# Convergence and Descent Properties for a Class of Multilevel Optimization Algorithms

Stephen G. Nash*

April 28, 2010

**Abstract**

I present a multilevel optimization approach (termed MG/Opt) for the solution of constrained optimization problems. The approach assumes that one has a hierarchy of models, ordered from fine to coarse, of an underlying optimization problem, and that one is interested in finding solutions at the finest level of detail. In this hierarchy of models calculations on coarser levels are less expensive, but also are of less fidelity, than calculations on finer levels. The intent of MG/Opt is to use calculations on coarser levels to accelerate the progress of the optimization on the finest level.

Global convergence (i.e., convergence to a Karush-Kuhn-Tucker point from an arbitrary starting point) is ensured by requiring a single step of a convergent method on the finest level, plus a line-search for incorporating the coarse level corrections. The convergence results apply to a broad class of algorithms with minimal assumptions about the properties of the coarse models.

I also analyze the descent properties of the algorithm, i.e., whether the coarse level correction is guaranteed to result in improvement of the fine level solution. Although additional assumptions are required to guarantee improvement, the assumptions required are likely to be satisfied by a broad range of optimization problems.

## 1 Introduction

I present MG/Opt, a multilevel optimization approach originally developed for unconstrained optimization and here extended to constrained optimization problems. It assumes that one has a hierarchy of models, ordered from fine to coarse, of an underlying optimization problem, and that one is interested in finding solutions at the finest level of detail. MG/Opt and related multilevel algorithms

*Systems Engineering and Operations Research Dept., George Mason University, Fairfax, VA 22030, USA. Email: snash@gmu.edu. The material in the paper was supported by the Department of Energy under Award DE-SC-0001691.

have been successfully used to solve a variety of unconstrained problems

$$\min_{x_h} f_h(x_h) \tag{1}$$

where the subscript $h$ refers to the level in the hierarchy of models (see, e.g., [4, 11, 13]). When applied to appropriate problems, MG/Opt is capable of achieving the excellent computational performance of multigrid algorithms applied to elliptic PDEs. MG/Opt has also been applied to optimization models with constraints in the case where the constraints are use to solve for some variables in terms of the others, resulting in a reduced problem that is effectively unconstrained [8, 14].

Here I extend MG/Opt to constrained optimization problems, allowing both equality and inequality constraints:

$$\begin{array}{ll} \displaystyle\min_{x_h} & f_h(x_h) \\ \text{subject to} & a_h(x_h) = 0 \\ & c_h(x_h) \le 0 \end{array} \tag{2}$$

I also present convergence theorems for the resulting algorithms, along with theorems showing that the search directions produced by MG/Opt are descent directions when appropriate assumptions are satisfied.

The MG/Opt algorithm for constrained problems is related to the algorithm in [7]. The convergence theorem in the unconstrained case is more general than earlier results for multilevel algorithms for unconstrained optimization [11, 14]; see also [4, 15] for convergence theorems for related multilevel algorithms. The theorems for the constrained case are new. MG/Opt is based on the principles underlying the full approximation multilevel scheme for solving nonlinear PDEs [9].

The results here provide a framework for applying multilevel approaches to a broad range of optimization models. The MG/Opt framework is general, in the sense that it does not specify the underlying optimization algorithm, providing great flexibility in how it is implemented. In particular it would be possible to choose an underlying optimization algorithm and implementation adapted to a particular optimization problem or computer architecture.

The results are developed in three stages. Unconstrained problems are considered first. These results are of independent interest, and they also illustrate the algorithm and theorems in their simplest form. Then I consider problems with equality constraints, followed by inequality constraints. The latter results are derived using the corresponding theorems for equality-constrained problems. Finally I summarize the overall algorithm for a problem with a mix of equality and inequality constraints, in a form better suited for software implementation.

## 2 Unconstrained Problems

In the unconstrained case the optimization problem is (1). To define and analyze the algorithm, it is only necessary to refer to two levels of models, with $h$

referring to the current finer level and $H$ referring to the coarser level. The algorithm requires that the user provide a downdate operator $I_h^H$ and an update operator $I_H^h$ to transform vectors from one level to the other. To specify the MG/Opt algorithm I make the following assumption

- **Assumption A1**: $\nabla f_h(x_h)$ is defined for all values of $x_h$.

Although additional assumptions will be needed to prove convergence of the algorithm, and to prove that the algorithm produces descent directions, this is the only assumption needed to define and run the algorithm. Here is the MG/Opt algorithm for an unconstrained problem.

- Given an initial estimate of the solution $x_h^0$, and integers $k_1, k_2 \geq 0$ satisfying $k_1 + k_2 > 0$, for $j = 0, 1, \ldots$ until converged:
  - *Pre-smoothing:* Apply $k_1$ iterations of a convergent optimization algorithm to (1) to obtain $\bar{x}_h$ (with $x_h^j$ used as the initial guess).
  - *Recursion:*
    - Compute $\bar{x}_H = I_h^H \bar{x}_h$ and $\bar{v}_H = \nabla f_H(\bar{x}_H) - I_h^H \nabla f_h(\bar{x}_h)$.
    - Minimize (perhaps approximately) the "surrogate" model

      $$f_s(x_H) \equiv f_H(x_H) - \bar{v}_H^T x_H$$

      to obtain $x_H^+$ (with $\bar{x}_H$ used as the initial guess). The minimization could be performed recursively by calling MG/Opt.
    - Compute the search directions $e_H = x_H^+ - \bar{x}_H$ and $e_h = I_H^h e_H$.
    - Use a line search to determine $x_h^+ = \bar{x}_h + \alpha e_h$ satisfying $f_h(x_h^+) \leq f_h(\bar{x}_h)$.
  - *Post-smoothing:* Apply $k_2$ iterations of the same convergent optimization algorithm to (1) to obtain $x_h^{j+1}$ (with $x_h^+$ used as the initial guess).

This description of MG/Opt is useful for understanding and analyzing the algorithm. The version of the algorithm in Section 5 is better suited for implementation purposes.

The vector $\bar{v}_H$ is chosen so that

$$\nabla f_s(\bar{x}_H) = I_h^H \nabla f_h(\bar{x}_h),$$

that is, the surrogate model matches the downdated fine model to first order. (It would be trivial to add a constant to the function $f_s$ so that $f_s(\bar{x}_H) = f_h(\bar{x}_h)$, ensuring that the surrogate model matched both function and gradient values. This would have little effect on the optimization algorithms.)

To prove convergence for MG/Opt, I make the following additional assumptions

- **Assumption A2**: The level set $S_h = \{x_h : f(x_h) \leq f(x_h^0)\}$ is compact, where $x_h^0$ is the initial guess of the solution of (1).

- **Assumption A3**: $\nabla^2 f_h(x_h)$ is continuous for all choices of $x_h \in S_h$.

These are standard assumptions for proving convergence of algorithms for unconstrained optimization, both for line search and trust region algorithms (see, e.g., [6]). It would be possible to prove analogous convergence results with a weaker version of assumption A3, namely that the gradient $\nabla f_h(x_h)$ is Lipschitz continuous for all choices of $x_h \in S_h$.

The MG/Opt algorithm is flexible about the choice of the convergent optimization algorithm used in the pre-smoothing and post-smoothing steps. I will examine both line search and trust region algorithms as possibilities. Let $\text{Opt}_{LS}$ be a line search algorithm, i.e., it computes a new estimate of the solution of the form

$$x_h^{j+1} = x_h^j + \alpha_j p_j$$

where $p_j$ is a search direction and $\alpha_j$ is a step length. I will assume that $\alpha_j$ is chosen to satisfy one of the Wolfe, strong Wolfe, or Goldstein conditions (see [12] for a definition of these conditions). In addition I will assume that algorithm $\text{Opt}_{LS}$ chooses the search direction $p_j$ so that is satisfies the condition

$$\frac{-p_j^T \nabla f_h(x_h^j)}{\|p_j\| \cdot \|\nabla f_h(x_h^j)\|} \geq \epsilon > 0.$$

This condition is satisfied by many algorithms. Here is a convergence theorem for $\text{Opt}_{LS}$.

**Theorem 1** *Assume that A1–A3 are satisfied. Suppose that optimization algorithm $Opt_{LS}$ is used to solve (1). Then*

$$\lim_{j \to \infty} \|\nabla f_h(x_h^j)\| = 0.$$

**Proof.** See [12]. □

It would also be possible to use a variety of trust-region methods. Suppose that $\text{Opt}_{TR}$ is the trust-region method for unconstrained optimization defined in [6]. Then we have the following theorem.

