A Line search Multigrid Method for Nonconvex Optimization

Zaiwen Wen † Donald Goldfarb †

Abstract

We present a line search multigrid method for solving discretized versions of general unconstrained infinite dimensional optimization problems. Introducing a new condition to a backtracking line search procedure, the step generated from the coarser levels is guaranteed to be a descent direction. This method is globally convergent under fairly minimal requirements on the minimization method used at all grid levels. In particular, our method does not require that these minimizations, or so-called "smoothing" steps, be taken at each grid level in contrast with multigrid algorithms for PDEs, which fail to converge without such steps. Preliminary numerical experiments show that our method is promising.

1 Introduction

Infinite dimensional optimization problems are a major source of large-scale finite dimensional optimization problems [9]. Since it is not possible or very hard to obtain explicit solutions for these problems, they are usually solved numerically either by an "optimize-then-discretize" strategy or a "discretize-then-optimize" strategy. In this paper, we consider problem

$$\min_{\mathbf{x}\in\mathcal{V}} \quad \mathcal{F}(\mathbf{x}), \tag{1}$$

where $\mathcal{F}$ is a mapping from an infinite-dimensional space $\mathcal{V}$ to $\mathbb{R}$. We follow the "discretize-then-optimize" strategy and propose a new class of numerical algorithms to solve the discretized version of (1) by using the idea of multigrid optimization.

Multigrid methods [5; 7; 8; 14; 23; 25; 26] are iterative methods that were originally proposed for linear elliptic partial differential equations (PDEs). In this approach, coarser grid corrections are recursively imbedded in an iterative process, in combination with so called "relaxation" or "smoothing" steps, to accelerate the convergence on the target grid. Several extensions for nonlinear PDEs have been well studied. One is the global linearization method [15; 23], which uses the multigrid method within Newton's method for nonlinear equations to solve the system of linear equations that provides the Newton step at each iteration. The second is the local linearization method, a special case of which is the full approximation scheme (FAS) [6] and the closely related nonlinear multigrid method (NMGM) [14], in which the multigrid methodology is directly applied to the original system of nonlinear equations and its corresponding system of nonlinear residual equations. A combination of global and linearization methods is studied in [27] and a projection multilevel method is proposed for quasilinear elliptic PDEs in [17; 18; 19], where the system of nonlinear equations is reformulated as a least-squares problem.

Multigrid methods for infinite dimensional optimization problems have also received considerable attention [1; 2; 4; 10]. However, until recently the essential thrust of these methods was based on employing multigrid methods for solving the nonlinear equations derived from the optimality condition of problem (2). In a new approach, Nash [16] (see also [20]) proposed a multigrid optimization framework for solving problem (2), where $f_h(x_h)$ is a convex function of x_h. A proof of the global convergence of Nash's method was given in [3]. This proof requires that at least one iteration of the optimization algorithm that is used at each level be performed either before going to a coarser level or after returning from a coarser level during a multigrid cycle. These iterations of the optimization algorithm are similar to prior smoothing or post smoothing steps in multigrid methods for PDEs. Expanding on Nash's approach, Gratton, Sartenaer and Toint [11; 12]

†Department of Industrial Engineering and Operations Research, Columbia University, NY 10027, USA.(zw2109@columbia.edu, goldfarb@columbia.edu). Research supported in part by NSF Grant DMS 06-06712, ONR Grant N000140310514 and DOE Grant GE-FG01-92ER-25126.

proposed a recursive trust region method that converges to a first-order optimal point without doing such smoothing steps at each multigrid cycle.

In this paper, we propose an abstract multigrid optimization framework for general nonconvex optimization problems. We then extend this framework to an easily implementable line search multigrid optimization method. Our algorithm depends on some "*basic*" iterative method, such as the steepest descent method or Newton's method, and uses coarser grid corrections recursively to accelerate the speed of this basic iterative scheme. Introducing a new condition to a backtracking line search procedure, the step generated from the coarser levels is guaranteed to be a descent direction. We interpret smoothing steps as steps of the basic iterative scheme; moreover our line search method does not require such steps at each multigrid cycle to guarantee global convergence, and converges at a rate that is at least R-linear in the strictly convex case. Using the limited memory "BFGS" method as the basic iterative scheme, our multigrid method is able to solve very large scale problems efficiently.

The main challenge in the traditional multigrid methods is the design of appropriate smoothers, coarse-grid operators, and inter-grid transfers to achieve convergence and optimal complexity. Although these features might be problem-specific and resistant to generic prescriptions, applying the idea of multigrid to optimizing the variational problems corresponding the PDEs might provide us with opportunities for designing new efficient algorithms. Specifically, optimization techniques are usually far superior for solving minimization problems than equation solving methods applied to the necessary conditions for such problems. This paper can be viewed as a generalization of the FAS scheme applied to problems that have a variational formulation.

This paper is organized as follows. In section 2, we explain briefly an abstract multigrid optimization framework from the point view of subsapce techniques. Then we state an easily implementable line search procedure for the multigrid method to solve unconstrained nonconvex problems coarser level model under which the abstract multigrid optimization framework is practical. Results on global convergence for general nonconvex functions as well as R-linear convergence for uniformly convex problems are also presented. Preliminary numerical results are given in Section 3.

