An Inexact Modified Subgradient Algorithm for Nonconvex Optimization*

Regina S. Burachik† C. Yalçın Kaya‡ Musa Mammadov§

August 19, 2008

Abstract

We propose and analyze an inexact version of the modified subgradient (MSG) algorithm, which we call the IMSG algorithm, for nonsmooth and nonconvex optimization over a compact set. We prove that under an approximate, i.e. inexact, minimization of the sharp augmented Lagrangian, the main convergence properties of the MSG algorithm are preserved for the IMSG algorithm. Inexact minimization may allow to solve problems with less computational effort. We illustrate this through test problems, including an optimal bang–bang control problem, under several different inexactness schemes.

Key words: Nonconvex optimization, Nonsmooth optimization, Sharp augmented Lagrangian, Modified subgradient method, Inexact minimization, Bang–bang control.

Mathematical Subject Classification: 90C26; 49M29; 93B40; 93B35

1 Introduction

We consider the continuous optimization problem

$$(P) \qquad \text{Minimize } f_0(x) \text{ over all } x \text{ in } X \text{ satisfying } f(x) = 0,$$

where X is a compact subset of $\mathbb{R}^n$, and the functions $f_0 : \mathbb{R}^n \to \mathbb{R}$ and $f : \mathbb{R}^n \to \mathbb{R}^m$ are continuous.

*Authors are indebted to the reviewers and the associate editor for their constructive and helpful comments which greatly improved the contents and exposition of this paper.

†Centre for Informatics and Applied Optimization, University of Ballarat, Victoria, Australia; and School of Mathematics and Statistics, University of South Australia, Mawson Lakes, S.A. 5095 Australia. The author acknowledges support by the Australian Research Council Discovery Project Grant DP0556685 for this study. E-mail: regina.burachik@unisa.edu.au

‡School of Mathematics and Statistics, University of South Australia, Mawson Lakes, S.A. 5095 Australia. E-mail: yalcin.kaya@unisa.edu.au

§Centre for Informatics and Applied Optimization, University of Ballarat, Victoria, Australia. The author acknowledges support by the Australian Research Council Discovery Project Grant DP0556685 for this study. E-mail: m.mammadov@ballarat.edu.au

A fundamental tool for tackling problem (P) is Lagrangian duality. Under certain classes of augmented Lagrangian schemes, the dual problem is nonsmooth and convex, and zero duality gap holds [20, 6]. In order to solve the dual problem one can typically use nonsmooth convex techniques such as subgradient methods and their extensions. One such extension is the modified subgradient (MSG) algorithm [8, 9, 4, 5], which uses the sharp augmented Lagrangian [20].

In [4], it is shown that using the MSG algorithm, dual convergence is achieved, and under some additional conditions, convergence to a primal solution can be obtained. In the same reference numerical experiments are also presented to illustrate the use and efficiency of the algorithm.

In order to calculate the step size and the modified subgradient direction within the MSG algorithm, it is necessary to find the global minimum of the augmented Lagrangian at a dual iterate. This task can be very time-consuming (sometimes it may even be impossible) because of the inherent difficulties of global minimization problems. A natural question to pose here is whether the convergence properties of MSG could be retained while we carry out the global minimization of the augmented Lagrangian in an approximate, or *inexact*, way. The aim of the present paper is to study such an inexact version of the MSG algorithm, namely the IMSG algorithm. As a result, we prove that the main convergence properties of MSG are preserved for IMSG. From a practical viewpoint, we have in mind obtaining computational savings, both in CPU time and function evaluations, in solving Problem (P) with IMSG. We carry out numerical experiments and demonstrate these computational savings. However, one has to accept the fact that finding a global minimum, even approximately, can still be a highly demanding task.

In the IMSG algorithm, we consider a dynamic step-size in the spirit of the one introduced by Polyak [19], and further studied, e.g., by Brännlund [3] and Nedić and Bertsekas [17]. For a broad choice of the dynamic step-size, we prove dual and primal convergence (see Theorems 4.2 and 4.3). Moreover, we establish equivalence between existence of dual solutions and boundedness of the dual sequence (see Theorem 4.1).

For implementation of the IMSG algorithm, we devise a practical (i.e., computationally implementable) step-size, which makes use of an inexact value of the Lagrangian (see Proposition 5.1). This step-size, which is proposed for the case when the optimal dual value is unknown, makes use of upper bound estimates of the optimal dual value. With finite termination (a natural assumption for any numerical algorithm), it suffices to choose a fixed upper-bound estimate of the optimal dual value. This is what we do in our numerical experiments.

The paper is organized as follows. In Section 2 we recall the sharp augmented Lagrangian duality framework and give preliminary properties of our inexact scheme. In Section 3 we introduce the IMSG algorithm and establish some of its basic properties. In Section 4 we give the main existence and convergence results. In Section 5, we discuss a practical step-size selection for implementation of the IMSG algorithm. In Section 6, we present numerical experiments. We illustrate the working of the algorithm and verify the computational savings achieved by means of three test problems. We also demonstrate the global search process in minimizing the sharp augmented Lagrangian as part of the MSG and IMSG algorithms through a simple problem involving a nonsmooth system of equations.

2 Duality Framework and Preliminaries

Zero duality gap results have been obtained in [20] for a wide family of augmented Lagrangians, including the sharp augmented Lagrangian.

With the notation of Problem (P), denote by

$$X_0 := \{x \in X \ : \ f(x) = 0\} \tag{1}$$

the constraint set. Let

$$\underline{f_0} := \inf_{x \in X_0} f_0(x) = \min_{x \in X_0} f_0(x)$$

be the optimal value of Problem (P). The *sharp augmented Lagrangian* $L : X \times \mathbb{R}^m \times \mathbb{R}_+ \to \mathbb{R}$ (see [20]) is given by

$$L(x,(u,c)) := f_0(x) + c\|f(x)\| - \langle u, f(x)\rangle.$$

The *dual function* induced by the sharp augmented Lagrangian is $H : \mathbb{R}^m \times \mathbb{R}_+ \to \mathbb{R} \cup \{-\infty\}$ defined as

$$H(u,c) := \inf_{x \in X} L(x,(u,c)) \ .$$

So the *dual problem* for (P) is

$$(D) \qquad\qquad \text{Maximize} \quad H(u,c) \ \text{ over all } (u,c) \in \mathbb{R}^m \times \mathbb{R}_+,$$

and hence the *optimal dual value* is

$$\overline{H} := \sup_{(u,c) \in \mathbb{R}^m \times \mathbb{R}_+} H(u,c) \ .$$

We say that *weak duality* holds if

$$H(u,c) \leq \underline{f_0}$$

for all $(u,c) \in \mathbb{R}^m \times \mathbb{R}_+$. We say that *strong duality* (in other words, *zero duality gap*) holds if

$$\overline{H} = \underline{f_0} \ .$$

Our duality scheme enjoys zero duality gap, as a consequence of [20, Theorem 11.59]. Before quoting this result, we need some definitions from [20]. Let $\mathbb{R}_{+\infty} := \mathbb{R} \cup \{+\infty\}$ and $\mathbb{R}_{\pm\infty} := \mathbb{R} \cup \{\pm\infty\}$. Given $\varphi : \mathbb{R}^n \to \mathbb{R}_{+\infty}$, consider the optimization problem

$$\text{Minimize } \varphi(x) \text{ over all } x \text{ in } \mathbb{R}^n. \tag{2}$$

A function $g : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}_{\pm\infty}$ is said to be a *duality parameterization* for Problem (2) when $\varphi(\cdot) = g(\cdot, 0)$. The *perturbation function* associated with g is $p(v) := \inf_{x \in \mathbb{R}^n} g(x,v)$. Any function $\sigma : \mathbb{R}^m \to \mathbb{R}_{+\infty}$ which is proper, convex and lower semicontinuous is said to be an *augmenting function* if

$$\sigma \geq 0, \quad \min \sigma = 0, \quad \operatorname{Argmin} \sigma = \{0\}.$$

The *augmented Lagrangian* $\bar{l} : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}_+ \to \mathbb{R}_{\pm\infty}$ corresponding to the duality parameterization g and the augmenting function σ is given by

$$\bar{l}(x,u,c) := \inf_{v \in \mathbb{R}^m} \left[g(x,v) + c\sigma(v) - \langle u, v \rangle \right]. \tag{3}$$

The *dual function* induced by the augmented Lagrangian $\bar{l}$ is

$$\widetilde{H}(u,c) := \inf_x \bar{l}(x,u,c).$$

So the (augmented) dual problem becomes

$$(D) \qquad \max_{u \in \mathbb{R}^m,\, c \geq 0} \widetilde{H}(u,c).$$

A duality parameterization $g : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}_{\pm\infty}$ is said to be *level bounded in x locally uniformly in v* if for each $\bar{v} \in \mathbb{R}^m$ and each $\beta \in \mathbb{R}$, there exists a neighbourhood $W \subset \mathbb{R}^m$ of $\bar{v}$ such that for all $w \in W$ we have that

$$\{x \in \mathbb{R}^n \;:\; g(x,w) \leq \beta\} \subset B,$$

where $B \subset \mathbb{R}^n$ is a bounded set.

Theorem 2.1 *([20, Theorem 11.59]) Consider a duality parameterization $g : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}_{\pm\infty}$ for Problem (2), an augmenting function $\sigma : \mathbb{R}^m \to \mathbb{R}_{+\infty}$, and its associated augmented Lagrangian $\bar{l}$ as in (3). Assume that the following hypotheses hold.*

(i) g is level bounded in x locally uniformly in v.

(ii) $\inf_{x \in \mathbb{R}^n} \bar{l}(x,u,c) > -\infty$ for at least one $(u,c) \in \mathbb{R}^m \times \mathbb{R}_+$.