**Theorem 2** *Assume that A1–A3 are satisfied. Suppose that optimization algorithm $Opt_{TR}$ is used to solve (1). Then*

$$\lim_{j \to \infty} \|\nabla f_h(x_h^j)\| = 0.$$

**Proof.** See [6]. □

I immediately obtain the following convergence result for MG/Opt. Note that the line search used in the recursion step of the algorithm only requires that the function value not increase.

**Theorem 3** *Assume that A1, A2, and A3 are satisfied, and that either* $Opt_{LS}$ *or* $Opt_{TR}$ *is the convergent optimization algorithm used in the pre- and post-smoothing steps of MG/Opt. Then MG/Opt is guaranteed to converge in the sense that*

$$\lim_{j\to\infty} \|\nabla f_h(x_h^j)\| = 0.$$

**Proof.** Since $k_1 + k_2 > 0$, each iteration of MG/Opt includes at least one iteration of the convergent optimization algorithm applied to (1). The recursion step at worst results in no improvement to the value of the objective function. To prove convergence of MG/Opt, it is straightforward to repeat the proof of convergence for either $\mathrm{Opt}_{LS}$ [12] or $\mathrm{Opt}_{TR}$ [6], taking into account that at some iterations the new estimate of the solution has a lower function value than that obtained by the underlying optimization algorithm. □

The convergence theorem applies to a more general algorithm of the following form:

- *Pre-smoothing:* Apply $k_1$ iterations of $\mathrm{Opt}_{LS}$ or $\mathrm{Opt}_{TR}$ to (1) to obtain $\bar{x}_h$ (with $x_h^j$ used as the initial guess).
- *Recursion:*
    - Find a point $x_h^+$ satisfying $f_h(x_h^+) \le f_h(\bar{x}_h)$.
- *Post-smoothing:* Apply $k_2$ iterations of the same optimization algorithm to (1) to obtain $x_h^{j+1}$ (with $x_h^+$ used as the initial guess).

Thus convergence is guaranteed by the structure of the MG/Opt algorithm, and does not depend on the surrogate model used in the recursion step. The performance of the algorithm, however, is strongly dependent on the choices of the surrogate model and the update and downdate operators $I_H^h$ and $I_h^H$.

The MG/Opt algorithm above requires that the objective function not increase in the Recursion step. This requirement could be relaxed in the context of an optimization algorithm based on a non-monotone line search; see, for example, [5].

My next goal is to determine under what conditions the search direction $e_h$ from the recursion step of MG/Opt is guaranteed to be a descent direction for $f_h$ at $\bar{x}_h$:

$$f_h(\bar{x}_h + \epsilon e_h) < f_h(\bar{x}_h) \quad \text{for sufficiently small } \epsilon > 0;$$

or, alternatively:

$$\nabla f_h(\bar{x}_h)^T e_h < 0.$$

For this purpose I make the following additional assumption:

- **Assumption A4**: $(I_H^h)^T = C_I I_h^H$ for some constant $C_I > 0$.

If the surrogate model is minimized exactly then

$$\nabla f_H(x_H^+) = \bar{v}_H = \nabla f_H(\bar{x}_H) - I_h^H \nabla f_h(\bar{x}_h).$$

If the surrogate model is only minimized approximately then

$$\nabla f_H(x_H^+) = \nabla f_H(\bar{x}_H) - I_h^H \nabla f_h(\bar{x}_h) + z$$

for some $z$. We can write this final equation as

$$\nabla f_s(x_H^+) = z.$$

I obtain the following theorem.

**Theorem 4** *Assume that A1–A4 are satisfied, and that $\nabla f_h(\bar{x}_h) \neq 0$. Then the search direction $e_h$ from the recursion step of MG/Opt will be a descent direction for $f_h$ at $\bar{x}_h$ if (a) $\|\nabla f_s(x_H^+)\|$ is sufficiently small, and (b) if*

$$e_H^T \nabla^2 f_H(\bar{x}_H + \eta e_H) e_H > 0$$

*for $0 \leq \eta \leq 1$.*

**Proof.** We test for a descent direction as follows:

$$\begin{aligned}
\nabla f_h(\bar{x}_h)^T e_h &= \nabla f_h(\bar{x}_h)^T [I_H^h (x_H^+ - \bar{x}_H)] \\
&= C_I [I_h^H \nabla f_h(\bar{x}_h)]^T (x_H^+ - \bar{x}_H) \\
&= C_I [\nabla f_H(\bar{x}_H) - \nabla f_H(x_H^+) + z]^T (x_H^+ - \bar{x}_H) \\
&= C_I [\nabla f_H(\bar{x}_H) - \nabla f_H(x_H^+)]^T e_H + C_I z^T e_H.
\end{aligned}$$

To analyze the first term in the last formula I use the mean-value theorem. If I define the real-valued function $F(y)$ by

$$F(y) \equiv [\nabla f_H(\bar{x}_H) - \nabla f_H(y)]^T e_H$$

then

$$F(\bar{x}_H + e_H) = F(\bar{x}_H) + \nabla F(\xi)^T e_H = -e_H^T \nabla^2 f_H(\xi) e_H$$

where $\xi = \bar{x}_H + \eta e_H$ for some $0 \leq \eta \leq 1$. Thus

$$\begin{aligned}
\nabla f_h(\bar{x}_h)^T e_h &= C_I [\nabla f_H(\bar{x}_H) - \nabla f_H(x_H^+)]^T e_H + C_I z^T e_H \\
&= -C_I e_H^T \nabla^2 f_H(\xi) e_H + C_I z^T e_H.
\end{aligned}$$

The theorem follows from this last formula. □

Both the additional assumptions in the theorem are necessary. If we do not minimize the surrogate model accurately enough then the point $x_H^+$ could be almost arbitrary, so there would be no guarantee that $e_h$ would be a descent direction.

The assumption that $e_H^T \nabla^2 f_H(\bar{x}_H + \eta e_H) e_H > 0$ is also needed. In particular $e_H \neq 0$. One also needs that $\nabla^2 f_H$ is positive definite along the line segment connecting $\bar{x}_H$ and $x_H^+$. For example, consider the one-dimensional example with $\bar{v}_H = 0$

$$f_s(x_H) = f_H(x_H) = x_H^3 - x_H$$

with $\bar{x}_H = -1$ and $x_H^+ = 1/\sqrt{3}$, a local minimizer of $f_s$. Then $e_H = 1 + 1/\sqrt{3} > 0$ and $f_s'(\bar{x}_H) = 3\bar{x}_H^2 - 1 = 2 > 0$ so $e_H$ is an *ascent* direction at $\bar{x}_H$. Hence both assumptions in the theorem are necessary.

One can guarantee descent in a different way by using a variant of MG/Opt where the recursion step is modified to: Obtain $x_H^+$ by solving

$$\begin{array}{ll} \min\limits_{x_H} & f_s(x_H) \equiv f_H(x_H) - \bar{v}_H^T x_H \\ \text{subject to} & \|x_H - \bar{x}_H\| \le \Delta \end{array} \tag{3}$$

for some value of $\Delta > 0$. The following theorem is obtained.

**Theorem 5** *Assume that A1–A4 are satisfied, and that the recursion step in MG/Opt includes the constraint $\|x_H - \bar{x}_H\| \le \Delta$. If $I_h^H \nabla f_h(\bar{x}_h) \ne 0$ and $\Delta$ is sufficiently small, then $e_h$ is a descent direction for $f_h$ at $\bar{x}_h$.*

**Proof.** If $\Delta \to 0$ then, in the limit, $e_H$ is proportional to the steepest-descent direction

$$p = -\nabla f_s(\bar{x}_H) = -I_h^H \nabla f_h(\bar{x}_h),$$

where the final formula follows from the definition of $\bar{v}_H$. If $\Delta$ is sufficiently small then

$$\begin{aligned} e_h^T \nabla f_h(\bar{x}_h) &= (I_H^h e_H)^T \nabla f_h(\bar{x}_h) \\ &= C_I e_H^T I_h^H \nabla f_h(\bar{x}_h) \\ &\approx -\gamma C_I \|I_h^H \nabla f_h(\bar{x}_h)\|_2^2 < 0 \end{aligned}$$

for some positive scalar $\gamma$. □

The version of MG/Opt used in [8] includes a constraint of the form $\|x_H - \bar{x}_H\|_\infty \le \Delta$. This is equivalent to adding bound contraints on the variables in the surrogate model.