2 A Line Search Multigrid Method

We first introduce an abstract discretization scheme of Problem (1). Let $\mathcal{V}_h$ be a standard finite element space of $\mathcal{V}$ with a basis $\{\phi_h^{(j)}\}_{j=1}^{n_h}$ at grid level h, where n_h is the dimension of $\mathcal{V}_h$. In our framework, for consecutive coarser levels, we choose nested spaces, so that $\mathcal{V}_{\mathrm{N}_0} \subset \cdots \subset \mathcal{V}_{\mathrm{N}-1} \subset \mathcal{V}_{\mathrm{N}} \subset \mathcal{V}$. Given $\mathbf{x}_h \in \mathcal{V}_h$, there exists a vector $x_h = (x_h^{(1)}, \cdots, x_h^{(n_h)})^\top \in \mathbb{R}^{n_h}$ such that $\mathbf{x}_h = \sum_{j=1}^{n_h} x_h^{(j)} \phi_h^{(j)}$. Then a discrete functional f_h is obtained by defining $f_h(x_h) := \mathcal{F}(\mathbf{x}_h)$. Therefore, the discretized version of problem (1) on level h is

$$\min_{x_h} f_h(x_h), \tag{2}$$

and the main purpose of this paper is to design a multigrid method for the uppermost finest level problem:

$$\min_{x_{\mathrm{N}}} f_{\mathrm{N}}(x_{\mathrm{N}}). \tag{3}$$

Define R_h to be the restriction operator from level h to level H and P_h be the prolongation operator from level H to level h. As in standard multigrid methods, we assume that:

Assumption 2.1. *The prolongation operator P_h and the restriction operator R_h satisfy:*

$$\sigma_h P_h = R_h^\top. \tag{4}$$

For simplicity, we take $\sigma_h = 1$, which does not affect our convergence analysis.

The mechanism of line search algorithms is to iteratively generate a descent search direction and then search along this direction for a step size. Specifically, starting from the point $x_{h,k} \in \mathcal{V}_h$ on level h, a line search algorithm determines the next point as

$$x_{h,k+1} = x_{h,k} + \alpha_{h,k} d_{h,k}, \quad d_{h,k} \in \mathcal{V}_h, \tag{5}$$

where $d_{h,k}$ is the search direction and $\alpha_{h,k}$ is the step size. In addition to constructing directly a search direction $\mathbf{d}_{h,k}$ in the finer space $\mathcal{V}_h$, which is associated with a problem of dimension n_h, multigrid optimization also considers the possibility of generating a direction in the subspace $\mathcal{V}_H \subset \mathcal{V}_h$ as

$$d_{h,k} = \arg \min_{d_H \in \mathcal{V}_H} f_h(x_{h,k} + P_h d_H), \tag{6}$$

which is a problem of dimension $n_H < n_h$. By carefully exploring as many as possible smaller size problems on coarser levels to provide a good search direction on finer levels, multigrid optimization is often able to reduce the overall computational cost dramatically.

We introduce below an general algorithmic framework of multigrid optimization. Our multigrid algorithm alternates between two kinds of search directions, a "*direct search*" direction, which is generated on the current level, and a "*recursive search*" direction, which is generated from the coarser levels. The construction of these search directions depends on a "*basic*" iterative scheme on a single level, for example, the steepest descent method or Newton's method. As we will see later, most of the "real" computational cost of constructing a recursive direction comes from this basic iterative scheme. However, the coarser level model (6) might be expensive since all level H calculations not only depends on grid H, but also on grid h. Hence, we describe next a cheaper scheme [16; 20] to approximate the coarse level problem (6).

To ensure convergence and efficiency, certain coherence between the fine level problem and the corresponding coarse level problem should be enforced. The coarse level problem is not a simply discretized problem (2) for the coarse level H, but rather:

$$\psi_H(x_H) = f_H(x_H) - (v_H)^\top x_H, \tag{7}$$

where $v_H = \nabla f_{H,0} - R_h g_{h,k}$ and we have used the notation that $g_{h,k} = \nabla \psi_{h,k} = \nabla \psi_h(x_{h,k})$. Furthermore, if we define $v_{\mathrm{N}} = 0$, the model (7) can be naturally extended to all levels and the uppermost level model problem is exactly problem (3). The function ψ_H depends on the point $x_{h,k}$ and the level h and will be different for different points. To simplify our notation, we omit such dependence on $\psi_H(\cdot)$'s parent level and iteration point, hopefully, without introducing any confusion. The same is true for all other quantities such as derivatives of the coarse level model ψ_H. Actually, the function (7) is a generalization of the coarse-grid correction equation of the FAS scheme in the context of optimization by noting the equivalence between the residual equation in the FAS scheme and the gradient of (7).

We now specify conditions for when to choose a direct search direction on the current level. Specifically, $d_{h,k}$ is computed directly on level h if

$$\|R_h g_{h,k}\| < \kappa \|g_{h,k}\| \text{ or } \|R_h g_{h,k}\| < \epsilon_h \tag{8}$$

holds, where $\kappa > 0$ and ϵ_h are positive small constants. The reason for this is that $R_h g_{h,k}$ may be zero while $g_{h,k}$ is not if $g_{h,k}$ lies in the null space of R_h; hence the current iterate appears to be a stationary point for ψ_H whereas it is not for ψ_h. These conditions were first used in the multigrid algorithm proposed in [11; 12; 13]. Many unconstrained optimization algorithms can be used to compute a direct search direction. In particular, we are able to prove global convergence if this direction satisfies certain conditions, for example:

Condition 2.2. *The step direction $d_{h,k}$ satisfies*

$$\|d_{h,k}\| \le \beta_{\mathcal{T}} \|g_{h,k}\| \text{ and } -(d_{h,k})^\top g_{h,k} \ge \eta_{\mathcal{T}} \|g_{h,k}\|^2, \tag{9}$$

where $\delta_{\mathcal{T}}$ and $\eta_{\mathcal{T}}$ are positive constants.