Then

(a) zero duality holds, i.e., $\inf_x \varphi(x) = \sup_{u,c} \widetilde{H}(u,c)$,

(b) Primal and (augmented) dual solutions (i.e., solutions of (D)) are characterized as saddle points of the augmented Lagrangian:

$$\begin{array}{c} \overline{x} \in \operatorname{Argmin}_x \varphi(x) \quad \textit{and} \quad (\overline{u}, \bar{c}) \in \operatorname{Argmax}_{u,c} \widetilde{H}(u,c) \\ \iff \quad \inf_x \bar{l}(x, \overline{u}, \bar{c}) = \bar{l}(\overline{x}, \overline{u}, \bar{c}) = \sup_{u,c} \bar{l}(\overline{x}, u, c). \end{array}$$

Given a set $A \subset \mathbb{R}^n$, the *indicator function* $\delta_A : \mathbb{R}^n \to \mathbb{R}_{+\infty}$ is defined as $\delta_A(x) = 0$ when $x \in A$ and $\delta_A(x) = +\infty$ otherwise. The duality properties of (P) can be obtained applying Theorem 2.1 with $\varphi := f_0 + \delta_{X_0}$, where X_0 is given as in (1). We use as augmenting function $\sigma(v) = \|v\|$, so our augmented Lagrangian is the *sharp augmented Lagrangian* (see [20, Example 11.58]). Define the duality parameterization $g : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}_{+\infty}$ as

$$g(x,v) = \begin{cases} f_0(x), & \text{if } x \in X \text{ and } f(x) = v, \\ +\infty, & \text{if } x \notin X \text{ or } f(x) \neq v. \end{cases}$$

It is clear that $\varphi = g(\cdot, 0)$. Moreover, since X is compact, it is easy to check that g is is level bounded in x locally uniformly in v. It is also straightforward to verify that the augmented Lagrangian $\bar{l} : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}_+ \to \mathbb{R}_{\pm\infty}$ associated with these choices of g and σ is

$$\bar{l}(x, u, c) = \begin{cases} f_0(x) + c\|f(x)\| - \langle u, f(x)\rangle, & \text{if } x \in X, \\ +\infty, & \text{if } x \notin X. \end{cases}$$

Therefore we can write $\widetilde{H}(u, c) = \inf_x \bar{l}(x, u, c) = \inf_{x\in X} f_0(x) + c\|f(x)\| - \langle u, f(x)\rangle = H(u, c) > -\infty$ for every (u, c) because the right-hand infimum is the minimization of a continuous function over a compact set. Therefore all hypotheses of Theorem 2.1 hold and conclusions (a) and (b) are valid for our scheme. Hence $\overline{H} = \underline{f_0}$. Thus the dual function H can be rewritten as ("inf" replaced by "min")

$$H(u, c) = \min_{x\in X} \left[f_0(x) + c\|f(x)\| - \langle u, f(x)\,\rangle\right]. \tag{4}$$

Since H is the minimum of concave and upper-semicontinuous (more precisely, affine) functions of (u, c), we conclude that H is concave and upper-semicontinuous. Because X is compact, H is finite everywhere. Using now the concavity, we conclude that H is continuous everywhere.

In what follows, we use the notation

$$\begin{aligned} S(P) &:= \{x \in \mathbb{R}^n \;:\; x \text{ solves } (P)\}, \\ S(D) &:= \{(u, c) \in \mathbb{R}^m \times \mathbb{R}_+ \;:\; (u, c) \text{ solves } (D)\}. \end{aligned}$$

We denote a typical element of $S(P)$ by $\bar{x}$, and a typical element of $S(D)$ by $\bar{z} = (\bar{u}, \bar{c})$. For convenience, we introduce the set

$$X(u, c) = \operatorname*{Argmin}_{x\in X} \left[f_0(x) + c\|f(x)\| - \langle u, f(x)\rangle\right] \;. \tag{5}$$

The minimization problem given in (4) is neither convex nor differentiable, so this problem might be very difficult. Therefore, it is convenient to develop a scheme which accepts approximate solutions of (5). Recently, methods for problem (P) which use approximate solutions of the Lagrangian dual were introduced in [24, 2, 15]. The Lagrangians studied in those papers, however, do not include the sharp augmented Lagrangian.

Let $r \geq 0$ and define the set

$$X_r(u, c) := \{x \in X \;:\; L(x, (u, c)) \leq H(u, c) + r\}. \tag{6}$$

In other words, $x \in X_r(u, c)$ if and only if it is an *r-minimizer* of the augmented Lagrangian.

We recall now two well-known tools from convex analysis. Given a concave function $H : \mathbb{R}^m \times \mathbb{R}_+ \to \mathbb{R}$ and a fixed $(u, c) \in \mathbb{R}^m \times \mathbb{R}_+$, the set

$$\begin{aligned} &\partial H(u, c) := \\ &\{(v, a) \in \mathbb{R}^m \times \mathbb{R} \;:\; H(u', c') \leq H(u, c) + \langle u' - u, v\rangle + a(c' - c),\ \forall\ (u', c') \in \mathbb{R}^m \times \mathbb{R}_+\} \end{aligned}$$

is called the *subdifferential* of H at (u, c), and each element of this set is called a *subgradient* of H at (u, c) (the terms "superdifferential" and "supergradient", respectively, are also used). Let $\varepsilon \geq 0$. The set

$$\partial_\varepsilon H(u,c) := \\ \{(v,a) \in \mathbb{R}^m \times \mathbb{R} \,:\, H(u',c') \leq H(u,c) + \langle c'-c, v\rangle + a(c'-c) + \varepsilon,\ \forall\ (u',c') \in \mathbb{R}^m \times \mathbb{R}_+\}$$

is called the *ε-subdifferential* of H at (u,c), and each element of this set is called an *ε-subgradient* of H at (u,c).

The simple fact below will be useful in the sequel.

Lemma 2.1 *Let H be the dual function defined in* (4) *and let $r \geq 0$. Let $(u,c) \in \mathbb{R}^m \times \mathbb{R}_+$, and let $\tilde{x} \in X_r(u,c)$. Then*

- *(a)* $(-f(\tilde{x}), \|f(\tilde{x})\|(1+\gamma)) \in \partial_{c\gamma\|f(\tilde{x})\|+r} H(u,c)$ *for every $\gamma \geq 0$. In particular, we have that* $(-f(\tilde{x}), \|f(\tilde{x}))\|) \in \partial_r H(u,c)$.
- *(b) If $(u,c) \in S(D)$, then $(u,d) \in S(D)$ for each $d > c$. In this situation, for every $\hat{x} \in X(u,d)$ we must have $f(\hat{x}) = 0$.*

Proof. (a) We must prove that, for all $(u',c') \in \mathbb{R}^m \times \mathbb{R}_+$,

$$H(u',c') \leq H(u,c) + \langle u'-u, -f(\tilde{x})\rangle + (c'-c)(1+\gamma)\|f(\tilde{x})\| + r + c\gamma\|f(\tilde{x})\|.$$

Indeed, by definition of H we have

$$\begin{aligned} H(u',c') &= \min_{x\in X} L(x,(u',c')) \\ &\leq f_0(\tilde{x}) + c\|f(\tilde{x})\| - \langle u, f(\tilde{x})\rangle \\ &\quad + (c'-c)\|f(\tilde{x})\| + \langle u'-u, -f(\tilde{x})\rangle \\ &\leq H(u,c) + r + \langle u'-u, -f(\tilde{x})\rangle + (c'-c)(1+\gamma)\|f(\tilde{x})\| - (c'-c)\gamma\|f(\tilde{x})\| \\ &\leq H(u,c) + r + \langle u'-u, -f(\tilde{x})\rangle + (c'-c)(1+\gamma)\|f(\tilde{x})\| + c\gamma\|f(\tilde{x})\|, \end{aligned}$$

where we used the definition of $\tilde{x}$ in the second inequality and the fact that $c' \geq 0$ in the last inequality. The second statement follows from the first one for $\gamma = 0$. The proof of (a) is complete.
(b) Since $(u,c) \in S(D)$, we must have $H(u,c) \geq H(u,d)$. On the other hand, take $\hat{x} \in X(u,d)$, where $d > c$. By item (a) for $\gamma = 0$, we have

$$\begin{aligned} H(u,d) \leq H(u,c) &\leq H(u,d) + \langle u-u, -f(\hat{x})\rangle + (c-d)\|f(\hat{x})\| \\ &= H(u,d) + (c-d)\|f(\hat{x})\| \leq H(u,d). \end{aligned}$$

Altogether, $H(u,c) = H(u,d)$ and hence (u,d) is also a dual solution. Since $d - c > 0$ we must have $f(\hat{x}) = 0$. □

Definition 2.1 *We say that $x \in X$ is an r-*solution of (P) *if $f(x) = 0$ and $f_0(x) \leq \underline{f_0} + r$. We say that $(u,c) \in \mathbb{R}^m \times \mathbb{R}_+$ is an r-*solution of (D) *when $\overline{H} - r \leq H(u,c)$.*

The result below justifies the stopping criterion used in the IMSG algorithm we will describe in the next section.

Theorem 2.2 *Let* $(\overline{u}, \overline{c}) \in \mathbb{R}^m \times \mathbb{R}_+$, $r \geq 0$ *and suppose that* $\overline{x} \in X_r(\overline{u}, \overline{c})$ *and* $f(\overline{x}) = 0$. *Then* $\overline{x}$ *is an* r*-solution of* (P) *and* $(\overline{u}, \overline{c})$ *is an* r*-solution of* (D).

