Research Article

An Augmented Lagrangian Algorithm for Solving Semiinfinite Programming

Qian Liu[1] and Changyu Wang[2]

[1] *Department of Mathematics, Shandong Normal University, Jinan, China*
[2] *Institute for Operations Research, Qufu Normal University, Qufu, China*

Correspondence should be addressed to Qian Liu, lq_qsh@163.com

We present a smooth augmented Lagrangian algorithm for semiinfinite programming (SIP). For this algorithm, we establish a perturbation theorem under mild conditions. As a corollary of the perturbation theorem, we obtain the global convergence result, that is, any accumulation point of the sequence generated by the algorithm is the solution of SIP. We get this global convergence result without any boundedness condition or coercive condition. Another corollary of the perturbation theorem shows that the perturbation function at zero point is lower semi-continuous if and only if the algorithm forces the sequence of objective function convergence to the optimal value of SIP. Finally, numerical results are given.

1. Introduction

We consider the semi-infinite programming (SIP):

$$\inf\{f(x) \mid x \in X\}, \tag{1.1}$$

where $X = \{x \in R^n \mid g(x,s) \leq 0, \text{for all } s \in \Omega\}$, the functions $f : R^n \to R$ and $g : R^n \times R^m \to R$ are continuously differentiable. $\Omega \subset R^m$ is a nonempty bounded and closed domain. In this paper, we assume that

$$\inf_{x \in R^n} f(x) > -\infty. \tag{1.2}$$

This assumption is very mild, because the objective function $f(x)$ can be replaced by $e^{f(x)}$ if the assumption is not satisfied.

Semi-infinite programming has wide applications such as engineering technology, optimal control, characteristic value calculation, and statistical design. Many methods have been proposed to solve semi-infinite programming (see [1–4]). As we know, the main difficulty for solving SIP is that it has infinite constraints. If transforming the infinite constraints into an integral function, SIP (1.1) is equivalent to a nonlinear programming with finite constraints.

For any given $x \in R^n$ and $s \in \Omega$, let

$$[g(x,s)]_+ = \max\{g(x,s), 0\}. \tag{1.3}$$

Define $\varphi : R^n \to R$ by

$$\varphi(x) = \int_\Omega [g(x,s)]_+ d\mu(s), \tag{1.4}$$

where $\mu \geq 0$ is a given probability measure on Ω, that is, $\int_\Omega d\mu(s) = \mu(\Omega) = 1$. Thus SIP (1.1) can be reformulated as the following nonlinear programming (NP) with one equality constraint:

$$\inf_{x \in R^n} \{f(x) \mid \varphi(x) = 0\}. \tag{1.5}$$

Then nonlinear programming (1.5) has the same optimal solution and optimal value with SIP (1.1).

For nonlinear programming with finite equality constraints, Hestenes [5] and Powell [6] independently proposed an augmented Lagrangian function by incorporating a quadratic penalty term in the conventional Lagrangian function. This augmented Lagrangian function avoids the shortcoming that the conventional Lagrangian function is only suitable for convex function. So the augmented Lagrangian function can be applied to nonconvex optimization problem. Later, the augmented Lagrangian function was extended to inequality constrained optimization problems and thoroughly investigated by Rockafellar [7]. Recently, Yang and Teo [8] and Rückmann and Shapiro [9] introduced the augmented Lagrangian function for SIP (1.1). In [9], necessary and sufficient conditions for the existence of corresponding augmented Lagrange multipliers were presented. [8] proposed a nonlinear Lagrangian method and established that the sequence of optimal values of nonlinear penalty problems converges to that of SIP (1.1), under the assumption that the level set of objective function is bounded. In this paper, using the equivalent relation of semi-infinite programming (1.1) and nonlinear programming (1.5), without any boundness condition, we present an augmented Lagrangian algorithm for SIP (1.1).