In the descent theorems, I made the assumption that the update and downdate operators satisfy

$$(I_H^h)^T = C_I I_h^H.$$

This assumption is used to guarantee that the recursion step in MG/Opt produces a descent direction. Suppose instead that

$$(I_H^h)^T = M I_h^H$$

where $M$ is a positive definite matrix. Then repeating the proof of Theorem 4 gives

$$\nabla f_h(\bar{x}_h)^T e_h = -e_H^T[\nabla^2 f_H(\xi) M] e_H + z^T M e_H.$$

Even if $\nabla^2 f_H(\xi)$ is positive definite, the product $B \equiv [\nabla^2 f_H(\xi) M]$ will be positive definite if and only if $B$ is a normal matrix [10]. For general choices of $M$ this is not guaranteed to be true.

# 3 Equality Constraints

I now consider an optimization problem with equality constraints:

$$\begin{array}{ll} \min\limits_{x_h} & f_h(x_h) \\ \text{subject to} & a_h(x_h) = 0 \end{array} \tag{4}$$

where the subscript $h$ refers to the level of the model. Also provided are an downdate operator $I_h^H$ and an update operator $I_H^h$ for the variables, as well as a downdate operator $J_h^H$ and an update operator $J_H^h$ for the constraints. In the case where the number of constraints remains the same on the fine and coarse levels, $J_h^H = J_H^h = I$. To specify the MG/Opt algorithm for this case I make the following assumption:

- **Assumption B1**: $\nabla f_h(x_h)$ and $\nabla a_h(x_h)$ are defined for all choices of $x_h$.

I define the Lagrangian function as

$$L_h(x_h, \lambda_h) = f_h(x_h) + a_h(x_h)^T \lambda_h$$

where $\lambda_h$ are Lagrange multipliers for the constraints.

As before the MG/Opt algorithm is defined in terms of a convergent optimization algorithm that can be applied to (4). Here I assume that this algorithm is based on a merit function $M_h(x_h)$. Merit functions are usually chosen in such a way that local solutions to (4) correspond to local minimizers of the merit function [12]; in some cases it is possible to prove that local minimizers of the merit function correspond to local solutions to (4) (see, e.g., [1]).

Here is the MG/Opt algorithm for an equality-constrained problem.

- Given an initial estimate of the solution $(x_h^0, \lambda_h^0)$, and integers $k_1, k_2 \geq 0$ satisfying $k_1 + k_2 > 0$, for $j = 0, 1, \ldots$ until converged:
  - *Pre-smoothing:* Apply $k_1$ iterations of a convergent optimization algorithm to (4) to obtain $(\bar{x}_h, \bar{\lambda}_h)$ (with $(x_h^j, \lambda_h^j)$ used as the initial guess), where the convergent optimization algorithm is based on a merit function $M_h(x_h)$.
  - *Recursion:*
    - ∗ Compute $\bar{x}_H = I_h^H \bar{x}_h$, $\bar{\lambda}_H = J_h^H \bar{\lambda}_h$,

      $$\bar{v}_H = \nabla_x L_H(\bar{x}_H, \bar{\lambda}_H) - I_h^H \nabla_x L_h(\bar{x}_h, \bar{\lambda}_h),$$

      and $\bar{s} = a_H(\bar{x}_H) - J_h^H a_h(\bar{x}_h)$.
    - ∗ Minimize (perhaps approximately) the "surrogate" model

      $$f_s(x_H) \equiv f_H(x_H) - \bar{v}_H^T x_H$$

subject to the "surrogate" constraints

$$a_s(x_H) \equiv a_H(x_H) - \bar{s} = 0$$

to obtain $(x_H^+, \lambda_H^+)$ (with $(\bar{x}_H, \bar{\lambda}_H)$ used as the initial guess). The minimization could be performed recursively by calling MG/Opt.

- ∗ Compute the search directions $e_H = x_H^+ - \bar{x}_H$ and $e_h = I_H^h e_H$.
- ∗ Use a line search to determine $x_h^+ = \bar{x}_h + \alpha e_h$ satisfying $M_h(x_h^+) \le M_h(\bar{x}_h)$. Compute Lagrange multiplier estimates $\lambda_h^+$.

– *Post-smoothing:* Apply $k_2$ iterations of the same convergent optimization algorithm to (4) to obtain $(x_h^{j+1}, \lambda_h^{j+1})$ (with $(x_h^+, \lambda_h^+)$ used as the initial guess).

Corresponding to the surrogate model and constraints in the recursion step, I define the surrogate Lagrangian as

$$L_s(x_H, \lambda_H) \equiv f_s(x_H, \lambda_H) + a_s(x_H)^T \lambda_H.$$

It is easy to check that

$$\nabla L_s(\bar{x}_H, \bar{\lambda}_H) = \begin{pmatrix} I_h^H & 0 \\ 0 & J_h^H \end{pmatrix} \nabla L_h(\bar{x}_h, \bar{\lambda}_h)$$

where the gradient is taken with respect to both $x$ and $\lambda$. In this sense the surrogate model is a first-order approximation to the downdated fine-level model.

Notice that the surrogate model has the same form as the original model. The objective is shifted by a linear term $\bar{v}_H^T x_H$ and the constraints are shifted by a constant vector $\bar{s}$. Thus if the original model (4) has linear constraints then so does the surrogate model. If the original objective is a quadratic function then so is the objective of the surrogate model. And so forth. Thus the same optimization algorithm can be applied to solve the surrogate model as is used in the pre- and post-smoothing steps. This is also true for the other versions of MG/Opt that I discuss.

It is possible to prove convergence for MG/Opt much as in the unconstrained case. A common approach to proving convergence for constrained optimization algorithm is to show that

$$\lim_{j \to \infty} \|\nabla M_h(x_h^j)\| = 0.$$

That is, the algorithm guarantees convergence to a stationary point of the merit function. If, for example, a typical line search algorithm is used as the underlying optimization algorithm in MG/Opt, then it would be straightforward to modify the proof of convergence for that algorithm to incorporate the possibility of the recursion step in MG/Opt.

For that reason I will focus on whether the search direction $e_h$ from the recursion step of MG/Opt is guaranteed to be a descent direction for the merit function $M_h$ at $\bar{x}_h$. I make the following assumptions.

- **Assumption B2**: All of the iterates on level $h$ lie in a compact set $S_h$.
- **Assumption B3**: $f_h$ is twice continuously differentiable on $S_h$ on all levels $h$.
- **Assumption B4**: $a_h$ is continuously differentiable on $S_h$ on all levels $h$.
- **Assumption B5**: The smallest singular value of $\nabla a_h$ is uniformly bounded away from zero on $S_h$ on all levels $h$.
- **Assumption B6**: At the end of the Pre-smoothing step in MG/Opt, the multipliers $\bar{\lambda}_h$ satisfy $\bar{\lambda}_h = \mu(\bar{x}_h)$, where $\mu(x_h)$ is the least-squares multiplier estimate at $x_h$ (see below).
- **Assumption B7**: The update and downdate operators satisfy
$$(I_H^h)^T = C_I I_h^H \quad \text{and} \quad (J_H^h)^T = C_J J_h^H$$
for constants $C_I, C_J > 0$.

Assumptions B2, B3, B4, and B7 are routine. Assumption B5 is a constraint qualification used to guarantee that the Lagrange multiplier estimates are bounded. Assumption B6 is easy to guarantee by computing $\bar{\lambda}_h = \mu(\bar{x}_h)$ if this is not done already by the optimization algorithm used in the pre-smoothing step.

If the surrogate model is minimized exactly then

$$\nabla_x L_H(x_H^+, \lambda_H^+) = \bar{v}_H = \nabla_x L_H(\bar{x}_H, \bar{\lambda}_H) - I_h^H \nabla_x L_h(\bar{x}_h, \bar{\lambda}_h).$$

Hence

$$I_h^H \nabla_x L_h(\bar{x}_h, \bar{\lambda}_h) = [\nabla_x L_H(\bar{x}_H, \bar{\lambda}_H) - \nabla_x L_H(x_H^+, \bar{\lambda}_H)] + \nabla a_H(x_H^+)(\bar{\lambda}_H - \lambda_H^+).$$

If the surrogate model is only minimized approximately then

$$\nabla_x L_H(x_H^+, \lambda_H^+) = \bar{v}_H + z_1$$

for some $z_1$. This condition can be written as

$$\nabla_x L_s(x_H^+, \lambda_H^+) = z_1 \tag{5}$$

where $L_s$ is the Lagrangian for the surrogate model and constraints.