Proof. Assume that $f(\bar{x}) = 0$ and $\bar{x} \in X_r(\bar{u}, \bar{c})$, then $L(\bar{x}, (\bar{u}, \bar{c})) = f_0(\bar{x}) \geq \underline{f_0}$. On the other hand, $L(\bar{x}, (\bar{u}, \bar{c})) \leq H(\bar{u}, \bar{c}) + r \leq \underline{f_0} + r$, where we used weak duality in the last inequality. Thus $\bar{x}$ is an r-solution of (P) (see Definition 2.1). On the other hand, noting that $\overline{H} = \underline{f_0}$ and $f(\overline{x}) = 0$ we can write

$$\overline{H} - r = \underline{f_0} - r \leq f_0(\overline{x}) - r = L(\overline{x}, (\bar{u}, \bar{c})) - r \leq H(\bar{u}, \bar{c}) \leq \overline{H},$$

so $(\overline{u}, \overline{c})$ is an r-solution of (D) (see Definition 2.1). □

3 The IMSG Algorithm

The distance between two given points in the dual space, $w, z \in \mathbb{R}^m \times \mathbb{R}_+$, will be taken as

$$\rho(w, z) := \|w - z\|^2 .$$

For brevity, we will use the following short-hand notation.

$$\begin{aligned}
&z_k := (u_k, c_k), \\
&x_k \in X_{r_k}(u_k, c_k), \\
&f_k := f(x_k), \\
&H_k := H(u_k, c_k), \\
&\overline{H} := H(\overline{u}, \overline{c}), \quad \text{where } (\overline{u}, \overline{c}) \in S(D), \\
&\rho_k := \rho(z, z_k).
\end{aligned}$$

We propose below an inexact version of the MSG algorithm, namely the IMSG algorithm, for solving Problem (P).

The IMSG Algorithm:

Step 0 (Initialization) Let $r^* > 0$ be a fixed prescribed tolerance. Choose $r_0 \geq 0$ and (u_0, c_0) with $c_0 \geq 0$. Let $\{r_k\}$ be a nonnegative sequence. Set $k = 0$.

Step k.0 (Subproblem and Stopping Criterion)

(a) Find $x_k \in X_{r_k}(u_k, c_k)$.

(b) If $f_k = 0$ and $r_k \leq r^*$ then STOP.

(c) If $f_k = 0$ and $r_k > r^*$ then $r_k := r_k/2$ and GO TO **(a)**.

Step k.1 (Update of Dual Variables) Set

$$\begin{aligned}
u_{k+1} &:= u_k - s_k f_k \, , \\
c_{k+1} &:= c_k + (s_k + \varepsilon_k)\|f_k\| \, ,
\end{aligned}$$

where the step-sizes $s_k, \varepsilon_k > 0$. Set $k = k + 1$ and GO TO **Step k.0**.

Remark 3.1 Employment of $\varepsilon_k > 0$ in Step k.1 of the IMSG algorithm allows a deflection of the "classical" r_k-subgradient direction $(-f_k, \|f_k\|)$ associated with the sharp augmented Lagrangian as described in Lemma 2.1(a) with $\gamma = 0$. This deflection is needed to ensure that the dual function iterates have strictly increasing values (see Proposition 3.1(c) below).

Remark 3.2 It should be noted again that Step k.0(a) may be rather hard to carry out.

Remark 3.3 Note that the IMSG algorithm is stated above for arbitrary step-size parameters $s_k, \varepsilon_k > 0$ and arbitrary error $r_k \geq 0$. In this section, some properties of the algorithm will be stated and proved for these arbitrary step-sizes and error. On the other hand, convergence and existence results will be obtained in Section 4 under step-sizes s_k, ε_k and error r_k satisfying Assumptions (A1)-(A2) on page 10. In fact, these assumptions force r_k to converge to 0 (see Theorem 4.2).

Remark 3.4 We say that the IMSG algorithm performs a *serious step* whenever Step k.1 is visited. The loop in Step k.0 (which occurs if $f_k = 0$ and $r_k > r^*$) is called a *null step.* It asks for a more accurate result from the global search performed in Step k.0(a).

Remark 3.5 Note that Step k.1 is visited only when $f_k \neq 0$. By Theorem 2.2, IMSG either stops with an r^*-solution of (P) and an r^*-solution of (D) (see Proposition 3.1(b) below) or it produces an infinite sequence of primal-dual iterates $\{(x_k, (u_k, c_k))\}$ such that $f_k \neq 0$.

Proposition 3.1 *Consider the notation and definitions of the IMSG algorithm with the arbitrary step-size parameters $s_k, \varepsilon_k > 0$ and arbitrary $r_k \geq 0$. Denote by $\overline{H}$ the optimal value of (P). Then the following properties hold.*

(a) $H_{k+1} - H_k \leq r_k + (2s_k + \varepsilon_k)\|f_k\|^2.$

(b) If IMSG stops at iteration k, then the primal dual pair $(x_k, (u_k, c_k))$ is an r^*-solution, i.e.,

$$f(x_k) = 0, \quad \overline{H} \leq f_0(x_k) \leq \overline{H} + r^*, \quad \overline{H} - r^* \leq H(u_k, c_k) \leq \overline{H}.$$

(c) If IMSG does not stop at iteration k, then either $(u_k, c_k) \in S(D)$ or

$$H_{k+1} > H_k.$$

Proof. Applying Lemma 2.1(a) for $\gamma = 0$, we get

$$H_{k+1} - H_k \leq r_k + \langle u_{k+1} - u_k, -f_k \rangle + (c_{k+1} - c_k)\|f_k\|.$$

Using also the definition of the algorithm, the right-hand side of the expression above can be rewritten as

$$H_{k+1} - H_k \leq r_k + s_k\|f_k\|^2 + (s_k + \varepsilon_k)\|f_k\|^2,$$

which is (a). In order to prove (b), assume that the algorithm stops at the kth iteration. Then it is clear that $f_k = 0$ and $r_k < r^*$. The conclusion now follows from Theorem 2.2

for $r := r_k$. To prove (c), assume now that the algorithm does not stop at iteration k and suppose that $(u_k, c_k) \notin S(D)$. Because Step k.1 is visited we must have $f_k \neq 0$. Using the definition of the algorithm we can write:

$$\begin{aligned}
H_{k+1} &= \min_{x\in X} [f_0(x) + c_{k+1} \|f(x)\| - \langle u_{k+1}, f(x)\rangle] \\
&= \min_{x\in X} [f_0(x) + [c_k + (s_k + \varepsilon_k)\|f_k\|] \|f(x)\| - \langle [u_k - s_k f_k], f(x)\rangle] \\
&= \min_{x\in X} [f_0(x) + c_k \|f(x)\| - \langle u_k, f(x)\rangle + (s_k + \varepsilon_k)\|f_k\| \|f(x)\| \\
&\quad + s_k\langle f_k, f(x)\rangle] \\
&\geq \min_{x\in X} [f_0(x) + c_k \|f(x)\| - \langle u_k, f(x)\rangle + (s_k + \varepsilon_k)\|f_k\| \|f(x)\| \\
&\quad - s_k\|f_k\| \|f(x)\|] \\
&= \min_{x\in X} [f_0(x) + (c_k + \varepsilon_k\|f_k\|) \|f(x)\| - \langle u_k, f(x)\rangle] = H(u_k, c_k + \varepsilon_k\|f_k\|).
\end{aligned}$$

Let $\hat{x}_k \in X$ be a solution of the minimization problem above. In other words $\hat{x}_k \in X(u_k, c_k + \varepsilon_k\|f_k\|)$. Assume first that $f(\hat{x}_k) = 0$. In this case Theorem 2.2 for $r = 0$ yields $(u_k, c_k + \varepsilon_k\|f_k\|) \in S(D)$. On the other hand, since $(u_k, c_k) \notin S(D)$ we must have $H(u_k, c_k) < H(u_k, c_k + \varepsilon_k\|f_k\|) \leq H_{k+1}$. Therefore the conclusion holds in this case. Assume now that $f(\hat{x}_k) \neq 0$. The above expression and the definition of $\hat{x}_k$ yield

$$\begin{aligned}
H_{k+1} &\geq H(u_k, c_k + \varepsilon_k\|f_k\|) = L(\hat{x}_k, (u_k, c_k)) + \varepsilon_k\|f_k\| \|f(\hat{x}_k)\| \\
&\geq \min_{x\in X} L(x, (u_k, c_k)) + \varepsilon_k\|f_k\| \|f(\hat{x}_k)\| = H_k + \varepsilon_k\|f_k\| \|f(\hat{x}_k)\| \\
&> H_k,
\end{aligned}$$

where we used $f_k \neq 0$ and $\varepsilon_k > 0$. This proves (c). □

Remark 3.6 Note that the steps might still be performed when $(u_k, c_k) \in S(D)$, the value of H in that case cannot increase, but the sequence $\{x_k\}$ changes during the process.

The following lemma, taken from [4], establishes a necessary and sufficient condition on the step-sizes s_k and ε_k for guaranteeing boundedness of the dual sequence.

Lemma 3.1 *Consider the notation and definitions of the IMSG algorithm with the arbitrary step-sizes $s_k, \varepsilon_k > 0$. The following statements are equivalent.*

(a) $\sum_{k=o}^{\infty}(s_k + \varepsilon_k)\|f_k\| < \infty$.

(b) The sequence $\{z_k\}$ is bounded.

4 Existence and Convergence

In this section, we prove convergence of the IMSG algorithm, and establish necessary and sufficient conditions for existence of dual solutions. In order to guarantee boundedness of the dual sequence generated by the algorithm, we use in our analysis the following assumptions on the sequence of step-sizes $\{s_k\}$ and the sequence $\{r_k\}$.

