on average, optimization would be efficient.

Here is an outline of the paper. Section 2 gives background material on the optimization problems and the multigrid algorithm. Section 3 discusses the diagnostic test. Section 4 demonstrates why the tests are sufficient to assess the performance of the multigrid algorithm. Section 5 describes computational experiments. Conclusions are in Section 6.

The paper builds on a variety of research results. It owes a great debt to the vast research on multigrid methods (e.g., [2, 17]). More specifically, the diagnostic tools for assessing the coarse grids arise from algebraic multigrid methods (e.g., [4]). However, those results cannot be used directly. Algebraic multigrid is typically used to solve linear equations, and assumes that the coefficient matrix is sparse and that its entries are known. This is not true for the optimization-based multigrid methods that we describe.

Related diagnostic tests have been developed for detecting nonlinearity in the context of truncated-Newton methods [8]. Various approaches have been considered for testing convexity (see, e.g., [5, 23]). Finally, analogous tests have been developed for other problem classes, such as branch-and-bound methods for integer programming

[7] and iterative solvers for linear equations [10, 15].

2. Background. Much of our discussion applies to a general optimization problem

$$\underset{a_h}{\text{minimize}}\, F_h(a_h)$$

where the objective function F_h and the variables a_h represent a family of problems of varying fidelity or complexity. The subscript h might represent a discretization, but other choices are possible. However, our prior work and our intuition are based on a more specific form of optimization problem

$$\underset{a}{\text{minimize}} \quad F(a) = f(a, u(a)), \tag{2.1}$$

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

$$S(a, u(a)) = 0 \tag{2.2}$$

in a and u. Here we assume that $S(a, u) = 0$ is solved for u given a.

While we assume that (2.2) is a differential equation, our approach applies to problems with other types of governing equations such as integral equations. For simplicity, we assume that there are no other constraints on the variables a and u, although this is not essential. The presence of additional constraints, including inequalities, makes the application of multigrid to optimization a true generalization of the multigrid approach [14].

The problem (2.1)–(2.2) represents a family of optimization problems, each corresponding to a particular discretization of the differential equation (2.2). We will solve the problem using the multigrid algorithm MG/Opt described below.

We will regularly use terms such as "grid", "coarse", "fine", "frequency", etc. This is a convenient shorthand used to refer to the various sub-problems, that is, instances of the overall optimization problem. The multigrid algorithm has only limited knowledge of the optimization problem and the meaning of the variables. In particular, when we use our algorithm MG/Opt with the TN truncated-Newton method as the underlying optimization algorithm, the software has the following information available:

- A procedure to compute the objective function and its gradient for given values of the design variables on any of the sub-problems.
- The dimensions of the vectors of design variables on the various sub-problems.
- Update and downdate procedures to transform vectors from one sub-problem to another.

MG/Opt does not know what the various sub-problems represent, nor does it know what the design variables represent.

For convenience, we will describe the sub-problems as being representations of a particular optimization problem on a given "grid" or "mesh". The downdate procedure will move from a "fine grid" to a "coarse grid", and the update procedure will do the reverse. We will describe the fine grid as representing a "shorter length scale" capable of representing the solution at "higher frequencies". And so on.

All of this is metaphorical. For example, the various sub-problems could correspond to a network with various degrees of aggregation. (This is appropriate for

certain VLSI problems [13].) We use this language because it is common when discussing multigrid methods for PDEs, and it is especially appropriate when considering a problem with its reduced Hessian equal to a discretized Laplacian or some similar operator.

It is, however, just a convenience. Our diagnostic tests do not rely on this interpretation. For example, they do not use Fourier transforms to get frequency information. The diagnostic tests only use information normally provided to MG/Opt.

We assume throughout that the user-provided information is correct. More specifically, we assume that the software to evaluate the objective function, the constraints, and their first derivatives, has been correctly programmed, and that the update and downdate operators are correct and effective (i.e., that the output is a good approximation to the input from the other grid). Our diagnostic tests are intended to augment basic error checking done by many software systems.

2.1. The Multigrid Algorithm. The algorithm MG/Opt given here is a realization of the more general algorithmic framework in [12], and corresponds to the algorithm used in the computational tests in that paper. We apply MG/Opt to the problem (2.1)–(2.2), and assume that the state equation (2.2) is solved for u given a.

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

$$I_H^h = C_I \times \left(I_h^H\right)^T .$$

for some constant C_I.

The algorithm is based on the truncated-Newton optimization algorithm TN [21], which in turn uses a conjugate-gradient (CG) algorithm to compute a search direction.

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 $a_{h,1}$. Downdate 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:

    $$\underset{a_H}{\text{minimize}}\, \tilde{F}(a_H) \equiv F_H(a_H) - v_H^T a_H$$

    subject to the bound constraints

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

    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 is to use a full multigrid initialization scheme (see, for example, [18]).

Within MG/Opt, we will interpret "partially minimize" to mean that at least one iteration of the TN optimization algorithm is applied to the optimization problem. If the state equation (2.2) is solved for u given a then, under appropriate assumptions, MG/Opt is guaranteed to converge to a stationary point of the optimization problem [20]. Also, the search direction e_h at each iteration will be a descent direction, ensuring that the estimate of the solution improves at every iteration of the multigrid algorithm. The line search used to obtain $a_{h,2} = a_{h,1} + \alpha e_h$ is the same as the line search in algorithm TN.

The recursion requires MG/Opt to solve a shifted version of the optimization problem subject to the bound constraints. The bounds used are the same as those in [12], namely

$$a_{H,low} = a_{H,1} - \gamma e$$
$$a_{H,up} = a_{H,1} + \gamma e$$

where

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

The shifted problem on the coarse grid is a first-order approximation to the optimization problem on the fine grid, and thus will only be accurate in a neighborhood of the point $a_{H,1}$. The bounds restrict the algorithm to such a neighborhood, analogous to a trust-region approach for optimization [6].

2.2. The Reduced Hessian. When multigrid is applied to a PDE, its effectiveness is determined by the properties of the differential operator. When multigrid is applied to the optimization problem (2.1)–(2.2), its effectiveness is determined by the properties of the *reduced Hessian* for this problem. It is far more challenging to analyze the reduced Hessian than it is to analyze the differential operator. The goal of this section is to explain those challenges. There are three aspects: (a) the formula for the reduced Hessian is complicated, (b) it may not be possible to determine the components of this formula, since they depend on the inverse operator for the (perhaps nonlinear) state equation (2.2), (c) the properties of the reduced Hessian may be dramatically different from those of the state equation (2.2), and hence contrary to intuition and expectation. The discussion here is condensed from [14].

The Hessian of the Lagrangian for (2.1)–(2.2) is

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

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

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

We then have the following expression for the Hessian of F with respect to a [14]:

$$\nabla^2 F = M_{aa} + M_{au} S_u^{-1} S_a + S_a^* S_u^{-*} M_{ua} + S_a^* S_u^{-*} M_{uu} S_u^{-1} S_a,$$

where S_a and S_u are the derivatives of S with respect to a and u, respectively. In this formula, * denotes the adjoint of an operator. This is the *reduced Hessian* [22] of $f(a, u)$ with respect to the equality constraints $S(a, u) = 0$.

When applying multigrid to the solution of Poisson's equation, for example, we have direct access to the representation A_h of the Laplacian on any grid. For our optimization problem, on the other hand, we do not directly have $\nabla^2 F(a)$ at our disposal.

Clearly the Hessian has a complicated structure. A further complication is that the reduced Hessian depends on S_u^{-1}, i.e., on the inverse of a Jacobian of the state equation. If the state equation is nonlinear it may be impossible to derive a formula for S_u^{-1}, and hence challenging to analyze the properties of the reduced Hessian.

The final complication is that the constraint operator $S(a, u)$ and the reduced Hessian can have qualitatively different properties. In [14] we study a simple problem with a linear differential equation

$$u_t + cu_x = 0$$

and a least-squares objective function. In this example, the constraint is a *hyperbolic* differential equation, and is not well-suited to multigrid. However the reduced Hessian for the problem is an *elliptic* operator and is similar to the Laplacian, which is an ideal operator for multigrid. Thus, even for simple problems, it may be challenging to guess if the optimization problem is well suited to multigrid.