If a "*recursive search*" direction is chosen, we first move to the next coarsest level H with an initial point $x_{H,0} = R_h x_{h,k}$. Next we compute the minimizer (or approximate minimizer) x_{H,i^*} of the coarse level problem

$$\min_{x_H} \psi_H(x_H), \tag{10}$$

where ψ_H is defined by (7). We compute the cumulative direction

$$\widetilde{d}_{H,i^*} = x_{H,i^*} - x_{H,0} = \sum_{i=0}^{i^*-1} \alpha_{H,i} d_{H,i}, \tag{11}$$

where $\alpha_{H,i}$ and $d_{H,i}$ are the step size and search direction, respectively, for the ith iteration on level H. Here each search direction $d_{H,i}$ from $x_{H,i}$ to $x_{H,i+1}$ for $i = 0, \cdots, i^* - 1$ is also computed recursively whenever possible. Then we project the direction $\widetilde{d}_{H,i^*}$ on level H back to level h to obtain the recursive search direction

$$d_{h,k} = P_h \widetilde{d}_{H,i^*}. \tag{12}$$

We state some properties of this recursive scheme as follows.

Lemma 2.3. *If the minimization on the coarse level H starts from the initial point $x_{H,0}$ and stops at x_{H,i^*}, and the recursive direction is defined as $d_{h,k} = P_h \widetilde{d}_{H,i^*}$, where $\widetilde{d}_{H,i^*} = x_{H,i^*} - x_{H,0}$, then the problems of the two consecutive levels h and H are first-order coherent in the sense that*

$$g_{H,0} = R_h g_{h,k}, \quad (d_{h,k})^\top g_{h,k} = (\widetilde{d}_{H,i^*})^\top g_{H,0}. \tag{13}$$

Suppose $f_H(x_H)$ is a convex function and $\psi_H(x_{H,i^*}) < \psi_H(x_{H,0})$, then $d_{h,k}$ is a descent direction; that is $(d_{h,k})^\top g_{h,k} < 0$. Moreover, the directional derivative $(d_{h,k})^\top g_{h,k}$ satisfies

$$-(d_{h,k})^\top g_{h,k} \geq \psi_{H,0} - \psi_{H,i^*}. \tag{14}$$

Although Lemma 2.3 shows that the recursive direction $d_{h,k}$ defined by (12) is a descent direction for convex problems, this might not be the case for general nonconvex problems. In fact, we might have that $\psi_h(x_{h,k} + d_{h,k}) > \psi_h(x_{h,k})$ because of the way that the coarser level model problem (7) is constructed. To see that $(\nabla \psi_{H,0})^\top (x_{H,i^*} - x_{H,0}) > 0$ can happen, it suffices to give an example on a single level. Consider the "Rosenbrock" function $\varphi(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$ with a local minimizer $x^* = (1, 1)^\top$. Starting from an initial point $x_0 = (-0.5, 0.5)^\top$, we can reach x^* if we "fully" search along the direction $x^* - x_0$. However, the direction $x^* - x_0$ is not a descent direction since $\nabla \varphi(x_0)^\top (x^* - x_0) = (47, 50)(1.5, 0.5)^\top = 95.5 > 0$ and traditional line search procedures cannot be used along this direction.

We now describe our line search procedure based on a backtracking line search scheme [21; 22]. Choose two constants ρ_1 and ρ_2 such that $0 < \rho_1 < \frac{1}{2}$ and $1 - \rho_1 \leq \rho_2 \leq 1$. If h is the finest level, we choose a step size $\alpha_{h,k}$ along $d_{h,k}$ that satisfies the Armijo condition

$$\psi_h(x_{h,k} + \alpha_{h,k} d_{h,k}) \leq \psi_{h,k} + \rho_1 \alpha_{h,k} (g_{h,k})^\top d_{h,k}. \tag{15}$$

If h is a coarser level, we choose a step size $\alpha_{h,k}$ along $d_{h,k}$ that satisfies the Armijo condition (15) as well as the condition

$$\psi_h(x_{h,k} + \alpha_{h,k} d_{h,k}) > \psi_{h,0} + \rho_2 g_{h,0}^\top (x_{h,k} + \alpha_{h,k} d_{h,k} - x_{h,0}). \tag{16}$$

Note that the condition (16) is similar to the Goldstein rule if $k = 0$ and $\rho_2 = 1 - \rho_1$. To select a step size $\alpha_{h,k}$ to satisfy these conditions, we use the traditional backtracking scheme stated as follows.

Algorithm 1. *Backtracking Line Search*

Step 1. Given $\alpha_\rho > 0$. Let $\alpha^{(0)} = \alpha_\rho$. Set $l = 0$.

Step 2. If ($h =$ N and condition (15) is satisfied) or if ($h <$ N and both conditions (15) and (16) are satisfied), RETURN $\alpha_{h,k} = \alpha^{(l)}$.