If the surrogate constraints are exactly satisfied, then

$$a_H(x_H^+) = \bar{s} = a_H(\bar{x}_H) - J_h^H a_h(\bar{x}_h)$$

so that

$$J_h^H a_h(\bar{x}_h) = a_H(\bar{x}_H) - a_H(x_H^+).$$

If the constraints are not exactly satisfied then

$$J_h^H a_h(\bar{x}_h) = a_H(\bar{x}_H) - a_H(x_H^+) + z_2$$

for some vector $z_2$. This condition can be written as

$$a_s(x_H^+) = z_2. \tag{6}$$

I will consider an augmented-Lagrangian merit function

$$M_h(x_h) \equiv f_h(x_h) + a_h(x_h)^T \mu(x_h) + \frac{\rho}{2} a_h(x_h)^T a_h(x_h),$$

where $\mu(x_h)$ is the least-squares estimate of the Lagrange multipliers at $x_h$:

$$\mu(x_h) \equiv -[\nabla a_h(x_h) \nabla a_h(x_h)^T]^{-1} \nabla a_h(x_h) \nabla f_h(x_h).$$

I obtain the following theorem.

**Theorem 6** *Assume that B1–B7 are satisfied. The search direction $e_h$ from the recursion step of MG/Opt will be a descent direction with respect to the augmented Lagrangian function $M_h$ if*

*(a) the penalty parameter $\rho$ is sufficiently large,*

*(b) $[\nabla a_h(\bar{x}_h) I_H^h - J_H^h \nabla a_H(\bar{x}_H)]^T a_h(\bar{x}_h)$ is sufficiently small,*

*(c) $\|\nabla a_H(\bar{x}_H) - \nabla a_H(\bar{x}_H + \alpha e_H)\|$ is sufficiently small for $0 \le \alpha \le 1$,*

*(d) $\|\nabla_x L_s(x_H^+, \lambda_H^+)\|$ is sufficiently small,*

*(e) $e_H^T P \nabla_{xx}^2 L_H(\xi, \bar{\lambda}_H) P e_H > 0$ for $\bar{x}_H \le \xi \le x_H^+$ where $P$ is a projection onto the null-space for the Jacobian of the constraints at $\bar{x}_H$, and*

*(f) $\|a_s(x_H^+)\|$ is sufficiently small.*

**Proof.** First note that $\bar{\lambda}_h = \mu(\bar{x}_h)$ because of Assumption B6. I test for descent by analyzing

$$\begin{aligned} e_h^T \nabla M_h(\bar{x}_h) &= [I_H^h e_H]^T \nabla M_h(\bar{x}_h) \\ &= e_H^T [C_I I_h^H \nabla_x L_h(\bar{x}_h, \bar{\lambda}_h) + \rho (I_H^h)^T \nabla a_h(\bar{x}_h) a_h(\bar{x}_h)] \\ &\qquad + e_h^T \nabla \mu(\bar{x}_h) a_h(\bar{x}_h) \\ &\equiv C_I T_1 + \rho T_2 + T_3, \end{aligned}$$

where

$$\begin{aligned} T_1 &= e_H^T I_h^H \nabla_x L_h(\bar{x}_h, \bar{\lambda}_h) \\ T_2 &= e_H^T (I_H^h)^T \nabla a_h(\bar{x}_h) a_h(\bar{x}_h) \\ T_3 &= e_h^T \nabla \mu(\bar{x}_h) a_h(\bar{x}_h). \end{aligned}$$

I now analyze the terms $T_1$, $T_2$, and $T_3$. First for $T_1$:

$$\begin{aligned} T_1 &= e_H^T I_h^H \nabla_x L_h(\bar{x}_h, \bar{\lambda}_h) \\ &= e_H^T [\nabla_x L_H(\bar{x}_H, \bar{\lambda}_H) - \nabla_x L_H(x_H^+, \bar{\lambda}_H) + z_1] + e_H^T \nabla a_H(x_H^+)(\bar{\lambda}_H - \lambda_H^+) \\ &= -e_H^T \nabla_{xx}^2 L_H(\xi, \bar{\lambda}_H) e_H + e_H^T z_1 + e_H^T \nabla a_H(x_H^+)(\bar{\lambda}_H - \lambda_H^+) \\ &\equiv T_{1a} + T_{1b} + T_{1c}. \end{aligned}$$

The vector $z_1$ comes from (5). In the analysis, I have used the mean-value theorem. The point $\xi$ is on the line segment connecting $\bar{x}_H$ and $x_H^+$.

I will discuss the first term $T_{1a}$ in connection with term $T_{2a}$ below. The second term $T_{1b}$ will be small if $\|\nabla_x L_s(x_H^+, \lambda_H^+)\|$ is small, i.e., if the coarse-level optimization problem is solved accurately enough. The third term $T_{1c}$ will be bounded because of Assumptions B2, B4, and B5, and assumption (c) of the theorem.

Now I analyze $T_2$:

$$\begin{aligned}
T_2 &= e_H^T (I_H^h)^T \nabla a_h(\bar{x}_h) a_h(\bar{x}_h) \\
&= e_H^T [\nabla a_h(\bar{x}_h)^T I_H^h]^T a_h(\bar{x}_h) \\
&= e_H^T [J_H^h \nabla a_H(\bar{x}_H)^T]^T a_h(\bar{x}_h) \\
&\qquad + e_H^T [\nabla a_h(\bar{x}_h)^T I_H^h - J_H^h \nabla a_H(\bar{x}_H)^T]^T a_h(\bar{x}_h) \\
&= C_J e_H^T \nabla a_H(\bar{x}_H) J_h^H a_h(\bar{x}_h) \\
&\qquad + e_H^T [\nabla a_h(\bar{x}_h)^T I_H^h - J_H^h \nabla a_H(\bar{x}_H)^T]^T a_h(\bar{x}_h) \\
&= C_J e_H^T \nabla a_H(\bar{x}_H)[a_H(\bar{x}_H) - a_H(x_H^+)] + C_J e_H^T \nabla a_H(\bar{x}_H) z_2 \\
&\qquad + e_H^T [\nabla a_h(\bar{x}_h)^T I_H^h - J_H^h \nabla a_H(\bar{x}_H)^T]^T a_h(\bar{x}_h) \\
&= -C_J e_H^T \nabla a_H(\bar{x}_H) \nabla a_H(\eta)^T e_H + C_J e_H^T \nabla a_H(\bar{x}_H) z_2 \\
&\qquad + e_H^T [\nabla a_h(\bar{x}_h)^T I_H^h - J_H^h \nabla a_H(\bar{x}_H)^T]^T a_h(\bar{x}_h) \\
&= -C_J \|\nabla a_H(\bar{x}_H)^T e_H\|_2^2 \\
&\qquad + C_J e_H^T \nabla a_H(\bar{x}_H)[\nabla a_H(\bar{x}_H) - \nabla a_H(\eta)]^T e_H \\
&\qquad + C_J e_H^T \nabla a_H(\bar{x}_H) z_2 \\
&\qquad + e_H^T [\nabla a_h(\bar{x}_h)^T I_H^h - J_H^h \nabla a_H(\bar{x}_H)^T]^T a_h(\bar{x}_h) \\
&\equiv T_{2a} + T_{2b} + T_{2c} + T_{2d}.
\end{aligned}$$

The vector $z_2$ comes from (6). I have again used the mean-value theorem. The point $\eta$ is on the line segment connecting $\bar{x}_H$ and $x_H^+$.

We can examine the terms $T_{1a}$ and $T_{2a}$ together:

$$C_I T_{1a} + \rho T_{2a} = -C_I e_H^T \nabla_{xx}^2 L_H(\xi, \bar{\lambda}_H) e_H - \rho C_J \|\nabla a_H(\bar{x}_H)^T e_H\|_2^2 \equiv -e_H^T W e_H$$

where

$$W = C_I \nabla_{xx}^2 L_H(\xi, \bar{\lambda}_H) + \rho C_J \nabla a_H(\bar{x}_H) \nabla a_H(\bar{x}_H)^T.$$

The matrix $W$ is similar in structure to the Hessian of an augmented-Lagrangian function, and hence is positive definite for $\rho$ sufficiently large if Assumption B5 and assumption (e) above are satisfied (see, e.g., [6]). Hence $C_I T_{1a} + \rho T_{2a}$ is negative for $\rho$ sufficiently large.

The second term $T_{2b}$ will be small if $\nabla a_H(\bar{x}_H) \approx \nabla a_H(\eta)$; if the constraints are linear this term will be zero. The third term $T_{2c}$ will be (nearly) zero if the coarse-level constraints are (nearly) satisfied. The fourth term $T_{2d}$ will be small if $\nabla a_h(\bar{x}_h)^T I_H^h \approx J_H^h \nabla a_H(\bar{x}_H)^T$ (this is a measure of how well the coarse-level constraints approximate the fine-level constraints), or if $a_h(\bar{x}_h)$ is small.