3. The Diagnostic Tests. The performance of MG/Opt depends on at least four properties of the optimization problem. As our subsequent analysis will show, if these properties are satisfied, we can expect that MG/Opt will display various aspects of ideal performance, as enumerated in the previous section. Our assessment techniques will attempt to identify if these properties are satisfied, and hence attempt to diagnose the performance of MG/Opt as it is solving a particular optimization problem.

1. *Nonlinearity*—MG/Opt relies on an underlying optimization algorithm, in this case the truncated-Newton method TN. TN, like many optimization algorithms, is based on a quadratic approximation to the optimization problem derived from the Taylor series. If this is not a good approximation to the problem, i.e., if there are significant higher-order terms ("nonlinearities"), then it will be more difficult for MG/Opt to find the solution to the optimization problem.
2. *Consistency of the Optimization Problems*—The heuristic idea behind multigrid is to use computations on coarse problems to improve the solution on fine problems. This is only sensible if the optimization problems are all approximations to the same underlying problem. If this is not true, then MG/Opt will not work effectively. For example, it is possible to over-coarsen the problems, choosing a discretization so coarse that the solution of this problem does not provide useful information about the finer problems.
3. *Grid Complementarity*—MG/Opt is designed with the hope that the smoother (that is, the underlying optimization method) will be successful at identifying the high-frequency components of the solution rapidly. Then the multigrid recursion will identify the low-frequency components. This is only possible if the coarse grids are chosen so they are complementary to the fine grids, i.e., so they represent "algebraically smooth" components of the solution (components that are not effectively resolved by the smoother).
4. *Separability along Frequency Scales*—If multigrid is to be effective, the high-frequency components of the solution on the fine grid must not be significantly influenced by the components of the solution on the coarse grid. For

this to happen, the reduced Hessian must be approximately separable along frequency scales. If the reduced Hessian were equal to the Laplacian (an ideal operator for multigrid), then the reduced Hessian would be separable—in fact, diagonal—in the Fourier (frequency) basis.

Our goal is to identify diagnostic tests that can be used to determine if the problem has the properties that would make it a good candidate for MG/Opt. The diagnostic tests will be required to either:

- Use information normally computed by the algorithm.
- Use information about the optimization problem that can be deduced from information provided by the user (update and downdate procedures, function and gradient procedures) with limited amounts of auxiliary computation.

Here "limited" would mean that the cost of the diagnostic tests must be small compared with the cost of optimization.

It would be reasonable to consider diagnostic tests that were more expensive but that were only invoked at the request of the user, but we do not discuss such tests here. We think it likely that, if MG/Opt discovered evidence suggesting that a particular property was not satisfied, the user would be better able to investigate further using information about the problem that is not provided to MG/Opt. For example, the user might be willing to compute and analyze the reduced Hessian of the optimization problem, at least for selected values of the design variables.

Because the tests that we consider are simple, and use such limited information about the optimization problem, they are not definitive. They may misdiagnose properties of the problem, giving either false positives or false negatives. By choosing tolerances appropriately, they could be made more or less sensitive.

Despite these limitations, it is our hope that the diagnostic tests would be able to automatically identify properties of the optimization problem that influence the behavior of MG/Opt, and provide useful information that could guide a more thorough investigation of the properties of the optimization problem.

The following subsections discuss the diagnostic tests associated with each of the above properties.

3.1. Nonlinearity. Most standard optimization methods are based on a Taylor series approximation. If this is not a good approximation to the nonlinear function, that is, if the higher order terms in the series are significant, then the optimization method may not perform effectively. In addition, if these higher-order terms are significant, the other tests that we describe will not be reliable. Thus, the other tests will not be performed under these circumstances.

The Taylor series approximation is an approximation to

$$F_h(a+p)$$

for some value of a, typically the current estimate of the solution. The optimization method will return a value of p that will be used as a search direction in the algorithm. We will be concerned with the quality of the Taylor series approximation for $F_h(a+\alpha p)$ when $0 \leq \alpha \leq 1$. The difference between the Taylor series and $F_h(a+\alpha p)$ will be referred to as the "nonlinearity" in the problem.

In our TN optimization algorithm a quadratic Taylor series approximation is used. This is already a nonlinear problem, which may make the term "nonlinearity" confusing. In fact, the term will refer to nonlinearity of the *first-order optimality conditions*. For a quadratic approximation $Q(a)$, the optimality condition is $\nabla Q(a) = 0$, and if Q is quadratic then this condition is a linear equation. This is consistent with

the terminology when multigrid is applied to PDEs. The solution to a linear elliptic PDE (i.e., a linear equation with a positive-semi-definite operator) is equivalent to the minimization of a quadratic function involving the same operator.

In general, it will not be possible to measure the magnitude of the nonlinearity directly, particularly given the limited information available to the algorithm. If $\| p \|$ is small, then the higher-order terms may well be small, since as $\| p \|$ goes to zero the higher-order terms become negligible. But if $\| p \|$ is not small, we may be able to test for nonlinearity indirectly via the line search in the optimization algorithm.

The TN algorithm, like Newton's method, is based on a quadratic Taylor series approximation to $F(a+p)$. The TN optimization algorithm computes the search direction p using a conjugate-gradient (CG) method using a finite-difference approximation to $\nabla^2 f$ for the required matrix-vector products. If k iterations of the CG method are performed, and if the CG method does not detect degeneracy or indefiniteness, then p will be a solution to

$$\underset{p=P_i v_i}{\text{minimize}}\, F(a) + p^T \nabla F(a) + \frac{1}{2} p^T \nabla^2 F(a) p,$$

where P_i is a basis for the i-th Krylov subspace generated by the CG algorithm:

$$\left\{ -\nabla F(a), -[\nabla^2 F(a)] \nabla F(a), \ldots, -[\nabla^2 F(a)]^{i-1} \nabla F(a) \right\}.$$

Thus p is of the form

$$p = -P_i [P_i^T \nabla^2 F(a) P_i]^{-1} P_i^T \nabla F(a)$$

for some i.

A line search is then performed to determine the new estimate of the solution: $a_+ \leftarrow a + \alpha p$, where α is an approximate solution to

$$\underset{\alpha}{\text{minimize}}\, s(\alpha) \equiv F(a + \alpha p).$$

If $F(a)$ were a quadratic function, then p would minimize $F(a + p)$ over some Krylov subspace, and $\alpha = 1$ would minimize $s(\alpha)$. If the Taylor series is a good approximation to $F(a + p)$, then we would expect that $\alpha = 1$ would approximately minimize $s(\alpha)$, and hence that $s'(1) \approx 0$. If we expand $s(\alpha)$ in a Taylor series about $\alpha = 0$, and substitute the above formula for p for any value of $i > 0$, we obtain

$$s'(1) = O(\| p \|^3).$$

That is, $s'(1)$ is the value of the higher-order terms in the Taylor series for this particular p, and hence is a measure of nonlinearity. For additional details, see [19]. As the solution is approached and $\| p \| \to 0$, a step of $\alpha = 1$ is likely to be accepted in the line search. Whenever $\alpha = 1$ is an acceptable search direction, we will say that the search direction p is *well scaled*.

Since TN is likely to produce well-scaled search directions when the nonlinearities are not severe, we can use the step length as a diagnostic for nonlinearity:

- *Nonlinearity Test*—Is $s'(1) \approx 0$ in the line-search for the TN method?

If this test is satisfied, then we will assume that the nonlinearities are relatively benign, and that it will be appropriate to consider using the other tests discussed in the article. This test is analogous to the "over solving" test from [8], although that test was developed for a different purpose.

A simpler test could be based on whether the line search used a step length equal to one, and this might be desirable if the line search algorithm did not use derivative values to determine the step length.

The nonlinearity test here applies to the optimization problem (2.1), where the state equation (2.2) is solved for u given a. In more general settings there might be additional constraints on the variables, or the state equation might only be solved approximately. In such settings the nonlinearity test would have to be modified to take these changes into account.