Step 3. Set $\alpha^{(l+1)} = \tau \alpha^{(l)}$, where $\tau \in (0, 1)$. Set $l = l + 1$ and go to Step 2.

We now are ready to state our line search multigrid method in Algorithm 2. It might happen that the acceptable step size $\alpha_{H,k} < \xi$, where ξ is a small constant. Then the minimization sequence might no longer

make significant progress if it keeps working on the current level. Besides, we have already seen that even a local minimizer might not provide us a descent direction. Therefore, it is reasonable to terminate current minimization sequence and return to the finer level.

Algorithm 2. $x_h = MNLS(h, x_{h,0}, \tilde{g}_{h,0})$

Step 1. Given $\kappa > 0$, $\epsilon_h > 0$ and $\xi > 0$ and an integer K.

Step 2. IF $h <$ N, compute $v_h = \nabla f_{h,0} - \tilde{g}_{h,0}$, set $g_{h,0} = \tilde{g}_{h,0}$;
ELSE set $v_h = 0$ and compute $g_{h,0} = \nabla f_{h,0}$.

Step 3. FOR $k = 0, 1, 2, \cdots$

3.1. IF $\|g_{h,k}\| \le \epsilon_h$ or IF $h <$ N and $k \ge$ K,
RETURN solution $x_{h,k}$;

3.2. IF $h =$ N$_0$ or $\|R_h g_{h,k}\| < \kappa \|g_{h,k}\|$ or $\|R_h g_{h,k}\| < \epsilon_h$
-Direct Search Direction Computation. Compute a descent search direction $d_{h,k}$ on the current level.

ELSE
-Recursive Search Direction Computation.
Call $x_{h-1,i^*} = MNLS(h-1, R_h x_{h,k}, R_h g_{h,k})$ to return a solution (or approximate solution) x_{h-1,i^*} of "$\min_{x_{h-1}} \psi_{h-1}(x_{h-1})$".
Compute $d_{h,k} = P_h \widetilde{d}_{h-1,i^*} = P_h(x_{h-1,i^*} - R_h x_{h,k})$.

3.3. Call backtracking line search to obtain a step size $\alpha_{h,k}$ to satisfy the condition (15) if $h =$ N or to satisfy conditions (15) and (16) if $h <$ N.

3.4. Set $x_{h,k+1} = x_{h,k} + \alpha_{h,k} d_{h,k}$. IF $\alpha_{h,k} \le \xi$ and $h <$ N, RETURN solution $x_{h,k+1}$.

The following lemma tells us that if condition (16) holds for the step size on the coarser level, then a descent recursive search step is well defined which enables us to do a line search on the finer level.

Lemma 2.4. *Suppose the iteration starts from the fine level h at $x_{h,k}$ and a recursive search direction is computed by Algorithm 2. If condition* (16) *holds on the coarse level H from iteration $i = 0$ to i^*, then the recursive step $d_{h,k}^{(i)} = P_h \widetilde{d}_{H,i}$, where $\widetilde{d}_{H,i} = x_{H,i} - x_{H,0}$, is a descent direction such that*

$$-(g_{h,k})^\top d_{h,k}^{(i)} > \rho_2^{-1}(\psi_{H,0} - \psi_{H,i}). \tag{17}$$

Proof. Since the condition (16) can be rewritten as $\psi_H(x_{H,i}) > \psi_{H,0} + \rho_2 g_{H,0}^\top \widetilde{d}_{H,i}$, and the fact that $g_{H,0}^\top \widetilde{d}_{H,i} = (R_h g_{h,k})^\top \widetilde{d}_{H,i} = g_{h,k}^\top P_h \widetilde{d}_{H,i}$, it follows that the direction $d_{h,k}^{(i)} = P_h \widetilde{d}_{H,i}$ is a descent direction on level h satisfying (17). □

Now, we prove that there exist a step size satisfies both condition (15) and (16).

Lemma 2.5. *Suppose $\psi_h(x_h)$ is continuously differentiable. Let $d_{h,k}$ be a descent direction at $x_{h,k}$ with $(g_{h,0})^\top d_{h,k} \neq 0$, and assume that ψ_h is bounded below along the ray $\{x_{h,k} + \alpha d_{h,k} \,|\, \alpha > 0\}$ for all $k \ge 0$. Then if $0 < \rho_1 < \frac{1}{2}$ and $1 - \rho_1 \le \rho_2 \le 1$, there exist intervals of step lengths satisfying both condition* (15) *and* (16) *for all $k \ge 0$.*