The term $T_3$ will be bounded because of the assumptions made at the beginning of this section [2]. More can be said about this term. If $a_h(\bar{x}_h) = 0$ then $T_3 = 0$; otherwise this term is dominated by $-e_H^T W e_H$ if $\rho$ is sufficiently large.

The theorem follows from these statements. □

Let me comment on reasonableness of the additional assumptions in the theorem. Assumption (a) can be dealt with through an appropriate implementation of the algorithm, and is a common assumption in the context of constrained optimization. Assumption (b) states that either the constraints are nearly satisfied, or that the coarse and fine level constraints are good approximations to each other in the sense that $\nabla a_h(\bar{x}_h) I_H^h \approx J_H^h \nabla a_H(\bar{x}_H)$. Assumption (c) limits the nonlinearity of the constraints, and would restrict how large $\alpha$ could be. Assumption (d) states that the coarse level model is solved to sufficient accuracy. Assumption (e) is analogous to assumption (b) in Theorem 4; see the discussion in Section 2. Assumption (f) states that the constraints are nearly satisfied.

If constraints are linear, then the Jacobian of the constraints will be constant on every level, and the assumption (c) in the theorem is unnecessary. Also, many classes of algorithms are able to ensure that linear constraints are satisfied at every iteration, and in that case assumptions (b) and (f) would also be unnecessary.

In the case of linear constraints, it is common to insist that the constraints remain satisfied at every iteration. As a consequence

$$A_H e_H = 0 \quad \text{and} \quad A_h e_h = 0.$$

Standard optimization techniques can be used to guarantee that $A_H e_H = 0$. If in addition $A_h I_H^h = J_H^h A_H$ then

$$A_h e_h = A_h I_H^h e_H = J_H^h A_H e_H = 0$$

as well. Further, if the constraints are always satisfied, then the merit function simplifies to

$$M_h(x_h, \lambda_h) = f_h(x_h)$$

and proving descent is analogous to the unconstrained case.

If the number of constraints is the same on all levels, i.e., $J_h^H = J_H^h = I$, then there is a slight simplification in the result. The second assumption becomes: (b) $[\nabla a_h(\bar{x}_h) I_H^h - \nabla a_H(\bar{x}_H)]^T a_h(\bar{x}_h)$ is sufficiently small.

## 3.1 The $\ell_1$ Merit Function

Another commonly used merit function is the $\ell_1$ merit function:

$$M_h(x_h) = f_h(x_h) + \rho \|a_h(x_h)\|_1.$$

In the context of sequential quadratic programming methods, it is possible to prove descent with respect to the $\ell_1$ merit function [3], and to obtain convergence results analogous to those for the augmented-Lagrangian merit function.

However, the search direction from MG/Opt is not guaranteed to be a descent direction for the $\ell_1$ merit function, as the following example demonstrates.

The example has a quadratic objective and linear constraints:

$$\min_{u,f} \frac{1}{2} \left[ \int_0^1 (u - u_*)^2 \, dx + \int_0^1 (f - f_*)^2 \, dx \right]$$

subject to

$$-u''(x) = f(x) + b_*(x), \quad 0 < x < 1$$

with $u(0) = u(1) = 0$. The functions $u_*(x)$, $f_*(x)$, and $b_*(x)$ are specified below.

To obtain the finite-dimensional models, I use a uniform discretization on the interval $[0, 1]$. I choose evenly spaced points $x_0 = 0 < x_1 < \cdots < x_n < x_{n+1} = 1$, where $x_i - x_{i-1} = h$. Then $u_i \approx u(x_i)$ and $f_i \approx f(x_i)$ for $1 \leq i \leq n$. If I set $u_0 = u_{n+1} = 0$ then for $1 \leq i \leq n$:

$$\frac{u_{i-1} + 2u_i - u_{i-1}}{h^2} = f_i + b_i.$$

I use the trapezoid rule to approximate the integrals in the objective function, since it has the same order of accuracy as the solution to the differential equation constraint.

The fine-level model uses the discretization $h = 1/16$, and the coarse-level model uses $H = h/2$. The functions $u_*$, $f_*$, and $b_*$ are

$$\begin{aligned} u_*(x) &= 1 + x^2 \\ f_*(x) &= \cos(x) \\ b_*(x) &= 20x(x-1)(x-0.1)(x-0.7) \end{aligned}$$

The penalty parameter in the merit function is $\rho = 100$.

The goal of these tests is to study the descent properties of the search direction from the Recursion step of MG/Opt. For that reason, the tests specify the value of $\bar{x}_h$, solve the coarse-level subproblem exactly, compute the search direction $e_h$, and then plot the values of $M_h(\bar{x}_h + \alpha e_h)$ for $0 \leq \alpha \leq 1$.

I choose $\bar{x}_h = x_* + 0.01w$ where $x_*$ is the solution to the fine-level problem and $w$ is a random vector obtained using the Matlab commands:

$$\texttt{randn('state',4)} \tag{7}$$

$$\texttt{w = randn(nh,1)} \tag{8}$$

Here `nh` is the number of variables on the fine level.

Figure 1 shows the results for the $\ell_1$ merit function where the search direction from MG/Opt is an *ascent* direction. Figures 2 and 3 plot the values of the objective function and the penalty term, respectively. Although the objective function is decreasing, the penalty term is increasing, so it is not possible to get descent for the $\ell_1$ merit function by increasing the penalty parameter.

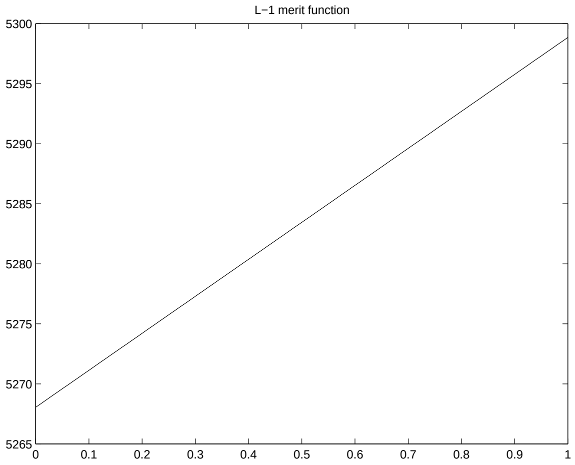

Figure 1: $\ell_1$ Merit Function (near solution)

# 4 Inequality Constraints

I now consider an optimization problem with inequality constraints:

$$\begin{array}{ll} \min\limits_{y_h} & g_h(y_h) \\ \text{subject to} & c_h(y_h) \leq 0 \end{array} \tag{9}$$

where the subscript $h$ refers to the level of the model. This problem uses different notation than before, because I will transform it to an equality-constrained problem, and the transformed problem will use the notation used earlier. As before, also provided are a downdate operator $I_h^H$ and an update operator $I_H^h$ for the variables, as well as a downdate operator $J_h^H$ and an update operator $J_H^h$ for the constraints. To specify the MG/Opt algorithm I make the following assumption:

- **Assumption C1**: $\nabla g_h(y_h)$ and $\nabla c_h(y_h)$ are defined for all choices of $y_h$.

I define the Lagrangian function as

$$\hat{L}_h(y_h, \lambda_h) = g_h(y_h) + c_h(y_h)^T \lambda_h.$$

As in the equality-constrained case, the algorithm is defined in terms of a convergent optimization algorithm that can be applied to (9), and that algorithm is based on a merit function $\hat{M}_h(y_h)$. Here is the MG/Opt algorithm for an inequality-constrained problem.