3.2. Consistency of the Optimization Problems. Because computations on the coarser grids are typically much cheaper than computations on finer grids, it is tempting to use coarser grids extensively. This is only reasonable if the optimization problems on the various grids are good approximations to each other. For example, they might all be discrete approximations to a single continuous problem. In this section we develop a test to determine if the optimization problems are indeed consistent, in the sense that they are all good approximations to each other.

Problems can be inconsistent even if they are all approximations to the same underlying continuous problem. For example, if too coarse a grid is chosen, the corresponding optimization problem will be a poor approximation to the original problem, and the results of these computations will be of little value. So, while cheap, they may be almost useless. This is at the least a waste of computational effort, and may possibly hinder the overall algorithm by suggesting confounding search directions. How do we determine if the optimization problems are consistent?

Our test will compare problems on two successive grids. Let us assume that the problem on the coarser grid is sensible as an optimization problem, i.e., that it has a solution and that its derivatives have the necessary smoothness. Then the optimization algorithm applied to this problem will be effective, and will work in an unremarkable fashion. Thus, any test of consistency will focus on the behavior of the multigrid recursion.

If the problems are not consistent, then the coarse-grid solution $a_{H,2}$ will not produce a good search direction e_h for the fine-grid problem. What does "not a good search direction" mean? We are using the coarse grid problem to identify a search direction for the fine grid problem. We obtained a search direction by applying an optimization method to the coarse grid problem. In other words, the optimization method produces a reduction in in the course grid problem. Since the coarse grid problem is assumed to be an approximation to the fine grid problem, we would hope that the reduction *predicted* by the course grid problem would be approximately equal to the *actual* reduction in the fine grid problem obtained via the line search.

Before stating the test, it will be helpful to derive approximations for the actual and predicted reductions based on

$$\begin{aligned} \langle \text{predicted reduction} \rangle &\equiv R_p = \tilde{F}_H(a_{H,1}) - \tilde{F}_H(a_{H,2}) \\ \langle \text{actual reduction} \rangle &\equiv R_a = F_h(a_{h,1}) - F_h(a_{h,2}). \end{aligned}$$

(For the definition of $\tilde{F}$, see the description of algorithm MG/Opt in Section 2.1.) The formula for the actual reduction assumes that the multigrid line search uses a step length of $\alpha = 1$ in the multigrid line search. The test can be performed in this way even if the line search does not accept $\alpha = 1$. There are theoretical reasons to expect that the search direction from the multigrid recursion will be well scaled (see [14] and Section 3.1), at least as the algorithm approaches the solution.

Taylor series expansions give

$$\begin{aligned}
-R_p &= \tilde{F}_H(a_{H,2}) - \tilde{F}_H(a_{H,1}) \\
&= F_H(a_{H,1} + e_H) - F_H(a_{H,1}) - v_H^T a_{H,1} - v_H^T e_H + v_H^T a_{H,1} \\
&= F_H(a_{H,1} + e_H) - F_H(a_{H,1}) - e_H^T \nabla F_H(a_{H,1}) + e_H^T I_h^H \nabla F_h(a_{h,1}) \\
&= e_H^T \nabla F_H(a_{H,1}) + \frac{1}{2} e_H^T \nabla^2 F_H(a_{H,1}) e_H + O(\| e_H \|^3) - e_H^T \nabla F_H(a_{H,1}) + e_H^T I_h^H \nabla F_h(a_{h,1}) \\
&= e_H^T I_h^H \nabla F_h(a_{h,1}) + \frac{1}{2} e_H^T \nabla^2 F_H(a_{H,1}) e_H + O(\| e^H \|^3),
\end{aligned}$$

and

$$\begin{aligned}
-R_a &= F_h(a_{h,1} + I_H^h e_H) - F_h(a_{h,1}) \\
&= (I_H^h e_H)^T \nabla F_h(a_{h,1}) + \frac{1}{2} (I_H^h e_H)^T \nabla^2 F_h(a_{h,1}) I_H^h e_H + O(\| I_H^h e_H \|^3) \\
&= e_H^T (I_H^h)^T \nabla F_h(a_{h,1}) + \frac{1}{2} e_H^T (I_H^h)^T \nabla^2 F_h(a_{h,1}) I_H^h e_H + O(\| e_H \|^3) \\
&= C_I \left[e_H^T I_h^H \nabla F_h(a_{h,1}) + \frac{1}{2} e_H^T I_h^H \nabla^2 F_h(a_{h,1}) I_H^h e_H + O(\| e_H \|^3) \right],
\end{aligned}$$

where C_I is the constant that relates I_h^H and I_H^h (see section 2.1). Thus

$$C_I^{-1} R_a - R_p = \frac{1}{2} e_H^T \left[\nabla^2 F_H(a_{H,1}) - I_h^H \nabla^2 f^h(a_0^h) I_H^h \right] e_H + O(\| e^H \|^3).$$

This formula involves two terms:

- A second-order term involving the difference between ⟨coarse-grid Hessian⟩ and ⟨restricted fine-grid Hessian⟩.
- A third-order term measuring "nonlinearity". As discussed in Section 3.1, this is a generic issue for nonlinear optimization. We can use the nonlinearity test to determine if the third-order term is likely to be significant.

Assuming that the higher-order terms can be ignored, i.e., that the nonlinearity test is satisfied, the difference between predicted and scaled actual reduction can be used to measure the difference between the coarse-grid Hessian and the restricted fine-grid Hessian. If these two matrices are dissimilar, we cannot expect the coarse-grid problem to be a useful approximation to the fine-grid problem.

For these reasons, our test for consistency will be of the form

- *Problem Consistency Test*—Is

$$\frac{|\langle \text{predicted reduction} \rangle - C_I^{-1} \langle \text{actual reduction} \rangle|}{|\langle \text{actual reduction} \rangle|} \leq \text{tolerance?}$$

Since in many applications this will be a test for over-coarsening, the test may only need to be applied on the coarsest grids.

It is possible that this test would fail for a simple reason that might be considered a programming error. Consider an optimization problem with objective function

$$F(a) = \int \int_0^T (u(x,t) - \phi(x,t))^2 \, dx \, dt.$$

If we were to discretize in space with mesh size h_x, and in time with mesh size h_t, then we could use the following discrete approximation to the objective function:

$$F_h(a_h) = h_x h_t \sum_{i=0}^{N_x} \sum_{j=0}^{N_t} C_{i,j} (u_{i,j} - \phi_{i,j})^2$$

where the constants $C_{i,j}$ depend on the quadrature rules used to estimate the integrals.

From the point of view of optimization on a single grid, the same solution would be obtained using the scaled objective function

$$\hat{F}_h(a_h) = \frac{1}{h_x h_t} F_h(a_h).$$

However, if MG/Opt is used, the corresponding coarse-grid problems would be different. Consider a coarse-grid problem with mesh sizes $H_x = 2h_x$ and $H_t = 2h_t$. Then $F_h(a_h) \approx F_H(I_h^H a_h)$ since they both are approximations to the same continuous problem. However $\hat{F}_h(a_h) \approx 4\hat{F}_H(I_h^H a_h)$. Since v_H in the coarse-grid optimization problem in MG/Opt mixes quantities derived from both the fine and coarse grids, MG/Opt would perform differently on F_h than on $\hat{F}_h$.

When using MG/Opt it is important that the optimization problems on the various grids be *mesh invariant*, i.e., that they are all discrete approximations to the same continuous problem. It is easy to overlook this issue. Typically, an optimization problem would first be tested on a single grid using a standard optimization method. The optimization method would behave much the same way on both F_h and $\hat{F}_h$, except perhaps for rounding errors. If MG/Opt is then applied, it may well solve the problem, but not with the efficiency expected of multigrid, and the cause of the inferior performance might be difficult to detect.

Our diagnostic test compares predicted and actual reduction, and would reveal that the optimization problems were not mesh invariant, but that requires completing a multigrid recursion before using the test. It makes sense to use a simple test for mesh invariance before using MG/Opt:

- *Mesh Invariance Test*—Is $F_h(a_h) \approx F_H(I_h^H a_h)$?