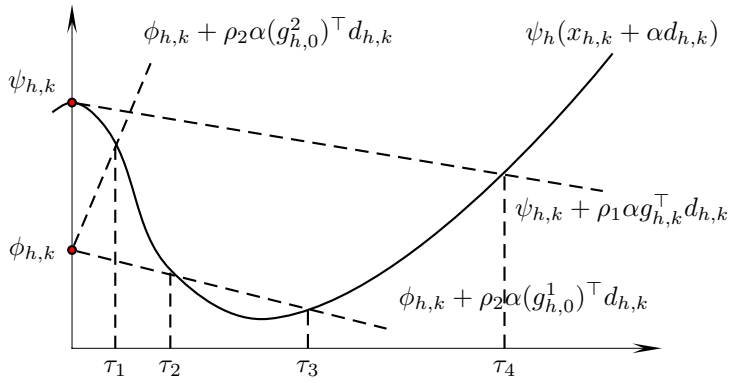
Proof. Since $y_k(\alpha) = \psi_h(x_{h,k} + \alpha d_{h,k})$ is bounded below for all $\alpha > 0$ and since $0 < \rho < \frac{1}{2}$, the line $l_k(\alpha) = \psi_{h,k} + \rho\alpha(g_{h,k})^\top d_{h,k}$ must intersect the graph of $y_k(\alpha)$ at least once. Let $\tau > 0$ be the smallest intersecting value of α. Define the term

$$\phi_{h,k} \overset{def}{=} \psi_{h,0} + \rho_2 g_{h,0}^\top (x_{h,k} - x_{h,0}), \tag{18}$$

then condition (16) can be rewritten as

$$\psi_h(x_{h,k} + \alpha_{h,k} d_{h,k}) > \phi_{h,k} + \alpha_{h,k} \rho_2 g_{h,0}^\top d_{h,k}. \tag{19}$$

Figure 1: An illustration of the line search procedure. If $g_{h,0} = g^1_{h,0}$, then the acceptable interval is $[0, \tau_2) \cup (\tau_3, \tau_4]$. If $g_{h,0} = g^2_{h,0}$, then the acceptable interval is $[0, \tau_1)$



If $k = 0$, then $\psi_{h,k} = \phi_{h,k}$. The line $\hat{l}_0(\alpha) = \psi_{h,0} + \rho_2\alpha(g_{h,0})^\top d_{h,0}$ lies below the line $l_0(\alpha)$ since $\rho_1 < \frac{1}{2} < 1 - \rho_1 \le \rho_2 \le 1$. Therefore, the line $\hat{l}_0(\alpha)$ must intersect with the graph of $y_0(\alpha)$ at least once with the smallest intersecting value $0 < \tau' < \tau$ and $(\tau', \tau]$ is a interval in which both conditions hold.

Suppose the lemma holds for iteration $k-1$, we now prove it will also holds for iteration k. Since condition (16) is satisfied at iteration $(h, k-1)$, we obtain that

$$\begin{aligned} \psi_{h,k} &= \psi_h(x_{h,k-1} + \alpha_{h,k-1} d_{h,k-1}) \\ &> \psi_{h,0} + \rho_2 g^\top_{h,0}(x_{h,k-1} + \alpha_{h,k-1} d_{h,k-1} - x_{h,0}) \\ &= \phi_{h,k}, \end{aligned}$$

which implies that there is an interval $[0, \tau''')$ with $\tau''' > 0$ ($\tau''' = \infty$ is possible), in which the line $\hat{l}_k(\alpha) = \phi_{h,k} + \alpha\rho_2 g^\top_{h,0} d_{h,k}$ lies below the graph of $y_k(\alpha)$ no matter what the slope of $\hat{l}_k(\alpha)$, $g^\top_{h,0} d_{h,k}$, is. If $\hat{l}_k(\alpha)$ intersects $y_k(\alpha)$ at least once with the smallest intersecting value $0 < \tau'' < \tau$, then $[0, \tau'')$ is an interval in which both conditions hold. Otherwise, the line $\hat{l}_k(\alpha)$ lies below $y_k(\alpha)$ in $[0, \tau)$, and this interval satisfies the requirement. Figure 1 illustrate the possible acceptable intervals for the step size α. If $g_{h,0} = g^1_{h,0}$, $[0, \tau_2)$ is an acceptable interval; the interval $(\tau_3, \tau_4]$ is also acceptable. If $g_{h,0} = g^2_{h,0}$, $[0, \tau_1)$ is an acceptable interval. □

We now briefly state the global convergence of Algorithm 2 and the detailed proof is referred to [24]. We define a constant

$$\varpi \stackrel{def}{=} \max\{1, \max_{i=\mathrm{N}_0,\cdots,\mathrm{N}} \|P_i\|, \max_{i=\mathrm{N}_0,\cdots,\mathrm{N}} \|R_i\|\} < \infty. \tag{20}$$

While the step sizes can be bounded from below by a constant in the uniformly convex case, the step size of the first iteration of each minimization sequence on the coarser levels can be bounded from below by the norm of gradient raised to some power in the general case. Hence, under some conditions the minimization sequence generated by Algorithm 2 on the uppermost finest level is globally convergent whereas the minimization sequences on all other coarser levels are either globally convergent or stop after at most K steps.

Theorem 2.6. *([24]) Assume*

1. *Condition 2.2 is satisfied by all direct search steps;*
2. *the level set $\mathcal{D}_h = \{x_h : \psi_h(x_h) \le \psi_h(x_{h,0})\}$ is bounded;*
3. *the objective function ψ_h is continuously differentiable and the gradient $\nabla\psi_h$ is Lipschitz continuous, i.e., there exists a constant $L > 0$ such that*

$$\|\nabla\psi_h(x_h) - \nabla\psi_h(\tilde{x}_h)\| \le L\|x_h - \tilde{x}_h\|, \text{ for all } x_h, \tilde{x}_h \in \mathcal{D}_h.$$

Then in Algorithm 2 with $1-\rho_1 \le \rho_2 < 1$ *at the uppermost level* $\lim_{k\to\infty}\|\nabla f_{\mathrm{N}}(x_{\mathrm{N},k})\| = 0$.