(A1) $s_k \geq \dfrac{\eta(\overline{H} - H_k) + \theta r_k}{\|f_k\|^2}$, for some fixed $\eta, \theta > 0$.

(A2) The sequence $\{s_k \|f_k\|\}$ is bounded.

Assumption (A1) is in the spirit of the classical dynamic step-size rule for subgradient methods (see, e.g. [19, 22, 17, 15]). Assumption (A2) is used in [12, Theorem 4.1] in the context of approximate subgradient methods for coercive problems. Assumption (A2) ensures that the step-size s_k remains small enough to guarantee boundedness of $\{z_k\}$, while (A1) ensures that s_k is not too small.

Remark 4.1 Note that the right-hand side of the inequality in (A1) requires the knowledge of $\overline{H}$ and H_k, which in principle are not available. Therefore, in Section 5 we propose a practical step-size (S) which is proved to satisfy (A1). The step-size rule (S) does not require the knowledge of $\overline{H}$, but uses a sequence $\{\widehat{H}_k\}$ such that $\widehat{H}_k \geq \overline{H}$ for all k.

Before establishing our convergence results, we need the following useful estimate.

Lemma 4.1 *Fix $z = (u, c) \in \mathbb{R}^m \times \mathbb{R}_+$. The following estimate holds.*

$$\|u - u_{k+1}\|^2 \leq \|u - u_k\|^2 + s_k^2 \, \|f_k\|^2 + 2s_k \left[r_k - (H(u, c) - H_k) + (c - c_k)\|f_k\|\right].$$

Proof. The definition of the algorithm yields

$$\begin{aligned} \|u - u_{k+1}\|^2 &= \|u - u_k + s_k f_k\|^2 \\ &= \|u - u_k\|^2 + 2s_k\langle u - u_k, f_k\rangle + s_k^2 \, \|f_k\|^2 . \end{aligned} \tag{7}$$

Using the r_k-subgradient inequality we obtain

$$\langle u - u_k, f_k\rangle \leq (c - c_k)\|f_k\| + H_k - H(u, c) + r_k \ ,$$

which combined with (7) yields the lemma. □

Theorem 4.1 *Assume that (A1) holds. Then the following statements are equivalent.*

(a) The sequence $\{z_k\}$ is bounded.

(b) $S(D) \neq \emptyset$ and Assumption (A2) holds.

Proof. Assume that (a) holds. Using (A1) and Lemma 3.1 we can write

$$0 = \lim_{k\to\infty} s_k\|f_k\| \geq \eta \limsup_{k\to\infty} \frac{\overline{H} - H_k}{\|f_k\|} \geq 0. \tag{8}$$

Noting that the sequence $\{f_k\}$ is bounded (because $x_k \in X_{r_k}(z_k) \subset X$, X is compact and f is continuous) we conclude that $H_k \to \overline{H}$. Take $\hat{z}$ an accumulation point of $\{z_k\}$. Since H is continuous, we must have $H(\hat{z}) = \overline{H}$ and hence $\hat{z} \in S(D)$. Thus $S(D) \neq \emptyset$. From Lemma 3.1, we have that the sequence $\{s_k\|f_k\|\}$ tends to zero. The latter fact readily yields (A2). Suppose now that (b) holds and take $(\overline{u},\overline{c}) \in S(D)$. For contradiction purposes, assume that $\{z_k\}$ is unbounded. As a consequence of Lemma 3.1, an unbounded $\{z_k\}$ necessarily implies that $\{c_k\}$ is unbounded. Lemma 4.1 for $(u,c)=(\overline{u},\overline{c})$ gives

$$\begin{aligned}\|\overline{u}-u_{k+1}\|^2 &\leq \|\overline{u}-u_k\|^2 + s_k\,\|f_k\|\left[s_k\|f_k\| - \frac{2(\overline{H}-H_k)}{\|f_k\|} + 2(\overline{c}-c_k) + 2\frac{r_k}{\|f_k\|}\right]\\ &\leq \|\overline{u}-u_k\|^2 + s_k\,\|f_k\|\left[s_k\|f_k\| + 2(\overline{c}-c_k) + 2\frac{r_k}{\|f_k\|}\right], \qquad (9)\end{aligned}$$

where we also used the fact that $H(\overline{u},\overline{c})=\overline{H}\geq H_k$ for all k. By (A1) we have that

$$s_k\|f_k\| \geq \theta\frac{r_k}{\|f_k\|}.$$

Using also (A2) we conclude that $\{r_k/\|f_k\|\}$ is bounded. Since $\{c_k\}$ tends to infinity and (A2) holds, there exists k_0 such that for every $k\geq k_0$ the expression between brackets in (9) is nonpositive. Altogether, for every $k\geq k_0$ we have

$$\|\overline{u}-u_{k+1}\| \leq \|\overline{u}-u_k\|.$$

We conclude that $\|\overline{u}-u_k\| \leq \|\overline{u}-u_0\|$ for every $k\geq k_0$. From the r_k-subgradient inequality we get (see Lemma 2.1(a))

$$\overline{H} \leq H_k + \langle \overline{u}-u_k, -f_k\rangle + (\overline{c}-c_k)\|f_k\| + r_k.$$

So,

$$\begin{aligned}(c_k-\overline{c})\|f_k\| &\leq -(\overline{H}-H_k) + \|\overline{u}-u_k\|\;\|f_k\| + r_k\\ &\leq \|\overline{u}-u_k\|\;\|f_k\| + r_k.\end{aligned}$$

Dividing the above expression by $\|f_k\|$ on both sides, we obtain

$$c_k \leq \overline{c} + \|\overline{u}-u_k\| + \frac{r_k}{\|f_k\|} \leq \overline{c} + \|\overline{u}-u_0\| + \frac{r_k}{\|f_k\|} \quad \text{for all } k\geq k_0. \qquad (10)$$

Since the right-hand side of (10) is bounded, this contradicts the unboundedness of $\{c_k\}$. Therefore, the sequence $\{z_k\}$ must be bounded. The proof is complete. □

Theorem 4.2 *(Dual Convergence) Assume that $S(D)\neq\emptyset$. Suppose (A1)-(A2) hold. Then $\{z_k\}$ converges to a dual solution. In particular, H_k tends to $\overline{H}$ and r_k tends to 0. Moreover, if $\{x_k\}$ has an accumulation point $\bar{x}$ such that $f(\bar{x})=0$, then $\bar{x}\in S(P)$.*

Proof. From Theorem 4.1, part (b)→(a), we conclude that $\{z_k\}$ is bounded, and every accumulation point belongs to $S(D)$. Fix $\hat{z}=(\hat{u},\hat{c})$ an accumulation point of $\{z_k\}$ and let $\{z_{k_j}\}_j$ be a subsequence converging to $\hat{z}$. Define the sequence $\{\rho(\hat{z},z_k)\}$, where $\rho(\hat{z},z_k) := \|\hat{u}-u_k\|^2 + |\hat{c}-c_k|^2$. Step k.1 and trivial manipulations yield

$$\begin{aligned}\rho(\hat{z},z_{k+1}) - \rho(\hat{z},z_k) &= (s_k^2 + (s_k+\varepsilon_k)^2)\|f_k\|^2 + 2s_k\left[\langle \hat{u}-u_k, f_k\rangle + (c_k-\hat{c})\|f_k\|\right]\\ &\quad +2\varepsilon_k(c_k-\hat{c})\|f_k\|. \end{aligned} \qquad (11)$$

From the definition of the algorithm, we know that the sequence $\{c_k\}$ is increasing, with the convergent subsequence $\{c_{k_j}\}$. This implies that the whole sequence must converge to $\hat{c}$. Altogether, we have

$$\hat{c} = \lim_{i\to\infty} c_i = \sup_{i\to\infty} c_i \geq c_k, \quad \forall\, k.$$

The above expression implies that the last term in (11) is nonpositive, which gives

$$\rho(\hat{z}, z_{k+1}) - \rho(\hat{z}, z_k) \;\leq\; (s_k^2 + (s_k + \varepsilon_k)^2)\|f_k\|^2 + 2s_k\left[\langle \hat{u} - u_k, f_k\rangle + (c_k - \hat{c})\|f_k\|\right] \quad (12)$$

Use the r_k-subgradient inequality to obtain

$$\overline{H} = H(\hat{z}) \leq H_k - \langle \hat{u} - u_k, f_k\rangle + (\hat{c} - c_k)\|f_k\| + r_k. \quad (13)$$

From Proposition 3.1(c), the sequence of dual values $\{H_k\}$ is increasing and hence

$$\overline{H} = H(\hat{z}) = \lim_{i\to\infty} H_i = \sup_{i\to\infty} H_i \geq H_k, \quad \forall\, k.$$

This fact, together with (13) yields

$$2s_k\left[\langle \hat{u} - u_k, f_k\rangle + (c_k - \hat{c})\|f_k\|\right] \leq 2s_k(H_k - \overline{H}) + 2s_k r_k \leq 2s_k r_k,$$

Combining the above expression with (12) gives

$$\rho(\hat{z}, z_{k+1}) - \rho(\hat{z}, z_k) \leq (s_k^2 + (s_k + \varepsilon_k)^2)\|f_k\|^2 + 2s_k r_k.$$