- Given an initial estimate of the solution $(y_h^0, \lambda_h^0)$, and integers $k_1, k_2 \geq 0$ satisfying $k_1 + k_2 > 0$, for $j = 0, 1, \ldots$ until converged:

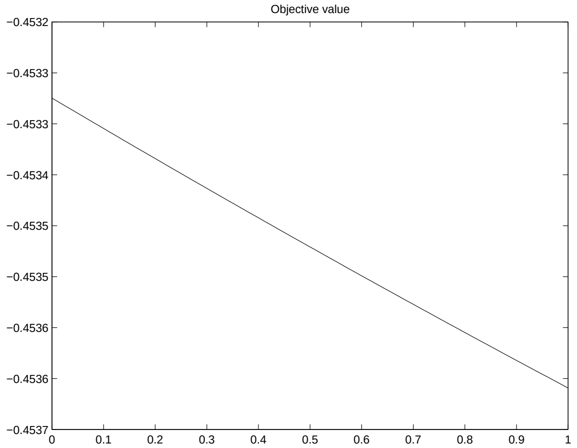

Figure 2: Objective Function

- *Pre-smoothing:* Apply $k_1$ iterations of a convergent optimization algorithm to (9) to obtain $(\bar{y}_h, \bar{\lambda}_h)$ (with $(y_h^j, \lambda_h^j)$ used as the initial guess), where the convergent optimization algorithm is based on a merit function $\hat{M}_h(y_h)$.
- *Recursion:*
  - ∗ Compute $\bar{y}_H = I_h^H \bar{y}_h$, $\bar{\lambda}_H = J_h^H \bar{\lambda}_h$, $\hat{v}_H = \nabla_y \hat{L}_H(\bar{y}_H, \bar{\lambda}_H) - I_h^H \nabla_y \hat{L}_h(\bar{y}_h, \bar{\lambda}_h)$, and $\hat{s} = c_H(\bar{y}_H) - J_h^H c_h(\bar{y}_h)$.
  - ∗ Minimize (perhaps approximately) the "surrogate" model

    $$g_s(y_H) \equiv g_H(y_H) - \hat{v}_H^T y_H$$

    subject to the "surrogate" constraints

    $$c_s(y_H) \equiv c_H(y_H) - \hat{s} \leq 0$$

    to obtain $(y_H^+, \lambda_H^+)$ (with $(\bar{y}_H, \bar{\lambda}_H)$ used as the initial guess). The minimization could be performed recursively by calling MG/Opt.
  - ∗ Compute the search directions $e_H = y_H^+ - \bar{y}_H$ and $e_h = I_H^h e_H$.
  - ∗ Use a line search to determine $y_h^+ = \bar{y}_h + \alpha e_h$ satisfying $\hat{M}_h(y_h^+) \leq \hat{M}_h(\bar{y}_h)$. Compute Lagrange multiplier estimates $\lambda_h^+$.
- *Post-smoothing:* Apply $k_2$ iterations of the same convergent optimization algorithm to (9) to obtain $(y_h^{j+1}, \lambda_h^{j+1})$ (with $(y_h^+, \lambda_h^+)$ used as the initial guess).

Corresponding to the surrogate model and constraints in the recursion step, I define the surrogate Lagrangian as

$$\hat{L}_s(y_H, \lambda_H) \equiv g_s(y_H, \lambda_H) + c_s(y_H)^T \lambda_H.$$

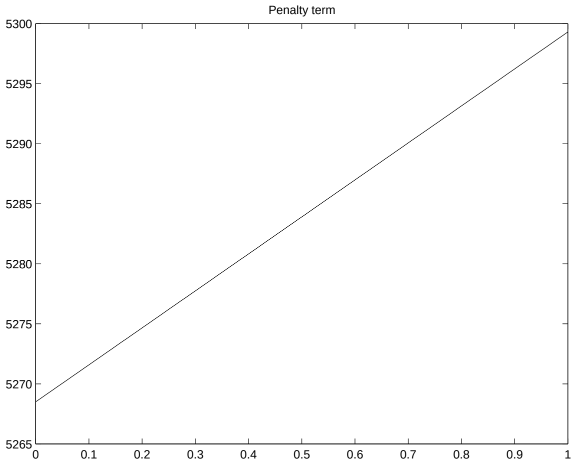

Figure 3: $\ell_1$ Penalty Term

The surrogate optimization model is chosen so that it is a first-order approximation to the downdated fine-level model in the sense that

$$\nabla \hat{L}_s(\bar{y}_H, \bar{\lambda}_H) = \begin{pmatrix} I_h^H & 0 \\ 0 & J_h^H \end{pmatrix} \nabla \hat{L}_h(\bar{y}_h, \bar{\lambda}_h).$$

Here the gradient is with respect to both the variables $y_H$ and the multipliers $\lambda_H$.

It will be useful in the later discussion to derive the above algorithm in another way. In the case where $J_h^H = J_H^h = I$, the algorithm above can be obtained by considering the following equality-constrained problem

$$\begin{array}{ll} \min\limits_{x_h} & f_h(x_h) \\ \text{subject to} & a_h(x_h) = 0 \end{array}$$

where

$$\begin{aligned} x_h &= \begin{pmatrix} y_h \\ z_h \end{pmatrix} \\ f_h(x_h) &= f_h(y_h, z_h) = g_h(y_h) \\ a_h(x_h)_i &= c_h(x_h)_i + (z_h)_i^2, \end{aligned}$$

i.e., I have used squared slack variables to convert the inequalities to equations, an approach that is also used in [1]. The results for equality-constrained problems can be applied to the transformed problem. In the following, I use $Z_h$ to represent the diagonal matrix with diagonal entries equal to $z_h$, and similarly for $\bar{Z}_h$, etc. With this notation the constraints for the transformed problem can be written as

$$a_h(x_h) = c_h(x_h) + Z_h z_h = 0.$$

To derive the surrogate model, look at the Lagrangian for the transformed problem:

$$L_h(x_h, \lambda_h) = f_h(x_h) + a_h(x_h)^T \lambda_h = g_h(y_h) + [c_h(y_h) + Z_h z_h]^T \lambda_h.$$

Then

$$\nabla_y L_h(\bar{x}_h, \bar{\lambda}_h) = \nabla g_h(\bar{y}_h) + \nabla c_h(\bar{y}_h)^T \bar{\lambda}_h$$

and

$$\nabla_z L_h(\bar{x}_h, \bar{\lambda}_h) = 2\bar{Z}_h \bar{\lambda}_h.$$

If the complementary slackness conditions are satisfied, then

$$\nabla_z L_h(\bar{x}_h, \bar{\lambda}_h) = 0.$$

Similarly,

$$\nabla_y L_H(\bar{x}_H, \bar{\lambda}_H) = \nabla g_H(\bar{y}_H) + \nabla c_H(\bar{y}_H) \bar{\lambda}_H$$

and

$$\nabla_z L_H(\bar{x}_H, \bar{\lambda}_H) = 2\bar{Z}_H \bar{\lambda}_H = 2\bar{Z}_h \bar{\lambda}_h = 0,$$

since in this special case $J_H^h = J_h^H = I$. Thus in the notation of the equality-constrained version of MG/Opt:

$$\begin{aligned}
\bar{v}_H &= \nabla_x L_h(\bar{x}_H, \bar{\lambda}_H) - I_h^H \nabla_x L_h(\bar{x}_h, \bar{\lambda}_h) \\
&= \begin{pmatrix} \nabla_y L_h(\bar{x}_H, \bar{\lambda}_H) - I_h^H \nabla_y L_h(\bar{x}_h, \bar{\lambda}_h) \\ \nabla_z L_h(\bar{x}_H, \bar{\lambda}_H) - I_h^H \nabla_z L_h(\bar{x}_h, \bar{\lambda}_h) \end{pmatrix} \\
&= \begin{pmatrix} \nabla_y \hat{L}_h(\bar{y}_H, \bar{\lambda}_H) - I_h^H \nabla_y \hat{L}_h(\bar{y}_h, \bar{\lambda}_h) \\ 0 \end{pmatrix} = \begin{pmatrix} \hat{v}_H \\ 0 \end{pmatrix}.
\end{aligned}$$

Hence the objective function for the coarse-level problem is $f_H(x_h) - \bar{v}_H^T x_H = g_H(y_H) - \hat{v}_H^T y_H$, as stated above.

The coarse-level constraints for the transformed problem are

$$\begin{aligned}
0 = a_s(x_H) &= a_H(x_H) - \bar{s} \\
&= a_H(x_H) - [a_H(\bar{x}_H) - a_h(\bar{x}_h)] \\
&= c_H(y_H) + Z_H z_H - [(c_H(\bar{y}_H) + \bar{Z}_H \bar{z}_H) - (c_h(\bar{y}_h) + \bar{Z}_h \bar{z}_h)] \\
&= c_H(y_H) + Z_H z_H - [(c_H(\bar{y}_H) + \bar{Z}_H \bar{z}_H) - (c_h(\bar{y}_h) + \bar{Z}_H \bar{z}_H)] \\
&= c_H(y_H) - [(c_H(\bar{y}_H) - (c_h(\bar{y}_h)] + Z_H z_H \\
&= c_H(y_H) - \hat{s} + Z_H z_H.
\end{aligned}$$

Hence we obtain

$$c_H(y_H) - \hat{s} \leq 0,$$

which are the constraints stated in the MG/Opt algorithm above. Note that, because $z_h = z_H$, we have that $\bar{s} = \hat{s}$.

I will use this equality-constrained formulation again below. But let me emphasize that the squared slack variables $z_h$ are only used for the purpose of deriving the MG/Opt algorithm, and for analyzing its behavior. It is not assumed that the optimization algorithms use squared slack variables.