Theorem 2.6 shows that there exists a subsequence of $\{x_{\mathrm{N},k}\}$ converging to a stationary point x_{N}^* of problem (3). Since the sequence $f_{\mathrm{N}}(x_{\mathrm{N},k})$ converges, every accumulation point of $\{x_{\mathrm{N},k}\}$ is a global optimal solution of problem (3) if f_{N} is convex. Furthermore, the rate of convergence of Algorithm 2 is at least R-linear if the functions $f_h(x)$ are uniformly convex.

Theorem 2.7. *([24]) Assume* $f_h(x)$ *is twice continuously differentiable and uniformly convex; that is, there exist constants* $0 < \chi_h < M_h < \infty$ *such that* $\chi_h\|d\|_2^2 \le d^\top\nabla^2 f_h(x)d \le M_h\|d\|_2^2, \forall d \in \Re^{n_h}$, *for all* $x \in \{x \mid f_h(x_h) \le f_h(x_{h,0})\}$. *Suppose Condition 2.2 is satisfied by all direct search steps and assume that the iterative sequence* $\{x_{\mathrm{N},k}\}$ *generated by Algorithm 2 with* $\rho_2 = 1$ *at the uppermost level converges to the unique minimizer* $\{x_{\mathrm{N}}^*\}$ *of* $f_{\mathrm{N}}(x_{\mathrm{N}})$. *Then the rate of convergence is at least R-linear, i.e,*

$$\|x_{\mathrm{N},k} - x_{\mathrm{N}}^*\| \le \sqrt{\frac{2}{\chi}}(1-\alpha^*\eta_{\mathrm{N}}\chi)^{\frac{k}{2}}(f_{\mathrm{N}}(x_{\mathrm{N},0}) - f_{\mathrm{N}}(x_{\mathrm{N}}^*))^{\frac{1}{2}},$$

where

$$\alpha^* = \min\left\{\alpha_\rho, \frac{2\tau(1-\rho_1)\eta_{\mathcal{T}}}{M\beta_{\mathcal{T}}^2}, \frac{2\tau c_1(1-\rho_1)}{M\mathrm{K}\varpi^2}\right\}, \quad \eta_{\mathrm{N}} = \left(\alpha^*\rho_1\kappa^2\right)^{\mathrm{N}-\mathrm{N}_0}\eta_{\mathcal{T}},$$

and $\chi = \min_h\{\chi_h\}, M = \max_h\{M_h\}$. *For any* $\epsilon > 0$, *after at most*

$$\tau = \frac{\log((f_{\mathrm{N}}(x_{\mathrm{N},0}) - f_{\mathrm{N}}(x_{\mathrm{N}}^*))/\epsilon)}{\log(1/c)}$$

iterations, where $0 < c = 1 - \frac{\chi\alpha^*\eta_{\mathrm{N}}}{2} < 1$, *we have* $f_{\mathrm{N}}(x_{\mathrm{N},k}) - f_{\mathrm{N}}(x_{\mathrm{N}}^*) \le \epsilon$.

Remark 2.8. *The analysis of convergence is based on the reduction of the objective function values rather than constructing a contraction operator which is highly depends on smoothers and inter-grid operators.*

3 Numerical Tests

In this section, we demonstrate the effectiveness of our multigrid approach by solving problem

$$\min \mathcal{F}(u) = \int_\Omega \frac{1}{2}|\nabla u|^2 - \lambda(ue^u - e^u) - uf\ dx, \tag{21}$$

with $u = 0$ on $\partial\Omega$, which is corresponding to the nonlinear PDE [15]:

$$\begin{aligned} -\Delta u + \lambda u e^u &= f \quad \text{in} \quad \Omega, \\ u &= 0 \quad \text{on} \quad \partial\Omega, \end{aligned}$$

where $\lambda = 10$, $\Omega = [0,1]\times[0,1]$ and $f = \left(9\pi^2 + \lambda e^{((x^2-x^3)\sin(3\pi y))}(x^2-x^3) + 6x - 2\right)\sin(3\pi y)$, and the exact solution is $u = (x^2-x^3)\sin(3\pi y)$. For simplicity, we discretize Ω at level h as a square grid $\Omega_h = \{(x_i, y_j) \mid x_i = i\omega_h^x, y_j = j\omega_h^y, i = 0,\cdots,n_h^x; j = 0,1,\cdots,n_h^y\}$, where the mesh size $\omega_h^x = 1/n_h^x$ and $\omega_h^y = 1/n_h^y$ and we take $n_h^x = n_h^y = 2^h$ for the sake of simplicity. Then the objective functional is discretized as

$$F(u) = \frac{1}{2}\sum_{i=0}^{n_h^x-1}\sum_{j=0}^{n_h^y-1}\mathcal{L}(\delta_x^+u_{i,j}, \delta_y^+u_{i,j}, u_{i,j}) + \mathcal{L}(\delta_x^-u_{i,j}, \delta_y^-u_{i,j}, u_{i,j}),$$

where $\delta_x^+u_{i,j}, \delta_y^+u_{i,j}$ and $\delta_x^-u_{i,j}, \delta_y^-u_{i,j}$ are, respectively, the forward and backward finite differences with respect to x and y. In our test problems the grid spacing is set to 2^{-3} at the coarsest level $h = 3$ and to 2^{-8} at the finest level $h = 8$, which gives a 9×9 grid and a 257×257 grid, respectively.