Boundedness of the sequence $\{z_k\}$ and (A1) imply that the right-hand side of the expression above is summable. Indeed, (A1) yields $s_k^2\|f_k\|^2 \geq \theta s_k r_k$. By Lemma 3.1 we have that $\{(s_k+\varepsilon_k)^2\|f_k\|^2\}$ is summable, and hence $\{s_k r_k\}$ is also summable. Altogether, $\lim_{k\to\infty} \rho(\hat{z}, z_k)$ exists. Because a subsequence of $\{\rho(\hat{z}, z_k)\}$ tends to zero, the whole sequence must tend to zero. We thus obtain full convergence of the dual sequence to a dual solution. Note that (A1) yields $s_k\|f_k\| \geq \theta r_k/\|f_k\|$. An argument similar to the one used in (8) yields $\lim_k r_k = 0$. In order to establish the statement on the primal sequence, assume that $\{x_k\}$ has an accumulation point $\overline{x}$ such that $f(\overline{x}) = 0$. Call $\{x_{k_j}\}$ the subsequence converging to $\overline{x}$. Because $\lim_j r_{k_j} = 0$, using the definition of x_{k_j} we have

$$H_{k_j} \leq L(x_{k_j}, z_{k_j}) \leq H_{k_j} + r_{k_j}.$$

Taking limits we obtain $\underline{f_0} \leq f_0(\overline{x}) = L(\overline{x}, \hat{z}) \leq \underline{f_0}$ with $f(\overline{x}) = 0$. This fact clearly yields $\overline{x} \in S(P)$. The proof is complete. □

Remark 4.2 Theorem 4.2 guarantees convergence of the dual sequence to a dual solution and optimality of a feasible accumulation point $\overline{x}$ (i.e., $f(\overline{x}) = 0$) of the primal sequence $\{x_k\}$. Without the assumption $f(\overline{x}) = 0$ nothing can be said about convergence of $\{x_k\}$. An example illustrating this fact for the exact version of the IMSG algorithm (i.e., with the choice $r_k = 0$ for all k) is given in [4, Example 1]. However, if we consider a *perturbed sequence* $\{\tilde{x}_k\}$, defined by the inclusion

$$\tilde{x}_k \in X_{r_k}(u_k, c_k + \beta) \quad \text{for a fixed } \beta > 0,$$

then optimality of all accumulation points of $\{\tilde{x}_k\}$ can be established. The latter result was proved for the exact version of the IMSG algorithm in [4].

We extend the result mentioned in the remark above to the case of the IMSG algorithm in the theorem below.

Theorem 4.3 *(Primal Convergence) Assume that $S(D) \neq \emptyset$. Suppose (A1)-(A2) hold. Then all accumulation points of the perturbed sequence $\{\tilde{x}_k\}$ are solutions of (P).*

Proof. From Theorem 4.2 we have that $\lim_{k\to\infty} r_k = 0$ and the sequence $\{z_k\}$ converges to a dual solution. Fix $\beta > 0$ and take $\tilde{x}_k \in X_{r_k}(u_k, c_k + \beta)$ for all k. Call $\|\tilde{f}_k\| := \|f(\tilde{x}_k)\|$. Take $a \geq 0$ an accumulation point of the sequence $\{\|\tilde{f}_k\|\}$. So there exists a subsequence $\{\|\tilde{f}_{k_j}\|\}$ such that $a = \lim_{j\to\infty} \|\tilde{f}_{k_j}\|$. Lemma 2.1(a) yields

$$\begin{aligned} H(u_{k_j}, c_{k_j}) = H_{k_j} &\leq H(u_{k_j}, c_{k_j} + \beta) + \langle(-\tilde{f}_{k_j}, \|\tilde{f}_{k_j}\|), (\mathbf{0}, -\beta)\rangle + r_{k_j} \\ &\leq H(u_{k_j}, c_{k_j} + \beta) - \beta\|\tilde{f}_{k_j}\| + r_{k_j} \end{aligned}$$

We can rewrite this as

$$\beta\|\tilde{f}_{k_j}\| \leq H(u_{k_j}, c_{k_j} + \beta\|f_{k_j}\|) - H_{k_j} + r_{k_j} \leq \overline{H} - H_{k_j} + r_{k_j}.$$

Using the fact that $\lim_j \overline{H} - H_{k_j} = \lim_j r_{k_j} = 0$ in the expression above yields

$$a = \lim_{j\to\infty} \|\tilde{f}_{k_j}\| = 0 \ . \tag{14}$$

Thus the sequence $\{\|\tilde{f}_k\|\}$ converges to zero. Take now $\tilde{x}$ as an accumulation point of $\{\tilde{x}_k\}$. Since zero is the limit of $\{\|\tilde{f}_k\|\}$, we must have $f(\tilde{x}) = 0$. Without loss of generality, assume that the whole sequence $\{\tilde{x}_k\}$ converges to $\tilde{x}$. Because our scheme has no duality gap and $\tilde{x}$ is feasible, we can write

$$\begin{aligned} \overline{H} \leq f_0(\tilde{x}) &= \lim_k f_0(\tilde{x}_k) + (c_k + \beta)\|\tilde{f}_k\| - \langle u_k, f(\tilde{x}_k)\rangle \\ &\leq \lim_k H(u_k, c_k + \beta) + r_k \leq \overline{H} \end{aligned}$$

where we have used (14), the fact that $\{z_k\}$ is bounded in the first equality and the definition of $\tilde{x}_k$ in the first inequality. The above expression yields $f_0(\tilde{x}) = \overline{H}$ and since $f(\tilde{x}) = 0$, $\tilde{x}$ is a primal solution. □

5 A Practical Step-Size Selection for Implementation

In the preceding section, we obtained convergence results for the IMSG algorithm under the broad step-size rule given in (A1). It is not practical simply to choose the step-size s_k to be equal to the right-hand side of the inequality in (A1). There are three issues of concern for not being able to make this choice, the issues we state and address below.

(i) H_k is not known. In fact, the quest of this paper is to avoid calculating H_k accurately so as to achieve computational savings. Let

$$L_k := L(x_k, (u_k, c_k))$$

and recall that

$$H_k \le L_k \le H_k + r_k \ . \tag{15}$$

In the implementation of the algorithm, we compute L_k, which is an inexact value of H_k associated with the inexactness parameter r_k. Our concern that H_k is not known is addressed by making use of the relationship in (15). We utilize L_k, instead of H_k, in the practical step-size (S) we are to devise in Proposition 5.1.

(ii) In optimization software, one can typically specify a termination tolerance on the value of the function being minimized, the tolerance we denote by $\widehat{r}_k$. In the practical step-sizes we are to propose, r_k is replaced by $\widehat{r}_k$. We also assume that a smaller $\widehat{r}_k$ corresponds to a smaller r_k. Numerical experiments and *a posteriori* computations indicate that this assumption is valid in general.

(iii) In subgradient methods, in the case when $\overline{H}$ is not known, common practice is to use an estimate $\widehat{H}$ of $\overline{H}$. Typically, one uses an upper bound of $\overline{H}$ as the estimate $\widehat{H}$, which can be obtained by evaluating the cost at a feasible point. In [1, 21], a dynamic approach is taken: $\widehat{H}$ is updated in each iteration. In our study, we consider a sequence $\{\widehat{H}_k\}$ of upper bound estimates of $\overline{H}$ (see Proposition 5.1).

The step-size parameter ε_k of the IMSG algorithm is prescribed in the numerical implementation here in the same way as in [4]:

$$\varepsilon_k = \alpha\, s_k\,, \qquad \alpha > 0\ . \tag{16}$$

Lemma 5.1 *For some fixed $\eta, \theta > 0$, if s_k satisfies*

$$s_k \ge \frac{\eta\,(\overline{H} - L_k) + (\eta+\theta)\, r_k}{\|f_k\|^2}\ ,$$

then it also satisfies Assumption (A1).

Proof. The lemma follows by virtue of (15) and straightforward manipulations. □

Proposition 5.1 *Suppose that, for some fixed $\mu > 0$, the conditions*

$$L_k \le \widehat{H}_k, \qquad \widehat{H}_k \ge \overline{H}, \qquad r_k \le \mu\,(\widehat{H}_k - L_k) \tag{17}$$

hold. Then the step-size

$$(\mathbf{S}): \qquad s_k = \delta\, \frac{\widehat{H}_k - L_k}{\|f_k\|^2}\ , \qquad \delta > 0\ ,$$

satisfies (A1).

Proof. Suppose that (17) holds. For any $\mu > 0$ and $\delta > 0$, there exist $\eta, \theta > 0$ such that $\mu = (\delta - \eta)/(\eta + \theta)$. Then

$$\begin{aligned} r_k &\le \frac{\delta - \eta}{\eta + \theta}\,(\widehat{H}_k - L_k) \\ (\delta - \eta)\,(\widehat{H}_k - L_k) &\ge (\eta+\theta)\, r_k \\ \delta\,(\widehat{H}_k - L_k) &\ge \eta\,(\widehat{H}_k - L_k) + (\eta+\theta)\, r_k \\ s_k &\ge \frac{\eta\,(\overline{H} - L_k) + (\eta+\theta)\, r_k}{\|f_k\|^2} \end{aligned}$$

because $\|f_k\| \neq 0$. We used in the last inequality the fact that $\widehat{H}_k \geq \overline{H}$. Now Lemma 5.1 furnishes the proposition. $\square$

Corollary 5.1 *Assume that $S(D) \neq \emptyset$ and (A2) holds. Under the step-size rule (S), the convergence results stated in Theorems 4.2 and 4.3 hold.*

Remark 5.1 The condition

$$r_k \leq \mu\,(\widehat{H}_k - L_k) \tag{18}$$

defines an adaptive rule for choosing r_k: if $L_k > \widehat{H}_k$, then r_k is reduced until after the condition $L_k \leq \widehat{H}_k$ is satisfied. Indeed, if $z_k \notin S(D)$, then

$$\widehat{H}_k \geq \overline{H} > H_k. \tag{19}$$

We claim that for small enough r_k, the inequality $L_k \leq \widehat{H}_k$ holds. Indeed, assume that, for a fixed iteration k, there exist a sequence of errors $\{{r_k}^l\}_l$ and a sequence $\{{x_k}^l\}_l$ such that $\lim_{l\to\infty} {r_k}^l = 0$ and

$$H_k \leq L({x_k}^l, z_k) \leq H_k + {r_k}^l \quad \text{for every } l \in \mathbb{N}. \tag{20}$$

Assume further that for every l we have

$$L({x_k}^l, z_k) > \widehat{H}_k. \tag{21}$$

Let $\overline{x}$ be an accumulation point of $\{{x_k}^l\}$. Taking limits in (20) for $l \to \infty$ we obtain $H_k \leq L(\overline{x}, z_k) \leq H_k$. So $\overline{x} \in X(z_k)$. Using also (21) we get $H_k = L(\overline{x}, z_k) \geq \widehat{H}_k \geq \overline{H}$, contradicting the right-hand inequality in (19). Hence, our claim is true and for small enough r_k, we have $L_k \leq \widehat{H}_k$.