We notice that although the constraints of NP (1.5) are finite, but the constraint function is nonsmooth. Therefore, existing gradient-based optimization methods cannot be used to solve NP (1.5) directly. To overcome this inconvenience, we have to smooth the constraint function. For SIP (1.1), [10–13] presented semismooth Newton methods and smoothing Newton methods. They proved that each accumulation point is a generalized stationary point of SIP (1.1). However, at each iteration of these methods, a Hessian matrix needs to be computed. When the size of the problem is large, computing a Hessian matrix is very expensive. Based on exact l_1 penalty function that is approximated by a family of

smoothing functions, a smoothed-penalty algorithm for solving NP (1.5) was proposed by [14]. They proved that if the constrained set is bounded or the objective function is coercive, the algorithm generates a sequence whose accumulation points are solutions of SIP (1.1).

In this paper, for SIP (1.1), we present a smooth augmented Lagrangian algorithm by smoothing the classical augmented Lagrangian function [7]. In this algorithm, we need not have to get an exact global optimal solution of unconstraint subproblem at each iteration. It is sufficient to search an inexact solution. It is not difficult to obtain an inexact solution, whenever the evaluation of the integral function is not very expensive. For this algorithm, we establish a perturbation theorem under mild conditions. As a corollary of the perturbation theorem, we obtain the global convergence result, that is, any accumulation point of the sequence generated by the algorithm is the solution of SIP (1.1). We get this global convergence result without any boundedness condition or coercive condition. It is noteworthy that the boundedness of the multiplier sequence is a sufficient condition in many literatures about Lagrangian method (see [15–17]). However, in our algorithm, the multiplier sequence can be unbounded. Another corollary of the perturbation theorem shows that the perturbation function at zero point is lower semi-continuous if and only if the algorithm forces the sequence of objective function convergence to the optimal value of SIP (1.1).

The paper is organized as follows. In the next section, we present a smooth augmented Lagrangian algorithm. In Section 3, we establish the perturbation theorem of the algorithm. By this theorem, we obtain a global convergence property and a sufficient and necessary condition in which the algorithm forces the sequence of objective functions convergence to the optimal value of SIP (1.1). Finally, we give some numerical results in Section 4.

2. Smooth Augmented Lagrangian Algorithm

Before we introduce the algorithm, some definitions and symbols need to be given. For $\varepsilon \geq 0$, we define the relaxed feasible set of SIP (1.1) as follows:

$$R_\varepsilon = \left\{ x \in R^n \mid \int_\Omega \left[g(x, s) \right]_+ d\mu(s) \leq \varepsilon \right\}. \tag{2.1}$$

Then R_0 is the feasible set of SIP (1.1). Let R_0^* be the set of optimal solutions of SIP (1.1). We assume that $R_0^* \neq \emptyset$ in this paper.

The perturbation function is defined as follows:

$$\theta_f(\varepsilon) = \inf_{x \in R_\varepsilon} f(x). \tag{2.2}$$

Thus the optimal value of SIP (1.1) is

$$\theta_f(0) = \inf_{x \in R_0} f(x). \tag{2.3}$$

It is easy to show that $\theta_f(\cdot)$ is upper semi-continuous at the point $\varepsilon = 0$.

For problem (1.5), the corresponding classical augmented Lagrangian function [7] is

$$L(x,\lambda,\rho) = f(x) + \frac{\rho}{2}\left[\varphi(x) + \frac{\lambda}{\rho}\right]^2 - \frac{\lambda^2}{2\rho}, \tag{2.4}$$

where λ is the Lagrangian multiplier and ρ is the penalty parameter. On base of it, we introduce a class of smooth augmented Lagrangian function:

$$f_r(x,\lambda,\rho) = f(x) + \frac{\rho}{2}\left[r\int_\Omega \phi\left(\frac{g(x,s)}{r}\right)d\mu(s) + \frac{\lambda}{\rho}\right]^2 - \frac{\lambda^2}{2\rho}. \tag{2.5}$$

Here r is the approximate parameter.

In the following, we suppose that the continuously differentiable function $\phi : R \to R$ satisfies

   (a) $\phi(\cdot)$ is nonnegative and monotone increasing;
   (b) for any $t > 0$, $\phi(t) \geq t$;
   (c) $\lim_{t\to+\infty}(\phi(t)/t) = 1$.