The "basic" iterative scheme is based on the L-BFGS method [21; 22]. We compared the following three algorithms: the standard L-BFGS method applied at the finest level, denoted by "L-BFGS", without recourse

Table 1: Summary of computational costs for Problem (21)

L-BFGS					
h	nls	nfe	nge	$\|g_{\mathrm{N}}^*\|_2$	CPU
8	431	463	432	8.858409e-06	56.4617

	FMLS						MRLS					
h	3	4	5	6	7	8	3	4	5	6	7	8
nls	182	101	50	28	13	8	16	27	49	58	69	59
nfe	234	144	73	41	18	11	20	31	53	61	75	60
nge	208	114	56	32	15	9	17	28	50	59	70	60
$\|g_{\mathrm{N}}^*\|_2$	5.190493e-06						9.401884e-06					
CPU	3.052931						12.225726					

to coarse level computations; the mesh refinement technique, denoted by "MRLS", where the discretized problems are solved in turn from the coarsest level to the finest level using the standard L-BFGS method and the starting point at the finer level is obtained by prolongating the solution obtained at its next coarser level; the full multigrid method using Algorithm 2 with one smoothing step, denoted by "FMLS". When we specify that a particular version of Algorithm 2 does k smoothing steps, we mean that before considering doing a recursive step, the algorithm first takes k direct search steps. For the multigrid Algorithm 2, we set

$$\kappa = 10^{-4}, \quad \epsilon_h = 10^{-5}/5^{\mathrm{N}-h}, \quad \mathrm{K} = 100, \quad \rho_1 = 10^{-3}, \quad \rho_2 = 1 - \rho_1.$$

The number of storage m was set to 5 in the L-BFGS method. All codes were written in MATALAB (Release 7.3.0). All experiments were performed on a Dell Precision 670 workstation with an Intel Xeon 3.4GHZ CPU and 6GB of RAM.

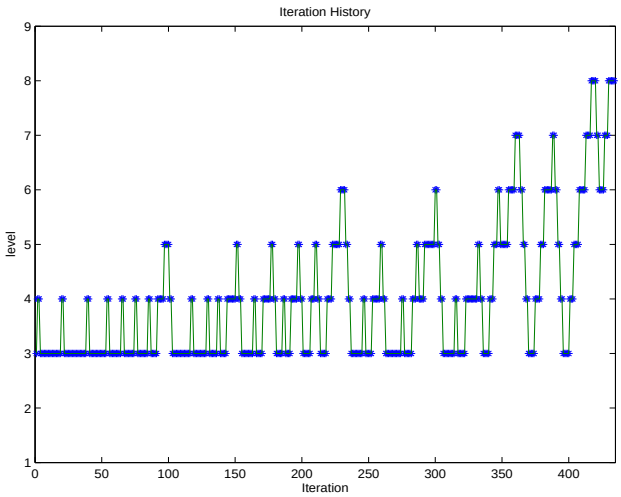
In Tables 1 we summarize the computational costs of the three methods. Specifically, h indicates the level; "nls", "nfe" and "nge" denote the total number of line searches, the total number of function evaluations and the total number of gradient evaluations at that level, respectively. We also report the total CPU time measured in seconds and the accuracy attained, which is measured by the Euclidean-norm $\|g_{\mathrm{N}}^*\|_2$ of the gradient at the final iteration. From Table 1, we can see that the "L-BFGS" method is not efficient, i.e., it terminated after 431 iterations and consumed a lot of CPU time. However, using the solutions on the coarser levels to initiate iterations on the finer levels, both the "MRLS" method and the "FMLS" method are much cheaper. The "FMLS" method took fewer iterations on the finer levels than on the coarser levels. The "FMLS" method required less functions and gradients evaluations on the finer levels than that of the "MRLS" method. This is most obvious on the finest levels, which the "FMLS" method exhibited an approximately 6-fold improvement in terms of the number of the function and evaluations gradient evaluations over the "MRLS" method. Therefore, the "FMLS" method consumed less CPU time than the "MRLS" method did, although the "FMLS" method took more iterations on the coarser levels than the "MRLS" method did.

To illustrate the multilevel behavior of the "FMLS" method, we plot the level versus iteration history for it in Figure 2. We can see that a recursive step is not always performed; it is only performed if it is necessary.

4 Discussion

In this paper, we present a new line search multigrid algorithm for general nonconvex unconstrained problems. The algorithm adopts the recursive steps as many as possible to accelerate the overall computational speed. Incorporating a new condition to a backtracking line search procedure, the recursive step is guaranteed to be a descent direction. In particular, our multigrid algorithm using the limited memory "BFGS" method as the direct search direction, though not shown to converge in theory yet, exhibits exceptional computational efficiency. Our future work includes developing "direct search" directions which are able to utilize information in history more efficiently and extending the proposed algorithmic framework to problems with constraints.

Figure 2: Level versus iteration history of the "FMLS" method for Problem (21) running with the finest level $h = 8$.



References

[1] U. M. Ascher and E. Haber, *A multigrid method for distributed parameter estimation problems*, Electron. Trans. Numer. Anal., 15 (2003), pp. 1–17 (electronic). Tenth Copper Mountain Conference on Multigrid Methods (Copper Mountain, CO, 2001).