This test would only need to be performed during the first iteration of MG/Opt.

3.3. Grid Complementarity. The coarse grids must be chosen to be complementary to the fine grids. That is, the components of the solution defined by the coarse grid must represent "algebraically smooth" components—components that are not well resolved by the smoother (i.e., the underlying optimization method). In a sense, our diagnostic test will attempt to determine if the coarse grid corresponds to algebraically smooth components of the solution.

Multigrid methods were first developed based on geometric principles, for example, the coarsening of a discretization of a simple region like a line segment. In such a setting, it is typically straightforward to select the coarser grids. Subsequent research has led to the development of algebraic multigrid methods which attempt to identify the coarse grids automatically, so problems can be solved by a multigrid method without any effort or understanding on the part of the user. Much of this research is focused on linear equations arising from the discretization of partial differential equations. The coefficient matrices are sparse, and many algebraic multigrid methods are defined using knowledge of the sparsity pattern and the entries of the coefficient matrix.

In our setting, the relevant coefficient matrix is the reduced Hessian. Typically we will not know its entries nor its sparsity pattern. It may even be a dense matrix, even when the components of the reduced Hessian are sparse. For these reasons we base our diagnostic test on some general principles of algebraic multigrid, rather than specific algebraic multigrid methods.

We refer only to the choice of grids, but the choice of the interpolants I_h^H and I_H^h can also affect the algorithm's ability to approximate algebraically smooth components

of the solution [4]. Because MG/Opt has such limited information available, it is not clear that MG/Opt can isolate the properties of the interpolant from the properties of the coarse grid. For this reason we conflate the two issues, and refer only to the choice of the coarse grid when discussing the diagnostic test.

In MG/Opt we use TN as an underlying optimization method, which is in turn based on the conjugate-gradient (CG) method. To simplify the explanation, let us assume that we are working with a quadratic optimization problem and that the reduced Hessian is positive definite.

It is well known that CG typically reduces the error in components of the solution corresponding to large eigenvalues of the coefficient matrix [9]. Then the complementary components of the solution will be those corresponding to the small eigenvalues. In the context of algebraic multigrid, such components are referred to as the "near null space" of the coefficient matrix because they correspond to small (near zero) eigenvalues [3]. (CG is not typically used within traditional multigrid methods, but the same reasoning applies.) One way to measure whether the coarse grid is appropriate is to compute a generalized Rayleigh quotient for the error η_h in the variables a_h:

$$RQ_h(\eta_h) = \frac{(G^h\eta_h)^T(G^h\eta_h)}{\parallel G^h \parallel_2 (G^h\eta_h)^T\eta_h},$$

where G^h is the reduced Hessian of the optimization problem. In the context of algebraic multigrid the coefficient matrix is known and, if the algorithm is applied to a linear system with zero right-hand side, the current estimate of the solution is also the error. In that setting this is a practical formula. For MG/Opt neither η_h nor G^h is available.

There are two steps to obtaining a practical test. The first is to replace the error η_h with the search direction e_h from the multigrid recursion. If MG/Opt is working effectively, the multigrid search direction should produce an accurate approximation to the components of the solution in the near null space. If that is the case, then the search direction e_h should be an accurate approximation to the error η_h. We can then approximate $G^h\eta_h$ with the matrix-vector product G^he_h, which can be estimated by finite-differencing of gradient values (a technique already used by TN in its CG algorithm).

That leaves $\parallel G^h \parallel_2 = \rho(G^h)$, the largest eigenvalue of G^h. This can be estimated as a by-product of the CG method. The CG method is equivalent to the Lanczos method for computing eigenvalues [24]. The Lanczos method iteratively computes a tridiagonal matrix T^h whose eigenvalues approximate those of the coefficient matrix G^h. Typically the largest eigenvalues are the first to converge. It is possible to use quantities from the CG method to construct this tridiagonal matrix. We can use $\rho(T^h) = \parallel T^h \parallel_2$ as an approximation to $\parallel G^h \parallel_2$. (We will not compute the full matrix T^h, but just use the tridiagonal matrix generated at the final iteration of the CG method as it computes a search direction for TN.)

Using these approximations, our test will be based on the following approximate Rayleigh quotient:

$$\hat{R}_h(e_h) = \frac{(G^he_h)^T(G^he_h)}{\rho(T^h)\,(G^he_h)^Te_h}.$$

Thus our diagnostic test for the appropriateness of the coarse grid will be

- *Grid Complementarity Test*—Is $\hat{R}(e_h)$ small?

Within a traditional algebraic multigrid method, it is possible to use techniques such as compatible relaxation to determine an appropriate coarse grid and interpolant for use within a multigrid method [3]. MG/Opt is not provided with enough information about the optimization problems to do this. However, the user might be able to take advantage of these techniques if the coarse-grid test were not satisfied.

If it is determined that the problem does not satisfy grid complementarity, but that the problem satisfies the other tests, it may be still be possible to use multigrid techniques. In such a case, the user may be able to identify a preconditioner that scales the problem so that grid complementarity is satisfied for the preconditioned problem.

3.4. Separability. Multigrid will be a successful strategy if the solutions to coarse-grid problems are good approximations to the low-frequency components of solutions on finer grids, and if the high-frequency components of the solution can be computed by the smoother in a nearly-independent manner. If we write the problem in a basis that separates high and low frequencies, then the reduced Hessian for the optimization problem will have the following block form

$$G^h = \begin{pmatrix} G_{hh} & G_{Hh}^T \\ G_{Hh} & G_{HH} \end{pmatrix}$$

where H and h are used to designate low- and high-frequency components, respectively. For multigrid to be successful, the off-diagonal block must be small compared to the diagonal blocks. Ideally, the off-diagonal block will be zero. Our goal is to develop a practical test to determine if the off-diagonal block of the reduced Hessian is small. We call this a test for *separability*, i.e., separability of high- and low-frequency components of the solution.

This is challenging for several reasons. First, the MG/Opt algorithm only uses function and gradient values from the optimization problem. It does not have Hessian information available, except in the sense that it approximates Hessian-vector products using finite differences of gradient values. Second, we are asking for information about the reduced Hessian, and the formulas for the reduced Hessian are complicated and involve the inverse operator for the linearized state equation (see Section 2.2). Third, the reduced Hessian is not likely to be represented in a basis that isolates high and low frequencies, such as a Fourier basis.

To determine separability, we would like to know if the off-diagonal block $G_{Hh} \approx 0$, when the reduced Hessian is represented in a basis that separates high and low frequency components. This is a challenging analysis even for simple, linear state equations, where all the component operators are available explicitly. How then can MG/Opt with its limited problem information have any hope of assessing this property?

Suppose that we were to use update and downdate operators I_H^h and I_h^H that perfectly isolated low- and high-frequency components, and consider a fine-grid vector z^h with fine-grid and coarse-grid components z_h and z_H, respectively. Then

$$I_h^H z^h = I_h^H \begin{pmatrix} z_h \\ z_H \end{pmatrix} = z_H \qquad \text{and} \qquad I_H^h z_H = \begin{pmatrix} 0 \\ z_H \end{pmatrix}.$$

Using these operators, a test for separability can be obtained using a Hessian-vector product for an appropriately chosen vector. Hessian-vector products are estimated by TN as a step in the computation of the TN search direction, so this calculation can be performed using the information normally available to MG/Opt.

Let $z^h = (z_h, 0)^T$ with $z_h \neq 0$, i.e., z^h consists only of high-frequency components. Such a vector can be obtained via

$$z^h = w^h - I_H^h I_h^H w^h$$

for some arbitrary initial vector $w^h = (w_h, w_H)^T$. Then we compute

$$I_h^H G^h z^h = I_h^H \begin{pmatrix} G_{hh} z_h \\ G_{Hh} z_h \end{pmatrix} = G_{Hh} z_h.$$

Thus, a downdated Hessian-vector product of this form is a measure of separability.