Remark 5.2 Let $\widehat{H}$ be fixed such that $\widehat{H} \geq \overline{H}$. If we take the step-size

$$s_k = \delta \frac{\widehat{H} - L_k}{\|f_k\|^2},$$

then a proof similar to the one in Proposition 5.1 shows that Assumption (A1) is verified when $L_k \leq \overline{H} < \widehat{H}$.

Remark 5.3 It will be illustrated in the numerical experiments that for small enough r_k, Condition (18) is satisfied easily.

6 Numerical Experiments

In what follows we illustrate some advantages of the IMSG algorithm using three test problems, which have previously been solved using the MSG algorithm in [4]. In particular, we illustrate that important computational savings can be obtained with the IMSG algorithm over the MSG algorithm.

In all three test problems, we have utilized MATLAB's function m-file `fminsearch`. In `fminsearch`, two termination parameters are set, namely `tolfun`, which is the termination tolerance on the (Lagrangian) function value $L(x, u_k, c_k)$, and `tolx`, which is the termination tolerance on the optimization variable x. Because `tolfun` is a measure of the distance between the function iterates in `fminsearch`, in principle it is not directly related to r_k.

There may be instances where `fminsearch` would terminate immediately even for very small values of `tolfun`, but these instances could rather be related with the convergence properties of the Nelder-Mead simplex algorithm [13, 14, 18], which forms the basis of `fminsearch`. In practice both r_k and `tolfun` provide a degree of accuracy of H_k, the optimum value of $L(x, u_k, c_k)$, in the subproblem. So in our experiments we use `tolfun` in place of r_k. For brevity we denote $\widehat{r}_k =$ `tolfun`.

6.1 Selection of the accuracy parameter $\widehat{r}_k$

Recall that, with the IMSG algorithm, the minimization of the augmented Lagrangian in Step $k.0$ is carried out in an inexact fashion, whereas with the MSG algorithm, the same minimization is performed exactly. In a computational medium, exact minimization does not mean that $\widehat{r}_k \equiv 0$, but rather $\widehat{r}_k \equiv r^*$, where r^* is a fine enough tolerance on the function being minimized, in order to achieve some reasonable accuracy. A threshold value a is also considered such that when $\|f_k\| < a$ an iterate x_k is regarded to be "close enough" to the feasible set.

In the numerical experiments, we consider the following cases for the sequence $\{\widehat{r}_k\}$:

I: $\widehat{r}_k = r^*$, $k = 0, 1, \ldots$

II: $\widehat{r}_k = r_0 > 0$, if $\|f_k\| > a$; $\widehat{r}_k = r^*$, if $\|f_k\| \leq a$.

III: $\widehat{r}_k = \max\{\widehat{r}_{k-1}/2, r^*\}$, if $\|f_k\| > a$; $\widehat{r}_k = r^*$, if $\|f_k\| \leq a$.

IV: $\widehat{r}_k = \max\{\widehat{r}_{k-1}/5, r^*\}$, if $\|f_k\| > a$; $\widehat{r}_k = r^*$, if $\|f_k\| \leq a$.

V: $\widehat{r}_k = \max\{\widehat{r}_{k-1}/10, r^*\}$, if $\|f_k\| > a$; $\widehat{r}_k = r^*$, if $\|f_k\| \leq a$.

In the rules II-V above, $\widehat{r}_0$ is specified and $k = 1, 2, \ldots$. Note that Case I corresponds to using the MSG algorithm (with a fine enough tolerance r^*), while the other cases to the IMSG algorithm. In Case II, $\widehat{r}_k$ is some constant positive value if the primal iterate is not close enough to the feasible set. Near the feasible set (i.e., when $\|f_k\| \leq a$), $\widehat{r}_k = r^*$, i.e. the MSG algorithm is used. In Cases III–V, $\widehat{r}_k$ is reduced in each step, until after either we reach the fine tolerance r^* associated with the exact MSG algorithm or the primal iterate gets close enough to the feasible set, i.e. $\|f_k\| \leq a$.

When $\widehat{r}_k > r^*$ the parameter `tolx` is set to a large value (say 10^5) to facilitate inexactness, because `fminsearch` ensures to meet both of the tolerances `tolfun` and `tolx` at the same time. When $\widehat{r}_k = r^*$ we set `tolx`$= \widehat{r}_k$.

We observe that $\widehat{r}_k =$ `tolfun` and the *a posteriori* computed r_k are, in general, practically close to one another in terms of absolute distance (see Tables 2, 4 and 6).

6.2 Test problems

In the three test problems, we use the step-size rule in (S). We employ the estimates $\widehat{H}_k = \widehat{H}$ for the unknown $\overline{H}$. It should be noted that the numerical implementation of the algorithm terminates in a finite number of iterations (when $\|f_k\| \approx 0$ and $r_k \leq r^*$). In this case, Assumption (A2) is automatically satisfied. For inexactness, we follow the rules given in Cases I-V above for $\widehat{r}_k$.

In the tabulation of the numerical results, the following notation is used.

$t_{\rm CPU}$: CPU time in seconds (in Problems 1 and 2 it is averaged over 100 runs),
n_L: Total number of (Lagrangian) function evaluations,
$n_{\rm IMSG}$: Number of IMSG iterations (i.e. number of serious steps in IMSG).

The CPU time is measured within MATLAB running on (single user) Windows XP Professional (Version 5.1) operating system with a 2.00 GHz Intel Pentium M processor with 1 GB of RAM.

Problem 1 We consider a problem by Murtagh and Saunders [16, 7], which has also been solved using the MSG algorithm in [4].

$$\begin{aligned} \min\ & f_0(x) = (x_1-1)^2 + (x_1-x_2)^2 + (x_2-x_3)^3 + (x_3-x_4)^4 + (x_4-x_5)^4 \\ \text{subject to}\ & f_1(x) = x_1 + x_2^2 + x_3^3 - 3\sqrt{2} - 2 = 0 \\ & f_2(x) = x_2 - x_3^2 + x_4 - 2\sqrt{2} + 2 = 0 \\ & f_3(x) = x_1\, x_5 - 2 = 0 \end{aligned}$$

In this problem, a typesetting error appearing in the first constraint function $f_1(x)$ in [4] has been corrected. The optimal dual value is obtained in [4] as $\overline{H} = 0.02931$. Further details of the solution can be found in [4]. The results for Cases I–V under the step-size rule (S) are depicted in Table 1.

	I	II	III	IV	V
$t_{\rm CPU}$	0.58	0.43	0.39	0.45	0.45
n_L	6400	3631	2990	4065	3873
$n_{\rm IMSG}$	7	7	7	7	7

Table 1: Problem 1 – Performance of the IMSG algorithm under (S), with $c_0 = 1$, $u_0 = (0,\ 1,\ 1)$, $\widehat{H} = 0.1$, $\alpha = 1$ (see (16)), $\delta = 0.1$; $\widehat{r}_0 = 10^{-6}$, $r^* = 10^{-10}$, and $a = 0.9$. ($t_{\rm CPU}$ is averaged over 100 runs.)

In each of Cases II–V (IMSG), clear computational savings are obtained over Case I (MSG), both in terms of CPU time and number of Lagrangian evaluations. It is remarkable that these savings can be achieved with relatively small values of $\widehat{r}_k$. It is worthwhile to note that in the inexact version, the number of IMSG iterations remains the same as those of MSG.

For illustration purposes, for each IMSG iteration under (S), we tabulate in Table 2, $\widehat{r}_k =$ `tolfun` and the lower bound of r_k, which is $(L_k - H_k)$, as well as $(\widehat{H} - L_k)$, in

k	$\widehat{r}_k$	$L_k - H_k$	$\widehat{H} - L_k$
0	1.0×10^{-6}	5.2×10^{-7}	3.6×10^{-1}
1	5.0×10^{-7}	5.4×10^{-7}	2.6×10^{-1}
2	2.5×10^{-7}	1.1×10^{-7}	1.9×10^{-1}
3	1.3×10^{-7}	3.2×10^{-8}	1.3×10^{-1}
4	6.2×10^{-8}	1.3×10^{-7}	9.7×10^{-2}
5	1.0×10^{-10}	1.1×10^{-16}	7.1×10^{-2}
6	1.0×10^{-10}	0	7.1×10^{-2}

Table 2: Problem 1 – A comparison of $\widehat{r}_k = \texttt{tolfun}$ and the lower bound of r_k, $(L_k - H_k)$, in the IMSG iterations under (S) for Case III.

that iteration, for Case III. It can easily be seen from the last column of Table 2 that Condition (18) is satisfied.

We observe that, for all k, r_k is close to $\widehat{r}_k$. In particular, except for $k = 1, 4$, $r_k \le \widehat{r}_k$. In Cases I-II and IV-V, r_k "follows" $\widehat{r}_k$ much more closely.