[2] M. Benzi, E. Haber, and L. Hanson, *Multilevel algorithms for large-scale interior point methods in bound constrained optimization*, tech. rep., Department of Mathematics and Computer Science, Emory University, 2006.

[3] A. Borzì, *On the convergence of the mg/opt method*, in Proceedings GAMM Annual Meeting, no. 5, 2005, pp. 735–736.

[4] A. Borzì and K. Kunisch, *A multigrid scheme for elliptic constrained optimal control problems*, Comput. Optim. Appl., 31 (2005), pp. 309–333.

[5] J. H. Bramble, *Multigrid methods*, vol. 294 of Pitman Research Notes in Mathematics Series, Longman Scientific & Technical, Harlow, 1993.

[6] A. Brandt, *Multi-level adaptive solutions to boundary-value problems*, Math. Comp., 31 (1977), pp. 333–390.

[7] ——, *Multigrid techniques: 1984 guide with applications to fluid dynamics*, vol. 85 of GMD-Studien [GMD Studies], Gesellschaft für Mathematik und Datenverarbeitung mbH, St. Augustin, 1984.

[8] W. L. Briggs, V. E. Henson, and S. F. McCormick, *A multigrid tutorial*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, second ed., 2000.

[9] E. D. Dolan, J. J. Moré, and T. S. Munson, *Benchmarking optimization software with cops 3.0*, tech. rep., Mathematics and Computer Science Division, Argonne National Laboratory, February 2004.

[10] T. Dreyer, B. Maar, and V. Schulz, *Multigrid optimization in applications*, J. Comput. Appl. Math., 120 (2000), pp. 67–84. SQP-based direct discretization methods for practical optimal control problems.

[11] S. Gratton, A. Sartenaer, and P. Toint, *Recursive trust-region methods for multilevel nonlinear optimization (part i): Global convergence and complexity*, tech. rep., Dept of Mathematics, FUNDP, Namur (B), 2004.

[12] ——, *Numerical experience with a recursive trust-region method for multilevel nonlinear optimization*, tech. rep., Dept of Mathematics, FUNDP, Namur (B), 2006.

[13] ——, *Second-order convergence properties of trust-region methods using incomplete curvature information, with an application to multigrid optimization*, J. Comput. Math., 24 (2006), pp. 676–692.

[14] W. Hackbusch, *Multigrid methods and applications*, vol. 4 of Springer Series in Computational Math-

ematics, Springer-Verlag, Berlin, 1985.
[15] V. E. Henson, *Multigrid methods nonlinear problems: an overview*, in Computational Imaging. Edited by Bouman, Charles A.; Stevenson, Robert L. Proceedings of the SPIE, Volume 5016, pp. 36-48 (2003)., C. A. Bouman and R. L. Stevenson, eds., June 2003, pp. 36–48.
[16] R. M. Lewis and S. G. Nash, *Model problems for the multigrid optimization of systems governed by differential equations*, SIAM J. Sci. Comput., 26 (2005), pp. 1811–1837.
[17] T. A. Manteuffel, S. F. McCormick, and O. Röhrle, *Projection multilevel methods for quasilinear elliptic partial differential equations: theoretical results*, SIAM J. Numer. Anal., 44 (2006), pp. 139–152 (electronic).
[18] T. A. Manteuffel, S. F. McCormick, O. Röhrle, and J. Ruge, *Projection multilevel methods for quasilinear elliptic partial differential equations: numerical results*, SIAM J. Numer. Anal., 44 (2006), pp. 120–138 (electronic).
[19] S. F. McCormick, *Projection multilevel methods for quasi-linear PDEs: V-cycle theory*, Multiscale Model. Simul., 4 (2005), pp. 1339–1348 (electronic).
[20] S. G. Nash, *A multigrid approach to discretized optimization problems*, Optim. Methods Softw., 14 (2000), pp. 99–116. International Conference on Nonlinear Programming and Variational Inequalities (Hong Kong, 1998).
[21] J. Nocedal and S. J. Wright, *Numerical optimization*, Springer Series in Operations Research and Financial Engineering, Springer, New York, second ed., 2006.
[22] W. Sun and Y.-X. Yuan, *Optimization theory and methods*, vol. 1 of Springer Optimization and Its Applications, Springer, New York, 2006. Nonlinear programming.
[23] U. Trottenberg, C. W. Oosterlee, and A. Schüller, *Multigrid*, Academic Press Inc., San Diego, CA, 2001. With contributions by A. Brandt, P. Oswald and K. Stüben.
[24] Z. Wen and D. Goldfarb, *A line search multigrid method for large-scale nonconvex optimization*, tech. rep., Dept of IEOR, Columbia University, 2007.
[25] P. Wesseling, *An introduction to multigrid methods*, Pure and Applied Mathematics (New York), John Wileypetsc-user-ref & Sons Ltd., Chichester, 1992.
[26] J. Xu, *An introduction to multilevel methods*, in Wavelets, multilevel methods and elliptic PDEs (Leicester, 1996), Numer. Math. Sci. Comput., Oxford Univ. Press, New York, 1997, pp. 213–302.
[27] I. Yavneh and G. Dardyk, *A multilevel nonlinear method*, SIAM J. Sci. Comput., 28 (2006), pp. 24–46 (electronic).