Unfortunately, this test is not useful in its raw form because the update and downdate operators need not perfectly isolate low- and high-frequency components. They may be based on interpolation schemes that cause some mixing of high- and low-frequencies. (This issue is illustrated in the computational tests in Section 5.)

Nevertheless, it is possible to refine this idea to produce a useful test. While the downdate operator provided by the user is unlikely to isolate low-frequency components precisely, it should be an approximate low-pass filter. That is, high-frequency components should be damped, and low-frequency components should undergo little change. If we begin with some random vector w^h on the fine grid, then repeated application of the downdate and update operators

$$\hat{w}^h = (I_H^h \, I_h^H)^k w^h$$

should result in a vector $\hat{w}^h$ that is dominated by low-frequency terms. Then $z^h = w^h - \bar{w}^h$ should be dominated by high-frequency terms. Likewise, we must process the resulting Hessian-vector product $G^h z^h$ with $(I_H^h \, I_h^H)^k$ to isolate the low-frequency components.

One additional point deserves comment. The test below uses uses $\| G^H \|$. An estimate for the norm of the Hessian is computed by the grid-complementarity test (see Section 3.3) and is used here. We divide by $\| G^H \|$ and not $\| G^h \|$ since we are trying to assess the relative magnitude of the diagonal and off-diagonal blocks.

In summary, here is the test for separability:

1. Generate a random vector w^h on the fine grid. Let $\bar{w}^h = w^h$.
2. Iterate k_1 times: $\bar{w}^h = I_H^h I_h^H \bar{w}^h$ [this vector is dominated by low-frequency terms]
3. Define $z^h = w^h - \bar{w}^h$. [this vector is dominated by high-frequency terms]
4. Compute the Hessian-vector product: $(Gz)^h = G^h z^h$. Let $(Gz)^H = I_h^H (Gz)^h$
5. Iterate k_2 times: $(Gz)^H = I_h^H I_H^h (Gz)^H$ [this vector is dominated by low-frequency terms]
6. Test for separability:
   - *Separability Test*—Is $\| (Gz)^H \| / (\| z^h \| \, \| G^H \|) \approx 0$?

This test only uses information available to MG/Opt. The software does not know what "coarse", "fine", "high frequency", and "low frequency" mean. Instead it uses the update and downdate procedures as a surrogate. If this test revealed that the problem might not be separable along frequency scales, it might be possible to ask the user to provide a more precise analysis (e.g., using a Fourier transform) to confirm the diagnosis.

If the reduced Hessian is positive definite, then there is some expectation that the separability test will be automatically satisfied. This is a consequence of the results

in [16]. Consider two symmetric matrices

$$A = \begin{pmatrix} M & R \\ R^T & N \end{pmatrix} \quad \text{and} \quad \tilde{A} = \begin{pmatrix} M & 0 \\ 0 & N \end{pmatrix}.$$

For any matrix A, let $\{\lambda_k(A)\}$ be the set of eigenvalues A ordered from smallest to largest, and let $\sigma(A)$ be the spectrum of A.

Using this notation, Theorem 1 in [16] states (in part): If $\lambda_1(A) \notin \sigma(N)$, then

$$|\lambda_1(A) - \lambda_1(\tilde{A})| \leq \frac{\| R \|_2^2}{\min_i |\lambda_1(A) - \lambda_i(N)|}.$$

We will apply this theorem with $M = G_{hh}$, $N = G_{HH}$, and $R = G_{hH}$. Hence $A = G^h$ and

$$\tilde{A} = \tilde{G}^h = \begin{pmatrix} G_{hh} & 0 \\ 0 & G_{HH} \end{pmatrix}.$$

We will assume that $\tilde{G}^h$ is positive definite. (If $\tilde{G}^h$ has a negative eigenvalue then so does G^h.) In applying the theorem, our main concern is whether G^h remains positive definite as $\| G_{hH} \|$ increases. Thus, we assume that $\min_i |\lambda_1(G^h) - \lambda_i(G_{HH})| = |\lambda_1(G^h) - \lambda_1(G_{HH})|$.

Then to guarantee that G^h is positive definite, the theorem implies that it is sufficient that $\| G_{Hh} \| < \lambda_1(G_{HH})$. If grid complementarity is satisfied then the smallest eigenvalue of G_{HH} will be small. Thus, if G^h is positive definite, then $\| G_{Hh} \|$ is expected to be small and G^h will be nearly separable.

This suggests that lack of separability may be a more significant issue at points where the Hessian is not positive definite, typically at points far from the solution.

4. Sufficiency of the Diagnostic Tests. Our goal in this section is to demonstrate that our diagnostic tests address the relevant issues for the performance of MG/Opt. We have identified four properties that are relevant: (a) nonlinearity, (b) consistency of optimization problems, (c) grid complementarity, and (d) separability along frequency scales.

Since MG/Opt is based on quadratic approximations to the optimization problem, the performance of MG/Opt will deteriorate if there is significant nonlinearity, since in that case the quadratic problems will be poor approximations to the optimization problem. For this reason, the nonlinearity test is clearly necessary in assessing the performance of MG/Opt.

To assess the importance of the other tests, we will assume now that the nonlinearity test is satisfied, and that the optimization problem (expressed in terms of the reduced Hessian) can be approximated effectively by a quadratic problem

$$\underset{a_h}{\text{minimize}}\, F_h(a_h) = \frac{1}{2} a_h^T G^h a_h - b_h^T a_h$$

where b_h is some vector. We assume that the optimization problem is well defined, i.e., that G^h is a positive-definite matrix. This is not an unreasonable assumption. The reduced Hessian is guaranteed to be positive-semi-definite at a local minimizer. Also, standard convergence theory assumes that the reduced Hessian will be positive definite at a local minimizer; without this assumption we will not be able to guarantee that either MG/Opt or TN converges.

In analyzing MG/Opt we will only refer to two grids and accept that only that an approximate solution is found to the coarse grid problem. Such an approximate solution could be obtained by applying the MG/Opt recursion. Thus our description in terms of only two grids applies to the case when more than two grids are used in MG/Opt, and is not a restrictive assumption.

On each grid we will apply TN to the quadratic problem. This is equivalent to applying the CG method with restarts to the quadratic problem. It is well known that the CG method reduces the error as measured in the G^h-norm, and it is straightforward to show that the line search in TN further reduces the error in the G^h-norm. Asymptotically TN will also reduce the error in the 2-norm, although this norm of the error will not decrease monotonically.

We will write the problem in terms of high-frequency and low-frequency components. In the ideal case $I_h^H = (0\ I)$, but typically we will have

$$I_h^H = (\,0 \quad I\,) + \Delta_I$$

for some matrix Δ_I where $\| \Delta_I \|$ will be small if the downdate operator is effective at identifying low-frequency components. (As discussed in Section 2, we have assumed that the user has done this.)

We will denote the solution by a^*, with fine- and coarse-grid components a_h^* and a_H^*, respectively.

In terms of high-frequency and low-frequency components, we have

$$G^h = \begin{pmatrix} G_{hh} & G_{hH} \\ G_{hH}^T & G_{HH} \end{pmatrix}$$

and the minimizer of the quadratic optimization problem is the solution to

$$G^h a_h = b_h \quad \text{or} \quad \begin{pmatrix} G_{hh} & G_{hH} \\ G_{Hh}^T & G_{HH} \end{pmatrix} \begin{pmatrix} (a_h)_h \\ (a_h)_H \end{pmatrix} = \begin{pmatrix} (b_h)_h \\ (b_h)_H \end{pmatrix}.$$

Suppose that the diagnostic tests are all satisfied. Then an iteration of MG/Opt will behave as follows:

1. MG/Opt will apply TN to the fine grid problem to obtain $a_h^{(1)}$, reducing the error in the G^h-norm.
2. In the multigrid recursion, MG/Opt will solve the coarse grid problem. If the separability and operator-consistency tests are satisfied, this will determine an approximation to $(a_h)_H$. If the coarse-grid and interpolants are appropriate, then this approximation will be in the near-null space of G^h, i.e., will represent components of the solution not well resolved by the TN method on the fine grid.
3. MG/Opt will apply TN to the fine grid problem, further reducing the error in the G^h-norm.