Problem 2 This Quadratic Integer Programming problem originates from [7], which has also been solved in [4] in the following nonsmooth form.

$$\begin{aligned}
\min\ \ & f_0(x) = a^T x + \frac{1}{2} x^T Q x \\
\text{subject to}\ \ & f_1(x) = \max(0, g_1(x) - 1) = 0 \\
& f_2(x) = \max(0, -(g_1(x) + 1)) = 0 \\
& f_3(x) = \max(0, g_2(x) - 2) = 0 \\
& f_4(x) = \max(0, -(g_2(x) + 3)) = 0 \\
& f_5(x) = \sum_{i=1}^{4} |(x_i - 1)(x_i + 1)| = 0
\end{aligned}$$

where $g_1(x) := x_1 x_2 + x_3 x_4$, $g_2(x) := x_1 + x_2 + x_3 + x_4$, and

$$a^T = [6 \ \ 8 \ \ 4 \ \ -2]\ , \qquad Q = \begin{bmatrix} -1 & 2 & 0 & 0 \\ 2 & -1 & 2 & 0 \\ 0 & 2 & -1 & 2 \\ 0 & 0 & 2 & -1 \end{bmatrix} .$$

In this quadratic integer programming problem, $\overline{H} = -20$. The results for Cases I–V are depicted in Table 3.

	I	II	III	IV	V
t_{CPU}	0.79	0.57	0.37	0.35	0.37
n_L	7112	1506	1255	1217	1469
n_{IMSG}	7	22	9	8	8

Table 3: Problem 2 – Performance of the IMSG algorithm under (S), with $c_0 = 1$, $u_0 = (-1,\ -1,\ -1,\ -1,\ -1)$, $\widehat{H} = -19$, $\alpha = 1$ (see (16)), $\delta = 0.05$; $\widehat{r}_0 = 10^{-1}$, $r^* = 10^{-10}$, and $a = 0.7$. (t_{CPU} is averaged over 100 runs.)

The computational savings in terms of Lagrangian evaluations is much more notable in this example; this is the case even when it takes many more (namely 22) IMSG iterations compared to just seven MSG iterations. The CPU times in Cases III–V are distinctively much less than that for MSG in Case I. One should note that $\widehat{r}_0$ is much larger than r^* in this test problem. In each iteration, it can be seen from the last column of Table 4 that Condition (18) holds.

For illustration purposes, we tabulate $\widehat{r}_k =$ `tolfun` and the lower bound of r_k, $(L_k - H_k)$, as well as $(\widehat{H} - L_k)$, for Case III, in Table 4.

k	$\widehat{r}_k$	$L_k - H_k$	$\widehat{H} - L_k$
0	1.0×10^{-1}	3.4×10^{-1}	1.6
1	5.0×10^{-2}	1.7×10^{-1}	1.6
2	2.5×10^{-2}	1.2×10^{-1}	1.5
3	1.3×10^{-2}	8.9×10^{-3}	1.4
4	6.3×10^{-3}	2.2×10^{-2}	1.2
5	3.1×10^{-3}	1.2×10^{-3}	1.1
6	1.6×10^{-3}	3.3×10^{-3}	1.0
7	1.0×10^{-10}	1.2×10^{-3}	1.0
8	1.0×10^{-10}	4.4×10^{-10}	1.0

Table 4: Problem 2 – A comparison of $\widehat{r}_k =$ `tolfun` and the lower bound of r_k, $(L_k - H_k)$, in the IMSG iterations under (S) for Case III.

We note that r_7 is much larger than $\widehat{r}_7$ in this case. However, because Condition (18) is satisfied, this does not cause any concern. In Cases I-II and IV-V, r_k "follows" $\widehat{r}_k$ much more closely.

Problem 3 This problem concerns finding a time-optimal concatenation of bang–bang arcs, which takes a control system from an initial state to a terminal state (in minimum time) [10, 4].

$$
\begin{aligned}
\min_{\xi} \ & f_0(\xi) = \xi_1 + \xi_2 + \xi_3 + \xi_4 \\
\text{subject to } \ & f_i(\xi) = z_i(\xi_1 + \xi_2 + \xi_3 + \xi_4) = 0 \ , \ i = 1, 2 \ , \\
& f_3(\xi) = \sum_{k=1}^{4} \min\{0, \xi_k\} = 0 \ ,
\end{aligned}
$$

where $z_i(\xi_1+\xi_2+\xi_3+\xi_4)$, $i = 1, 2$, are the solution components of the ordinary differential equations

$$
\begin{aligned}
\dot{z}_1(t) &= z_2(t) \ , \\
\dot{z}_2(t) &= -z_1(t) - (z_1^2(t) - 1)\, z_2(t) + v(t) \ ,
\end{aligned}
$$

with the initial condition $z(0) = (1,\ 1)$, and

$$
v(t) = (-1)^{k+1} \qquad \text{for} \qquad t_{k-1} \le t \le t_k \ ,
$$

$t_0 = 0$, $t_k = \sum_{j=1}^{k} \xi_j$, $k = 1, \ldots, 4$.

Further details of the problem description and the solution using the MSG algorithm can be found in [4]; however we only note here that $\overline{H} = 3.09520$, which represents the minimum time to reach the given target state. For the solution of the ordinary differential equations the MATLAB function m-file `ode45` has been utilized with the relative and absolute tolerances of 10^{-8}. The results for Cases I–V are depicted in Table 5.

	I	II	III	IV	V
$t_{\rm CPU}$	56	37	56	48	40
n_L	1815	912	1834	1498	1207
$n_{\rm IMSG}$	6	6	7	6	6

Table 5: Problem 3 – Performance of the IMSG algorithm under (S), with $c_0 = 2$, $u_0 = (-1,\ -1,\ -5)$, $\widehat{H} = 4$, $\alpha = 1$ (see (16)), $\delta = 0.01$; $\widehat{r}_0 = 10^{-4}$, $r^* = 10^{-7}$, and $a = 0.7$.

Because each Lagrangian evaluation requires the solution of a system of differential equations, the computations are intensive in terms of CPU time. Computational savings are achieved in three of the four cases over Case I, or MSG. It is worthwhile to note that one extra IMSG iteration makes Case III as expensive as Case I. This prompts the importance of a careful choice of the sequence $\{\widehat{r}_k\}$.

For illustration purposes, we tabulate $\widehat{r}_k =$ `tolfun` and the lower bound of r_k, $(L_k - H_k)$, as well as $(\widehat{H} - L_k)$, in that iteration for Case III, in Table 6.

k	$\widehat{r}_k$	$L_k - H_k$	$\widehat{H} - L_k$
0	1.0×10^{-4}	3.8×10^{-2}	9.9×10^{-1}
1	5.0×10^{-5}	1.7×10^{-4}	1.0×10^{0}
2	2.5×10^{-5}	1.7×10^{-4}	9.7×10^{-1}
3	1.3×10^{-5}	1.4×10^{-4}	9.4×10^{-1}
4	1.0×10^{-7}	5.1×10^{-6}	9.1×10^{-1}
5	1.0×10^{-7}	3.3×10^{-6}	9.0×10^{-1}
6	1.0×10^{-7}	1.5×10^{-7}	9.0×10^{-1}

Table 6: Problem 3 – A comparison of $\widehat{r}_k =$ `tolfun` and the lower bound of r_k, $(L_k - H_k)$, in the IMSG iterations under (S) for Case III.

In this case r_k is consistently greater than $\widehat{r}_k$. However, as in the previous two problems, Condition (18) is satisfied.

In Problems 1-3, we employed `fminsearch` which only finds a local minimum. In these problems, we were successful in implementing the IMSG algorithm; however, in principle, a global search is required to find an x_k satisfying (15). Problem 4 in the following section is devised to illustrate an advantage of the IMSG algorithm in doing this global search.

6.3 A simple illustrative example

In this section, we illustrate how savings can be achieved on a simple, nonlinear and nonsmooth system of equations. Simplicity of the problem allows us to discuss the problem graphically, and thus present some of the main arguments of the IMSG algorithm more clearly.

Note that, systems of equations arise from very important applications in many areas (see [25] and the references therein). Iterative methods have been developed for solving nonlinear systems of equations (see [11]). However, the most common and efficient methods for solving these equations involve optimization problems, in particular those in the form of (P) (see [23]).

Each serious step in the MSG algorithm requires a solution of the global optimization problem (5) which is in general quite expensive to obtain (and in some cases unsuccessful). The IMSG algorithm relaxes the problem as in (6) requiring to find an r_k-optimal solution, that is to find a point x_k such that (15) is satisfied. This may entail significant computational savings. In this case, $\overline{H}$ is known: $\overline{H} = 0$. We graph the Lagrangians in each IMSG iteration in order to furnish our discussion.

Problem 4 Consider the following simple, nonlinear and nonsmooth system of equations:

$F_0(x) := x^2 - 1 = 0;$

$f_1(x) := \min\{10(x+1)^2,\, 10(x-1)^2 + 1\} = 0;$

$f_2(x) := x + 1 = 0;$

$x \in X := [-2, 2].$

We formulate a corresponding optimization problem as follows.

$$\begin{aligned} \text{Minimize } \; & f_0(x) = \frac{1}{2}\,[F_0(x)]^2, \\ \text{s.t. } \; & x \in X,\; f_1(x) = 0, \;\; f_2(x) = 0. \end{aligned}$$

The augmented Lagrangian of this problem is:

$L(x, (u_1, u_2, c)) := f_0(x) + c\sqrt{[f_1(x)]^2 + [f_2(x)]^2} - u_1\, f_1(x) - u_2\, f_2(x).$

Obviously $\overline{H} = 0$. Next we will carry out the steps of the IMSG algorithm using the step-size (S) with $\widehat{H}_k = \overline{H} = 0$. Let $k = 0$, $u_0 = (1,\ 1)$, $c_0 = 1$, $r_0 = 0.1$.