Convergence theorems for MG/Opt can be obtained as in the equality constrained case. Hence my main focus is to determine under what conditions the search direction $e_h$ from the recursion step of MG/Opt is guaranteed to be a descent direction for the merit function $\hat{M}_h$ at $\bar{y}_h$. I will make the following assumptions (these are similar to the assumptions made in the equality constrained case):

- **Assumption C2**: All of the iterates on level $h$ lie in a compact set $S_h$.
- **Assumption C3**: $g_h$ is twice continuously differentiable on $S_h$ on all levels $h$.
- **Assumption C4**: $c_h$ is continuously differentiable on $S_h$ on all levels $h$.
- **Assumption C5**: The smallest singular value of $\nabla \bar{c}_h$ is uniformly bounded away from zero on $S_h$ on all levels $h$, where $\bar{c}_h$ is the set of active and violated constraints.
- **Assumption C6**: At the end of the Pre-smoothing step in MG/Opt, the multipliers $\bar{\lambda}_h$ satisfy $\bar{\lambda}_h = \hat{\mu}(\bar{y}_h)$, where $\hat{\mu}(y_h)$ is the least-squares multiplier estimate at $y_h$ (see below).
- **Assumption C7**: The update and downdate operators satisfy

$$(I_H^h)^T = C_I I_h^H \quad \text{and} \quad (J_H^h)^T = C_J J_h^H$$

  for constants $C_I, C_J > 0$.

The multipliers are computed using the least-squares formula from the last section, based on the current set of active constraints. Multipliers for inactive constraints are zero.

In the following I will refer both to the original optimization problem (9) as well as the corresponding equality-constrained problem involving squared slack variables. I will also define the set of constraint violations

$$\hat{c}_h(y_h) = \max\{c_h(y_h), 0\}.$$

We can write the constraints in three different ways:

$$\begin{aligned} c_h(y_h) &\leq 0 \\ a_h(x_h) = c_h(y_h) + Z_h z_h &= 0 \\ \hat{c}_h(y_h) &= 0 \end{aligned}$$

There will be analogous definitions for the coarse-level model. The discussion below does not assume that $J_h^H = J_H^h = I$.

If the surrogate model is minimized exactly then $\nabla_y \hat{L}_s(y_H^+, \lambda_H^+) = 0$ where $L_s$ is the Lagrangian for the surrogate model, but in general

$$\nabla_y \hat{L}_s(y_H^+, \lambda_H^+) = z_1 \tag{10}$$

for some $z_1$. Similarly if the surrogate constraints are exactly satisfied then $a_s(x_H^+) = 0$ but more generally

$$a_s(x_H^+) = z_2 \tag{11}$$

for some $z_2$.

I will consider an augmented-Lagrangian merit function

$$\hat{M}_h(y_h) = g_h(y_h) + \hat{c}_h(y_h)^T \hat{\mu}(y_h) + \frac{\rho}{2} \hat{c}_h(x_h)^T \hat{c}_h(x_h),$$

where $\hat{\mu}(y_h)$ is the least-squares estimate of the Lagrange multipliers at $y_h$. In terms of the transformed model we have that

$$\hat{c}_h(y_h) = c_h(y_h) + Z_h z_h = a_h(x_h),$$

assuming that the slack variables are defined appropriately. Thus, if we define $\mu(x_h) = \hat{\mu}(y_h)$ then we can define the merit function for the transformed problem as

$$M_h(x_h) \equiv f_h(x_h) + a_h(x_h)^T \mu(x_h) + \frac{\rho}{2} a_h(x_h)^T a_h(x_h).$$

Before analyzing the descent properties of MG/Opt, I derive formulas for the gradient of the merit function at $\bar{x}_h = (\bar{y}_h, \bar{z}_h)$:

$$\begin{aligned} \nabla_z M_h(\bar{y}_h, \bar{z}_h) &= 2\bar{Z}_h \mu(\bar{y}_h, \bar{z}_h) + 2\rho \bar{Z}_h [c_h(\bar{y}_h) + \bar{Z}_h \bar{z}_h] \\ &= 2\bar{Z}_h \bar{\lambda}_h + 2\rho \bar{Z}_h [c_h(\bar{y}_h) + \bar{Z}_h \bar{z}_h]. \end{aligned}$$

As discussed earlier, $\bar{Z}_h \bar{\lambda}_h = 0$ because of complementary slackness. The other term is also zero because if $(\bar{z}_h)_i \neq 0$ then $c_h(\bar{y}_h)_i + (\bar{z}_h)_i^2 = 0$. Hence

$$\nabla_z M_h(\bar{y}_h, \bar{z}_h) = 0.$$

In addition

$$\begin{aligned} \nabla_y M_h(\bar{y}_h, \bar{z}_h) &= \nabla g_h(\bar{y}_h) + \rho \nabla c_h(\bar{y}_h)[c_h(\bar{y}_h) + \bar{Z}_h \bar{z}_h] \\ &\qquad + \nabla c_h(\bar{y}_h) \bar{\lambda}_h + \nabla \mu(\bar{y}_h, \bar{z}_h)[c_h(\bar{y}_h) + \bar{Z}_h \bar{z}_h] \\ &= \nabla_y f_h(\bar{y}_h, \bar{z}_h) + \nabla_y a_h(\bar{y}_h, \bar{z}_h) \bar{\lambda}_h \\ &\qquad + \rho \nabla_y a_h(\bar{y}_h, \bar{z}_h) a_h(\bar{y}_h, \bar{z}_h) + \nabla \mu(\bar{y}_h, \bar{z}_h) a_h(\bar{y}_h, \bar{z}_h) \\ &= \nabla_y L_h(\bar{x}_h, \bar{\lambda}_h) + \rho \nabla_y a_h(\bar{x}_h) a_h(\bar{x}_h) + \nabla \mu(\bar{x}_h) a_h(\bar{x}_h). \end{aligned}$$

If we test for descent for the transformed problem then the search direction on the fine level is

$$p_h \equiv \begin{pmatrix} e_h \\ z_h^+ - \bar{z}_h \end{pmatrix}.$$

However, since $\nabla_z M_h(\bar{y}_h, \bar{z}_h) = 0$, we have that

$$p_h^T \nabla M_h(\bar{y}_h, \bar{z}_h) = e_h^T \nabla_y M_h(\bar{y}_h, \bar{z}_h) = e_h^T \nabla \hat{M}_h(\bar{y}_h).$$

As a result, we can take advantage of the analysis from the equality-constrained case.

The theorem for equality constraints applies immediately, but its assumptions involve derivatives with respect to all of the variables for the transformed problem. However, as the above analysis indicates, the only non-zero terms are associated with the derivatives with respect to the variables $y_h$ and $y_H$, and not the derivatives with respect to the slack variables $z_h$ and $z_H$. Thus we obtain the theorem below.

**Theorem 7** *Assume that C1–C7 are satisfied. The search direction $e_h$ from the recursion step of MG/Opt will be a descent direction with respect to the augmented Lagrangian function $\hat{M}_h$ if*

*(a) the penalty parameter $\rho$ is sufficiently large;*

*(b) $[\nabla c_h(\bar{y}_h) I_H^h - \nabla c_H(\bar{y}_H)]^T \hat{c}_h(\bar{y}_h)$ is sufficiently small,*

*(c) $\|\nabla c_H(\bar{y}_H) - \nabla c_H(\bar{y}_H + \alpha e_H)\|$ is sufficiently small for $0 \le \alpha \le 1$,*

*(d) $\|\nabla_y \hat{L}_s(y_H^+, \lambda_H^+)\|$ is sufficiently small,*

*(e) $e_H^T P \nabla_{xx}^2 L_H(\xi, \bar{\lambda}_H) P e_H > 0$ for $\bar{y}_H \le \xi \le y_H^+$ where $P$ is a projection onto the null space for the Jacobian of the active constraints at $\bar{y}_H$. and*

*(e) $\|\hat{c}_s(y_H^+)\|$ is sufficiently small where $\hat{c}_s$ corresponds to the constraint violations in the surrogate model.*

If the constraints are linear, then the Jacobian of the constraints will be constant on every level, and the assumption (c) is unnecessary. Also, many classes of algorithms are able to ensure that linear constraints are satisfied at every iteration, and in that case assumptions (b) and (e) would also be unnecessary.

If the constraints are always satisfied, then the merit function simplifies to

$$M_h(x_h, \lambda_h) = f_h(x_h)$$

and proving descent is analogous to the unconstrained case. This will be true for some algorithms in the case of linear constraints. It will also be true if interior-point methods are used and the iterates are feasible.