This gives a rough idea of why the diagnostic tests are sufficient to assess the performance of MG/Opt. We can make this argument more precise using perturbation analysis. The central issue will be how well MG/Opt approximates a_H^* via the multigrid recursion. The structure of the MG/Opt algorithm ensures the overall convergence of the algorithm, with TN taking care of the high-frequency components of a_h*.

The value of a_H^* can be found by solving the linear system

$$(G_{HH} - G_{hH}^T G_{hh}^{-1} G_{hH}) a_H = (b_h)_H - G_{hH}^T G_{hh}^{-1} (b_h)_h. \tag{4.1}$$

We will demonstrate that our diagnostic tests are sufficient by showing that MG/Opt solves a perturbation of this equation, and derive bounds on the perturbation that are related to our diagnostic tests. To do this we find formulas for some of the terms that will arise in our discussion.

If we apply MG/Opt to the quadratic optimization problem, then the recursion will approximately solve a perturbed equation. We can derive a formula for that equation, involving a residual term η_H corresponding to the residual that results from the approximate solve. That is, we can determine that the multigrid recursion estimates a_H^* by solving for $\hat{a}_H^*$ in the coarse-grid equation

$$\begin{aligned} G^H \hat{a}_H &= b_H + v_H + \eta_H \\ &= b_H + [G^H(I_h^H a_h^{(1)}) - b_H] - [I_h^H(G^h a_h^{(1)} - b_h)] + \eta_H \\ &= G^H(a_h^{(1)})_H + G^H \Delta_I a_h^{(1)} - [(\,0 \quad I\,) + \Delta_I](G^h a_h^{(1)} - b_h) + \eta_H \\ &= (b_h)_H - G_{hH}^T(a_h^{(1)})_h + (G^H - G_{HH})(a_h^{(1)})_H \\ &\qquad + G^H \Delta_I a_h^{(1)} - \Delta_I(G^h a_h^{(1)} - b_h) + \eta_H. \end{aligned}$$

We can assess the performance of MG/Opt by finding a bound for $\| a_H^* - \hat{a}_H^* \|$.

We can write the coarse-grid equation as

$$G^H \hat{a}_H = \bar{b}_H$$

where

$$\begin{aligned} \bar{b}_H = b_H &- G_{hH}^T(a_h^{(1)})_h + (G^H - G_{HH})(a_h^{(1)})_H \\ &+ G^H \Delta_I a_h^{(1)} - \Delta_I(G^h a_h^{(1)} - b_h) + [(b_h)_H - b_H] + \eta_H. \end{aligned}$$

Then we can write (4.1) as

$$(G^H + \delta G)(\hat{a}_H + \delta a) = \bar{b}_H + \delta b$$

where

$$\begin{aligned} \delta G &= (G_{HH} - G^H) - G_{hH}^T G_{hh}^{-1} G_{hH} \\ \delta b &= -G_{hH}^T G_{hh}^{-1}(b_h)_h + G_{hH}^T(a_h^{(1)})_h - (G^H - G_{HH})(a_h^{(1)})_H \\ &\qquad - G^H \Delta_I a_h^{(1)} + \Delta_I(G^h a_h^{(1)} - b_h) - \eta_H. \end{aligned}$$

Then standard perturbation analysis for linear equations [9] gives

$$\frac{\| \delta a \|}{\| \hat{a}_H^* \|} \leq \kappa(G^H) \left(\frac{\| \delta G \|}{\| G^H \|} + \frac{\| \delta b \|}{\| \bar{b}_H \|} \right) + (\text{higher-order terms}).$$

Here $\kappa(G^H)$ is the condition number of G^H: $\kappa(G^H) = \| G^H \| \cdot \| (G^H)^{-1} \|$.

It is straightforward to conclude that

$$\begin{aligned} \| \delta G \| &\leq \| G_{hH} \|^2 \| G_{hh}^{-1} \| + \| G^H - G_{HH} \| \\ \| \delta b \| &\leq \| G_{hH} \| \cdot \| G_{hh}^{-1} \| \cdot \| (b_h)_h \| + \| G_{hH} \| \cdot \| (a_h^{(1)})_h \| \\ &\qquad + \| G^H - G_{HH} \| \cdot \| (a_h^{(1)})_H \| + \| G^H \Delta_I a_h^{(1)} \| \\ &\qquad + \| \Delta_I(G^h a_h^{(1)} - b_h) \| + \| \eta_H \|. \end{aligned}$$

These terms will be small if our diagnostic tests are satisfied. If the problem consistency test is satisfied then $\| G^H - G_{HH} \|$ will be small. If the separability test is satisfied then $\| G_{hH} \|$ is small. If the update and downdate operators are chosen appropriately (a responsibility of the user) then $\| \Delta_I \|$ will be small. Also, G_{hh} will correspond to the large eigenvalues of G^h, so the multigrid recursion and TN methods will be complementary, and $\| G_{hh}^{-1} \|$ will be as small as possible. The quantities b_h, $(a_h^{(1)})_h$, and $(a_h^{(1)})_H$ are computed explicitly by MG/Opt, and can be monitored directly. Finally, the structure of MG/Opt will ensure that η_H will go to zero as the solution is approached.

Thus the performance of MG/Opt is governed by the properties associated with our diagnostic tests.

5. Computational Tests. We apply the MG/Opt algorithm together with the diagnostic tests to a set of test problems chosen to isolate the various properties that are of interest. We do not include in this paper computational experiments for the nonlinearity test, since this topic is addressed in [8], for example. We focus on the remaining tests which are less familiar. The multigrid software used is an adaption of the software from [14] with the addition of the diagnostic tests. It is a Matlab implementation of the MG/Opt algorithm discussed in Section 2.1.

Our goal in this Section is to demonstrate that the proposed diagnostic tests isolate the relevant properties of the optimization problem. For that reason, we choose test problems that may seem simple, but that are useful because they isolate these properties. We do not attempt to perform experiments of a production-grade multigrid algorithm on realistic applications. That is beyond the scope and aims of this paper.

Before discussing the computational test, we first present the revised MG/Opt algorithm including the diagnostic tests.

5.1. Details of the Algorithm. Below is a revised description of the MG/Opt algorithm, indicating how the diagnostic tests are incorporated. As before, 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 $a_{h,1}$; *perform nonlinearity test; estimate* $\| G^h \|$*, the norm of the reduced Hessian, for use in the grid-complementarity test.* Downdate the result to obtain $a_{H,1} = I_h^H a_{h,1}$.
  - Compute $v_H = \nabla F_H(a_{H,1}) - I_h^H \nabla F_h(a_{h,1})$.
  - *If this is the first MG/Opt iteration, perform the mesh invariance test.*
  - Recursively apply MG/Opt (with initial estimate $a_{H,1}$) to solve:

$$\underset{a_H}{\text{minimize}}\, \tilde{F}_H(a_H) \equiv F_H(a_H) - v_H^T a_H$$

    subject to the bound constraints

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

    to obtain $a_{H,2}$.
  - *Perform separability test.*

– Compute the search direction $e_h = I_H^h(a_{H,2} - a_{H,1})$.
– Use a line search to obtain $a_{h,2} = a_{h,1} + \alpha e_h$.
– *Perform problem-consistency test and grid-complementarity test.*
– Partially minimize $F_h(a)$, with initial estimate $a_{h,2}$, to obtain $a_h^{(1)}$; *estimate* $\| G^h \|$*, the norm of the reduced Hessian, for use in the separability test on the next finer grid.*

5.2. Grid Complementarity. In assessing the grid-complementarity test, it is revealing to consider two related computational examples that isolate this property.

Both are quadratic optimization problems, i.e., there is no constraint in the optimization problem, just an objective function:

$$F(a) = \frac{1}{2} a^T G a - b^T a,$$

for some positive-definite matrix G and vector b. For this reason, the matrices for these problems represent the reduced Hessian for some equivalent constrained problem.