Step 0.0. The aim is to r_0-minimize the function $L(x, (u_0, c_0))$. Starting from an initial guess $x = 0$ an r_0-minimization gives a solution $x_0 = 0.6$ with $L(x_0, (u_0, c_0)) = -0.9$, accurate to one digit after the decimal point (see Figure 1). The global minimum value of $L(x, (u_0, c_0))$, i.e. the value of $H(u_0, c_0)$, is -1, with the global minimizer $x = 1.4$. So x_0 is indeed an r_0-minimizer.

So $f_{1,0} = f_1(x_0) = 2.6$, $f_{2,0} = f_2(x_0) = 1.6$ and $\|f_0\| = \sqrt{f_{1,0}^2 + f_{2,0}^2} \approx 3.05$. Because $f_0 \neq 0$, we perform next a serious step of the algorithm.

Step 0.1 Taking $\delta = 1$ in the step-size rule (S),

$$s_0 = \frac{\overline{H} - L_0}{\|f_0\|^2} \approx 0.0966\ .$$

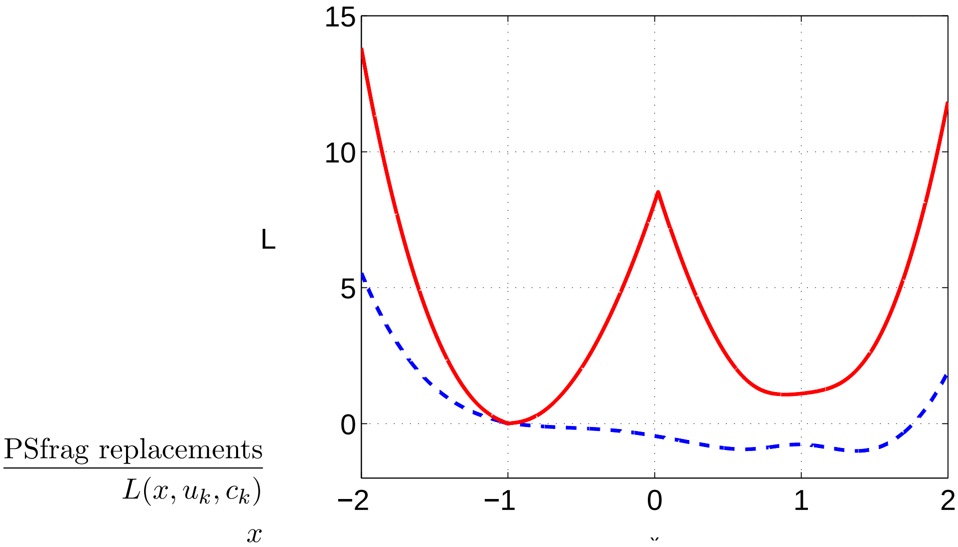

Figure 1: Problem 4 – Graphs of the augmented Lagrangians to minimize in Steps 0.0 and 1.0 of the IMSG algorithm: $L(x, u_0, c_0)$ is given by the dashed curve and $L(x, u_1, c_1)$ by the solid curve.

Note that Condition (17) is satisfied with $\mu = 1/4$:

$$r_0 = 0.1 \leq \mu\,(\overline{H} - L_0) = 0.225\ .$$

Using s_0 and $\varepsilon_0 := s_0$, the Lagrange multipliers are updated as $u_1 \approx (0.75,\ 0.85)$ and $c_1 \approx 1.59$. Next we set $k = 1$, $r_1 = r_0/2$.

Step 1.0. Carry out an r_1-minimization of the function $L(x, (u_1, c_1))$.

The graph of this function is depicted in Figure 1, which indicates that $H(u_1, c_1) = 0$ and that there are two local minimizers, namely $x' = -1$ and $x'' \approx 0.89$. However the condition

$$x \in X \text{ and } L(x, (u_1, c_1)) \leq H(u_1, c_1) + r_1$$

is only satisfied in some small neighbourhood of the local minimizer $x' = -1$. Therefore one needs to employ a global search technique to find an x satisfying the above condition. The condition $L_1 \leq \overline{H} = 0$ in (17) then yields $x_1 = x' = -1$ as the only r_1-minimizer which is also a local minimizer of $L(\cdot, (u_1, c_1))$.

In this example, both the MSG and IMSG algorithms find the solution performing two serious steps which involve two global optimization problems. The advantage of the IMSG algorithm is that, in Step 0.0, it terminates the global search once it satisfies $r_0 \leq \mu\,(\overline{H} - L_0)$; however, the MSG algorithm requires the completion of the global search.

References

[1] M. S. Bazaraa and H. D. Sherali, "On the choice of step sizes in subgradient optimization," *European Journal of Operations Research*, vol. 7, pp. 380–388, 1981.

[2] E. G. Birgin, C. A. Floudas and J. M. Martínez, "Global minimization using an augmented Lagrangian method with variable lower-level constraints", *Preprint*, 2007.

[3] U. Brännlund, "A generalized subgradient method with relaxation step", *Mathematical Programming*, vol. 71, pp. 207–219, 1995.

[4] R. S. Burachik, R. N. Gasimov, N. A. Ismayilova and C. Y. Kaya, "On a modified subgradient algorithm for dual problems via sharp augmented Lagrangian," *Journal of Global Optimization*, vol. 34(1), pp. 55–78, 2006.

[5] R. S. Burachik and C. Y. Kaya, "An update rule and a convergence result for a penalty function method," *Journal of Industrial and Management Optimization*, vol. 3(2), pp. 381-398, 2007.

[6] R. S. Burachik and A. M. Rubinov, "On the absence of duality gap for Lagrange-type functions," *Journal of Industrial Management and Optimization*," vol. 1(1), pp. 33–38, 2005.

[7] A. C. Floudas, P. M. Pardalos, C. S. Adjiman, W. R. Esposito, Z. H. Gümüş, S. T. Harding, J. L. Klepeis, C. A. Meyer and C. A. Schweiger, *Handbook of Test Problems in Local and Global Optimization*, Kluwer Academic Publishers, the Netherlands, 1999.

[8] R. N. Gasimov, "Augmented Lagrangian duality and nondifferentiable optimization methods in nonconvex programming," *Journal of Global Optimization*, vol. 24, pp. 187–203, 2002.

[9] R. N. Gasimov and N. A. Ismayilova, "The modified subgradient method for equality constrained nonconvex optimization problems," Qi, Liqun (ed.) et al., Optimization and Control with Applications, New York, Springer, Applied Optimization, vol. 96, pp. 257–270, 2005.

[10] C. Y. Kaya and J. L. Noakes, "Computational method for time-optimal switching control," *Journal of Optimization Theory and Applications*, vol. 117(1), pp. 69–92, 2003.

[11] C. T. Kelley, *Iterative methods for Linear and Nonlinear Equations*, SIAM, Philadelphia, 1995.

[12] K. Kiwiel, "Convergence of approximate and incremental subgradient methods for convex optimization," *SIAM Journal of Optimization*, vol. 14(3), pp. 807–840, 2004.

[13] J. C. Lagarias, J. A. Reeds, M. H. Wright and P. E. Wright, "Convergence properties of the Nelder-Mead simplex method in low dimensions," *SIAM Journal of Optimization*, vol. 9(1), pp. 112–147, 1998.

[14] K. I. M. McKinnon, "Convergence of the Nelder-Mead simplex method to a nonstationary point," *SIAM Journal of Optimization*, vol. 9(1), pp. 148–158, 1998.

[15] E. Mijangos, "Approximate subgradient methods for nonlinearly constrained network flow problems," *Journal of Optimization Theory and Applications*, vol. 128(1), pp. 167–190, 2006.

[16] B. A. Murtagh and M. A. Saunders, *MINOS 5.4 User's Guide*, Systems Optimization Laboratory, Department of Operations Research, Stanford University, Technical Report SOL 83-20R, 1983.

[17] A. Nedić and D. P. Bertsekas, "Incremental subgradient methods for nondifferentiable optimization", *SIAM Journal of Optimization*, vol. 12(1), pp. 109–138 (electronic), 2001.

[18] J. C. Price, I. D. Coope and D. Byatt, "A convergent variant of the Nelder-Mead Algorithm," *Journal of Optimization Theory and Applications*, vol. 113(1), pp. 5–19, 2002.

[19] B. T. Polyak, "Minimization of unsmooth functionals", *Z. Vychisl. Mat. i Mat. Fiz.*, vol. 9, pp. 509–521, 1969.

[20] R. T. Rockafellar and R. J.-B. Wets, *Variational Analysis*, Springer, Berlin, 1998.

[21] H. D. Sherali, G. Choi and C. H. Tuncbilek, "A variable target value method for nondifferentiable optimization," *Operations Research Letters*, vol. 26, pp 1–8, 2000.

[22] N. Z. Shor, *Minimization Methods for Nondifferentiable Functions*, Springer-Verlag, Berlin, 1985.

[23] X. Tong, L. Qi and Y. Yang, "The Lagrangian globalization method for nonsmooth constrained equations," *Computational Optimization and Applications,* vol 33, pp. 89–109, 2006.

[24] C. Y. Wang, X. Q. Yang and X. M. Yang, "Nonlinear Lagrange duality theorems and penalty function methods in continuous optimization," *Journal of Global Optimization*, vol. 27, pp. 473–484, 2003.

[25] A. J. Wood and B. F. Wollenberg, *Power Generation and Control*, John Wiley and Sons, New York, USA, 1996.