# 5 Summary of MG/Opt Algorithm

The earlier description isolates the essentials of the algorithm, in a form suitable for analyzing convergence properties. The following description is more useful for purposes of implementation. It applies to the general optimization problem (2), and assumes the availability of appropriate update and downdate operators: $I_h^H$ and $I_h^H$ for the variables $x_h$, $J_h^H$ and $J_H^h$ for the equality constraints $a_h$, and $K_h^H$ and $K_H^h$ for the inequality constraints $c_h$. I will use $\lambda_h$ to refer to the multipliers for the equality constraints and $\mu_h$ to refer to the multipliers for the inequality constraints. The Lagrangian for (2) is

$$L_h(x_h, \lambda_h, \mu_h) = f_h(x_h) + a_h(x_h)^T \lambda_h + c_h(x_h)^T \mu_h.$$

The algorithm also assumes the availability of a convergent optimization algorithm Opt defined as a function of the form

$$(x^+, \lambda^+, \mu^+) \leftarrow \mathrm{Opt}(f(\cdot), v, a(\cdot), s_a, c(\cdot), s_c, \bar{x}, \bar{\lambda}, \bar{\mu}, k)$$

which applies $k$ iterations of a convergent optimization algorithm to the problem

$$\begin{array}{ll} \min\limits_{x} & f(x) - v^T x \\ \text{subject to} & a_h(x_h) - s_a = 0 \\ & c_h(x_h) - s_c \leq 0 \end{array}$$

with initial guess $(\bar{x}, \bar{\lambda}, \bar{\mu})$ to obtain $(x^+, \lambda^+, \mu^+)$. If the parameter $k$ is omitted, the optimization algorithm continues to run until its termination criteria are satisfied. The algorithm Opt is assumed to be based on a merit function $M_h(x_h)$. The algorithm MG/Opt has non-negative integer parameters $k_1$ and $k_2$ satisfying $k_1 + k_2 > 0$.

It is straightforward to modify this algorithm to apply to an unconstrained problem or a problem with only equality constraints. In those cases the optimization algorithm Opt and its calling sequence would be simplified. In the unconstrained case the merit function would just be the objective function.

There is considerable flexibility in how the algorithm is implemented. The convergence of MG/Opt only depends on the convergence of the underlying algorithm used for optimization on the finest level. Hence it would be possible to change the values of $k_1$ and $k_2$ from iteration to iteration. This might be appropriate if the initial guess were poor, and it was desirable to use a lower-cost method at points far from the solution. It would also be possible to adjust the characteristics of the underlying optimization method, as long as the convergence guarantees were maintained.

Here then is the algorithm: Given an initial estimate of the solution $(x_h^0, \lambda_h^0, \mu_h^0)$, set $v_h = 0$, $s_{a,h} = 0$, and $s_{h,c} = 0$. Then for $j = 0, 1, \ldots$, set

$$(x_h^{j+1}, \lambda_h^{j+1}, \mu_h^{j+1}) \leftarrow \mathrm{MG/Opt}(f_h(\cdot), v_h, a_h(\cdot), s_{a,h}, c_h(\cdot), s_{c,h}, x_h^j, \lambda_h^j, \mu_h^j)$$

where the function MG/Opt is defined as follows:

- *Coarse-level solve:* If on the coarsest level,

$$(x_h^{j+1}, \lambda_h^{j+1}, \mu_h^{j+1}) \leftarrow \mathrm{Opt}(f_h(\cdot), v_h, a_h(\cdot), s_{a,h}, c_h(\cdot), s_{c,h}, x_h^j, \lambda_h^j, \mu_h^j).$$

  Otherwise,

- *Pre-smoothing:*

$$(\bar{x}_h, \bar{\lambda}_h, \bar{\mu}_h) \leftarrow \mathrm{Opt}(f_h(\cdot), v_h, a_h(\cdot), s_{h,a}, c_h(\cdot), s_{c,h}, x_h^j, \lambda_h^j, \mu_h^j, k_1)$$

- *Recursion:*
  - Compute

$$\begin{aligned}
\bar{x}_H &= I_h^H \bar{x}_h \\
\bar{\lambda}_H &= J_h^H \bar{\lambda}_h \\
\bar{\mu}_H &= K_h^H \bar{\mu}_h \\
\bar{v}_H &= I_h^H v_h + \nabla L_H(\bar{x}_H, \bar{\lambda}_H, \bar{\mu}_H) - I_h^H \nabla L_h(\bar{x}_h, \bar{\lambda}_h, \bar{\mu}_h) \\
\bar{s}_{a,H} &= J_h^H s_{a,h} + a_H(\bar{x}_H) - J_h^H a_h(\bar{x}_h) \\
\bar{s}_{c,H} &= K_h^H s_{c,h} + c_H(\bar{x}_H) - K_h^H c_h(\bar{x}_h)
\end{aligned}$$

  - Apply MG/Opt recursively to the surrogate model:

$$(x_h^+, \lambda_h^+, \mu_h^+) \leftarrow \text{MG/Opt}(f_H(\cdot), \bar{v}_H, a_H(\cdot), \bar{s}_{a,H}, c_H(\cdot), \bar{s}_{c,H}, \bar{x}_H, \bar{\lambda}_H, \bar{\mu}_H)$$

  - Compute the search directions $e_H = x_H^+ - \bar{x}_H$ and $e_h = I_H^h e_H$.
  - Use a line search to determine $x_h^+ = \bar{x}_h + \alpha e_h$ satisfying $M_h(x_h^+) \leq M_h(\bar{x}_h)$.
  - Compute the new multipliers $\lambda_h^+$ and $\mu_h^+$.
- *Post-smoothing:*

$$(x_h^{j+1}, \lambda_h^{j+1}, \mu_h^{j+1}) \leftarrow \text{Opt}(f_h(\cdot), v_h, a_h(\cdot), s_{a,h}, c_h(\cdot), s_{c,h}, x_h^+, \lambda_h^+, \mu_h^+, k_2)$$

# 6 Acknowledgements

I would like to thank Paul Boggs, David Gay, and Michael Lewis for their many helpful comments, and in particular to thank Michael Lewis for suggesting the example used in Section 3.1.

# References

[1] P. T. Boggs, A. J. Kearsley, and J. W. Tolle, *A global convergence analysis of an algorithm for large-scale nonlinear optimization problems*, SIAM Journal on Optimization, 9 (1999), pp. 833–862.

[2] P. T. Boggs and J. W. Tolle, *Sequential quadratic programming*, Acta Numerica, 4 (1995), pp. 1–52.

[3] R. H. Byrd and J. Nocedal, *An analysis of reduced Hessian methods for constrained optimization*, Mathematical Programming, 49 (1991), pp. 285–323.

[4] S. Gratton, A. Sartenaer, , and P. L. Toint, *Recursive trust-region methods for multilevel nonlinear optimization*, SIAM Journal on Optimization, 19 (2008), pp. 414–444.

[5] L. Grippo, F. Lampariello, and S. Lucidi, *A nonmonotone line search technique for Newton's method*, SIAM Journal on Numerical Analysis, 23 (1986), pp. 707–716.

[6] I. Griva, S. G. Nash, and A. Sofer, *Linear and Nonlinear Optimization*, SIAM, Philadelphia, 2008.

[7] N. Kydes, *A Multigrid Solution of the Continuous Dynamic Disequilibrium Network Design Problem*, PhD thesis, School of Information Technology and Engineering, George Mason University, Fairfax, Virginia, 2002.

[8] R. M. Lewis and S. G. Nash, *Model problems for the multigrid optimization of systems governed by differential equations*, SIAM Journal on Scientific Computing, 26 (2005), pp. 1811–1837.

[9] S. F. McCormick, *Multilevel Projection Methods for Partial Differential Equations*, Society for Industrial and Applied Mathematics, 1992.

[10] A. Meenakshi and C. Rajian, *On a product of positive semidefinite matrices*, Linear Algebra and its Applications, 295 (1999), pp. 3–6.

[11] S. G. Nash, *A multigrid approach to discretized optimization problems*, Journal of Computational and Applied Mathematics, 14 (2000), pp. 99–116.

[12] J. Nocedal and S. Wright, *Numerical Optimization*, Springer Series in Operations Research, Springer, New York, 1999.

[13] M. P. Rumpfkeil and D. J. Mavriplis, *Optimization-based multigrid applied to aerodynamic shape design*, tech. report, Department of Mechanical Engineering, University of Wyoming, Laramie, 2009.

[14] M. Vallejos and A. Borzì, *Multigrid optimization methods for linear and bilinear elliptic optimal control problems*, Computing, 82 (2008), pp. 31–52.

[15] Z. Wen and D. Goldfarb, *A line search multigrid method for large-scale convex optimization*, tech. report, Department of IEOR, Columbia University, 2007.