The first has a uniformly discretized one-dimensional Laplacian as its Hessian ("Laplacian"). The second uses the same Laplacian, but with a random permutation of the assignment of eigenvalues to eigenvector ("Permuted"). That is, if $G = V^T D V$ is the spectral decomposition of G, then the diagonal entries in the diagonal matrix D are randomly permuted.

The first example is ideal for MG/Opt. It satisfies separability (it is separable in the Fourier basis), problem consistency, and grid complementarity. The second also satisfies separability and problem consistency, but not grid complementarity since the coarse grid does not correspond to the near null-space for the problem.

For the first example, the vector b in the linear term is obtained by summing the eigenvectors of the Hessian, and then artificially constraining the components to be between -2 and 2. (Any entry outside this range is set to -2 if it is too small, and to 2 if it is too big.) Appropriate transformations are used to form the corresponding vector for the other example in such a way as to keep the solution of the optimization problem the same. The initial guess for the Laplacian problem is chosen randomly as follows: In Matlab, the random number generator is initialized with `rand('twister',5489)` and then we use the initial guess `v0 = 20*rand(n,1);` where `n` is the dimension of the problem. We transform this vector to get the equivalent initial guess for the Permuted problem.

In these two test problems we use a uniform discretization of the interval $[0, 1]$ with $n = 2^m - 1$ interior points, for $m = 3, 4, \ldots, 10$.

Table 5.2 shows the results of the grid-complementarity test for these two problems. For each entry we give values of n_H and n_h, the dimensions of the two grids used in the test. It is clear that the results for the permuted Laplacian are worse than for the Laplacian, and the difference grows ever more dramatic as the dimensions of the grids get smaller. This is not surprising. For larger dimensions there is not a dramatic difference in magnitude between the norms of G^h and G^H. It is only as the dimensions get smaller that there are significant reductions in G^H.

5.3. Problem Consistency. If the underlying optimization problem is sensible, and if the optimization problems on each grid are consistent discretizations of the same underlying optimization problem, then in many circumstances the problem consistency test will be satisfied. Even in this situation, however, it is possible to over-coarsen the discretization, and in that case the problem-consistency test may be

TABLE 5.1
Grid-Complementarity Test

n_H	n_h	Laplacian	Permuted	Ratio
7	15	0.03	0.34	11.7
15	31	0.04	0.21	4.8
31	63	0.04	0.57	13.6
63	127	0.07	0.30	4.5
127	255	0.13	0.25	1.9
255	511	0.20	0.34	1.7
511	1023	0.26	0.77	3.0

violated. But in general, a "sensible" optimization problem is likely to satisfy the problem consistency test.

For these reasons, we construct a set of artificial optimization problems to illustrate the problem consistency test. The problems will be be based on the original Laplacian test problem from the previous subsection. We will apply MG/Opt with two grids: $n = 1023$ and $n = 511$. For $n = 1023$ the optimization problem is the same as before. But for $n = 511$ we artificially distort the Hessian using an orthogonal transformation. Because of the way in which the optimization problems are constructed, they satisfy the nonlinearity test (the problem remains quadratic), the separability test (the fine-grid Hessian is unchanged), and the grid complementarity test (the orthogonal transformation does not alter the overall magnitude of the eigenvalues). However, as the distortion becomes greater, the problem consistency test is violated.

To define the optimization problems, we need only describe how the coarse-grid Hessian G^H is modified. We first generate a random skew-symmetric matrix R using the following Matlab commands: `randn('state',1000), R = randn(511,511), R = R-R', R = R/norm(R)`. The resulting matrix satisfies $\| R \| = 1$. Then for various values of $\alpha \in [0, 1]$ we define an orthogonal matrix Q via

$$Q = (I + \alpha R)(I - \alpha R)^{-1}.$$

It is straightforward to verify that Q is orthogonal. Finally we define the distorted Hessian by

$$G^H \leftarrow Q^T G^H Q.$$

For $\alpha = 0$, the Hessian G^H is unchanged, but as α increases, the distortions grow ever greater. For each value of α we ran one iteration of MG/Opt, with the same initial guess each time, and computed the value of the problem-consistency test after the recursion.

The results are in Figure 5.3. As expected, the problem consistency test is satisfied for $\alpha \approx 0$, but for larger values of α it is clearly violated. The test is sensitive to distortions in the Hessian.

5.4. Separability. To demonstrate the separability test we will use a set of test matrices derived from the one-dimensional Laplacian problem described in Section

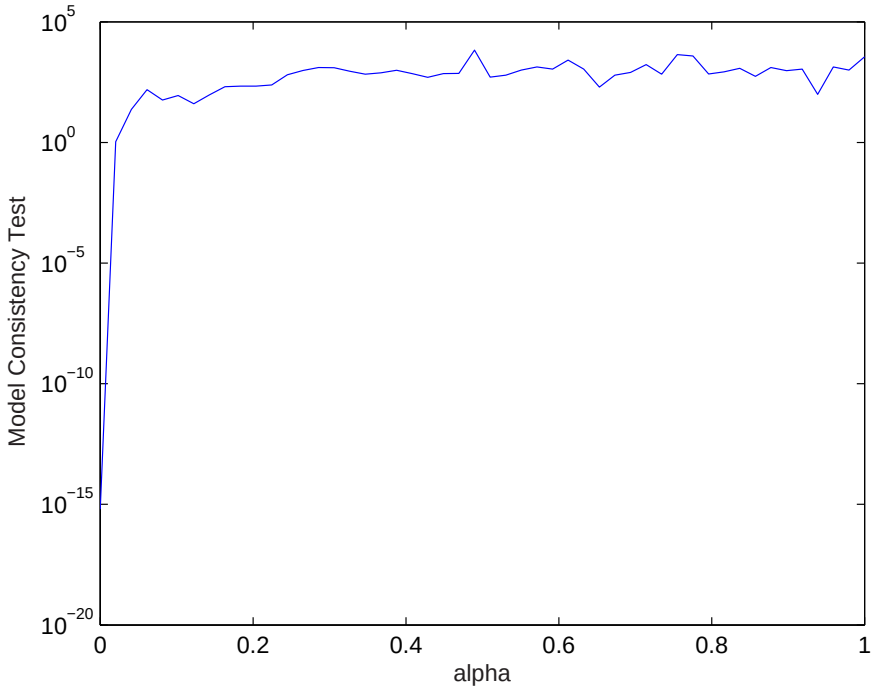

FIG. 5.1. *Problem Consistency Test*

5.2. We use two grids, corresponding to $n = 63$ and $n = 31$. To construct the test matrices, we first use the matrix of eigenvectors V to diagonalize the Laplacian with the eigenvalues ordered from large to small. We will write the resulting diagonal matrix as

$$V^T G^h V = \begin{pmatrix} D_h & 0 \\ 0 & D_H \end{pmatrix}.$$

The test matrices will have the form

$$G^h(\alpha) \equiv V \begin{pmatrix} D_h & \alpha R \\ \alpha R^T & D_H \end{pmatrix} V^T,$$

where α is a scalar, and R is a random matrix obtained using the Matlab commands `randn('state',1000), R = randn(n1,n2), R = R/norm(R)`. Here `n1` and `n2` are the dimensions of the off-diagonal block of $G^h(\alpha)$. We perform a series of experiments for an increasing sequence of positive values of α.

As α increases, at some point $G^h(\alpha)$ will no longer be positive definite. As mentioned in Section 3.4, if we limit our attention to positive-definite matrices, then that will seriously limit the size of the off-diagonal entries in $G^h(\alpha)$, and limit our ability to evaluate this diagnostic test. Large off-diagonal entries may only occur at points far from the solution of the optimization problem, but we still believe that it is valuable to have information on separability even under these circumstances.

We will perform two sets of tests. One will use the the user-provided update and downdate procedures, the other will use Fourier transforms to perform "exact" update

and downdate operations. As mentioned in Section 3.4, the user-provided procedures are likely to be based on interpolation, and will result in some mixing of frequencies. This mixing limits the sensitivity of the separability test. In some applications it may be feasible to use Fourier transforms or comparable techniques to isolate frequencies and produce a more sensitive diagnostic test, if desired.

The results are illustrated in Figure 5.4. As can be seen, the separability test using "exact" update and downdate operators is ideal, in the sense that there is a near-linear relationship between the norm of the off-diagonal perturbation and the value of the separability measure. If the user-supplied update and downdate operators are used, the test is not quite as sensitive. It is less able to detect very small off-diagonal perturbations, but is successful at detecting larger off-diagonal perturbations. In this case, the use-supplied update and downdate operators can detect off-diagonal perturbations of a magnitude corresponding to about 2% of the norm of the matrix.

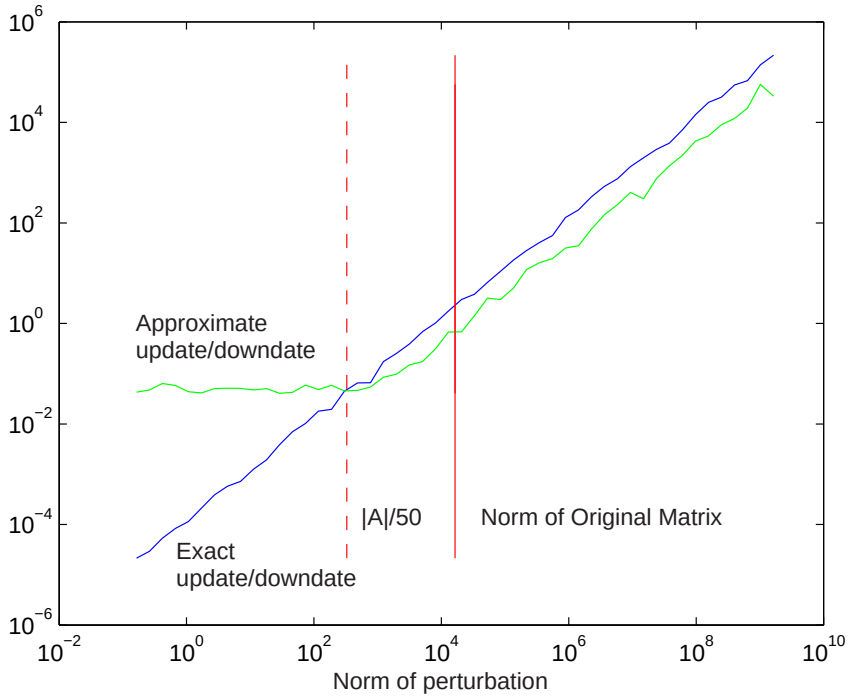

FIG. 5.2. *Separability Test*

6. Conclusions. Optimization-based multigrid algorithms such as MG/Opt can be a powerful technique for solving large and challenging optimization problems, but they are not appropriate for all optimization problems. We have demonstrated that the performance of such algorithms depends on four properties of the optimization problem: nonlinearity, grid complementarity, problem consistency, and separability along frequency scales. In addition, we have developed computational tests that can be used to assess whether the optimization problem has the desired properties. These computational tests are "cheap" since they only require modest computational effort beyond that required by the optimization method itself, and do not require any

additional information about the optimization problem beyond that already provided by the user. Further, the tests can be applied while the optimization problem is being solved, and thus can be useful even if the characteristics of the optimization problem change from one point to another. In addition to determining whether multigrid is an appropriate algorithm for a particular optimization problem, the tests identify relevant properties of the optimization problem, and thus might be used to guide a user to an optimization algorithm that is better suited to the particular optimization problem. In the long term, it might be possible to automate this process, and have optimization software automatically identify an appropriate algorithm to apply to a particular optimization problem to achieve good performance and efficiency.

7. Acknowledgements. We would like to thank Ray Tuminaro for pointing us toward relevant results from algebraic multigrid. The contributions of Stephen G. Nash were based on work supported by the National Science Foundation, while he was working at the Foundation.

REFERENCES

[1] A. Borzì and U. Hohenester, *Multigrid optimization schemes for solving Bose-Einstein condensates control problems*, SIAM Journal on Scientific Computing, 30 (2008), pp. 441–462.
[2] A. Brandt, *Multi-level adaptive solutions to boundary-value problems*, Mathematics of Computation, 31 (1977), pp. 333–390.
[3] J. Brannick and L. Zikatanov, *Algebraic multigrid methods based on compatible relaxation and energy minimization*, Tech. Report AM304, Center for Computational Mathematics and Applications, The Pennsylvania State University, University Park, PA, 2006.
[4] M. Brezina, R. Falgout, S. MacLachlan, T. Manteuffel, S. McCormick, and J. Ruge, *Adaptive algebraic multigrid*, SIAM Journal on Scientific Computing, 27 (2006), pp. 1261–1286.
[5] J. Chinneck, *Analyzing mathematical programs using MProbe*, Annals of Operations Research, 104 (2001), pp. 33–48.
[6] A. R. Conn, N. I. M. Gould, and P. L. Toint, *Trust-Region Methods*, vol. 1 of MPS/SIAM Series on Optimization, SIAM, Philadelphia, 2000.
[7] G. Cornuéjols, M. Karamanov, and Y. Li, *Early estimates of the size of branch-and-bound trees*, INFORMS Journal on Computing, 18 (2006), pp. 86–96.
[8] S. C. Eisenstat and H. F. Walker, *Choosing the forcing terms in an inexact newton method*, SIAM Journal on Scientific Computing, 4 (1996), pp. 16–32.
[9] G. H. Golub and C. V. Loan, *Matrix Computations*, Johns Hopkins University Press, Baltimore, 1996.
[10] L. Hageman and R. B. Kellogg, *Estimating optimum overrelaxation parameters*, Mathematics of Computation, 22 (1968), pp. 60–68.
[11] T. G. Kolda, R. M. Lewis, and V. Torczon, *Optimization by direct search: New perspectives on some classical and modern methods*, SIAM Review, 45 (2003), pp. 385–482.
[12] R. M. Lewis and S. G. Nash, *A multigrid approach to the optimization of systems governed by differential equations*. AIAA Paper 2000-4890, 2000.
[13] ———, *Practical aspects of multiscale optimization methods for vlsicad*, in Multiscale Optimization and VLSI/CAD, J. Cong and J. R. Shinnerl, eds., Kluwer Academic Publishers, Boston, 2002, pp. 265–291.
[14] ———, *Model problems for the multigrid optimization of systems governed by differential equations*, SIAM Journal on Scientific Computing, 26 (2005), pp. 1811–1837.
[15] T. Manteuffel, *Adaptive procedure for estimating parameters for the nonsymmetric tchebychev iteration*, Numerische Mathematik, 31 (1978), pp. 183–208.
[16] R. Mathias, *Quadratic residual bounds for the Hermitian eigenvalue problem*, SIAM Journal on Matrix Analysis, (1998), pp. 541–550.
[17] S. F. McCormick, ed., *Multigrid Methods*, Frontiers in Applied Mathematics, Society for Industrial and Applied Mathematics, Philadelphia, 1987.
[18] ———, *Multilevel Adaptive Methods for Partial Differential Equations*, Society for Industrial and Applied Mathematics, 1989.

[19] S. G. Nash, *Newton-type minimization via the Lanczos method*, SIAM Journal on Numerical Analysis, 21 (1984), pp. 770–788.
[20] ———, *A multigrid approach to discretized optimization problems*, Journal of Computational and Applied Mathematics, 14 (2000), pp. 99–116.
[21] 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.
[22] S. G. Nash and A. Sofer, *Linear and Nonlinear Programming*, McGraw–Hill, New York, 1996.
[23] I. Nenov, D. Fylstra, and L. Kolev, *Convexity determination in the Microsoft Excel solver using automatic differentiation techniques*, tech. report, Frontline Systems Inc., Incline Village, NV, 2004.
[24] C. C. Paige and M. A. Saunders, *Solution of sparse indefinite systems of linear equations*, SIAM Journal on Numerical Analysis, 12 (1975), pp. 617–629.