It is easy to check that there are many continuously differentiable functions satisfying conditions (a), (b), and (c). For example,

$$\phi_1(t) = \log(1 + e^t).$$

$$\phi_2(t) = \begin{cases} 2e^t, & t < 0, \\ t + \log(1 + t) + 2, & t \geq 0. \end{cases}$$

$$\phi_3(t) = \begin{cases} e^t, & t < 0, \\ t + 1, & t \geq 0. \end{cases} \tag{2.6}$$

$$\phi_4(t) = \frac{1}{2}\left(t + \sqrt{t^2 + 4}\right).$$

Using conditions (a) and (c), for any $t \in R$, we have

$$\lim_{r\to 0^+} r\phi\left(\frac{t}{r}\right) = \max\{0,t\}. \tag{2.7}$$

From the above equation, under conditions (a)–(c), the smooth function $f_r(x,\lambda,\rho)$ approximates to the classical augmented Lagrangian function $L(x,\lambda,\rho)$ as r approaches to zero, that is,

$$\lim_{r\to 0^+} f_r(x,\lambda,\rho) = L(x,\lambda,\rho) = f(x) + \frac{\rho}{2}\left[\varphi(x) + \frac{\lambda}{\rho}\right]^2 - \frac{\lambda^2}{2\rho}. \tag{2.8}$$

Based on the smooth augmented Lagrangian function $f_\rho(x,\beta,r)$, we present the following smooth augmented Lagrangian algorithm.

Algorithm 2.1. Set $x_0 \in R^n$, $r_0 > 0$, $\varepsilon_0 > 0$, $\rho_0 > 0$, $\lambda_0 \in (0, \min\{1/\sqrt{r_0}, 1/\sqrt{\varepsilon_0}\}]$, $k := 0$.
Step 1. Compute

$$x_k \in \arg\min_{x \in R^n} f_{r_k}(x, \lambda_k, \rho_k). \tag{2.9}$$

Otherwise, seek on inexact global optimal solution x_k satisfying

$$f_{r_k}(x_k, \lambda_k, \rho_k) \leq \inf_{x \in R^n} f_{r_k}(x, \lambda_k, \rho_k) + \varepsilon_k. \tag{2.10}$$

Step 2. Set $r_{k+1} = (1/2)r_k$, $\varepsilon_{k+1} = (1/2)\varepsilon_k$,

$$\rho_{k+1} = \begin{cases} \rho_k, & x_k \in R_{\varepsilon_k}; \\ 2\rho_k, & x_k \notin R_{\varepsilon_k}, \end{cases}$$

$$\lambda_{k+1} \in \left(0, \min\left\{\frac{1}{\sqrt{r_{k+1}}}, \frac{1}{\sqrt{\varepsilon_{k+1}}}\right\}\right]. \tag{2.11}$$

Step 3. Set $k := k + 1$, go back step 1.

Since $f(x)$ is bounded below and $\phi(\cdot)$ is nonnegative, an inexact solution satisfying (2.10) always exists. Thus Algorithm 2.1 is feasible.

3. Convergence Properties

In this section, by using a perturbation theorem of Algorithm 2.1, we will obtain a global convergence property, a sufficient and necessary condition that Algorithm 2.1 forces the sequence of objective functions convergence to the optimal value of SIP (1.1). To prove the perturbation theorem, we first give the following two lemmas.

Let $\Omega_+(x) = \{s \in \Omega \mid g(x, s) > 0\}$, $\Omega_0(x) = \{s \in \Omega \mid g(x, s) = 0\}$, $\Omega_-(x) = \{s \in \Omega \mid g(x, s) < 0\}$.

Lemma 3.1. *Suppose that the point sequence $\{x_k\}$ is generated by Algorithm 2.1. Then for any $\varepsilon > 0$, there exists a positive integer k_0 such that $x_k \in R_\varepsilon$, for all $k > k_0$.*

Proof.

Case 1. When $k \to +\infty$, ρ_k tends to a finite number. From Algorithm 2.1, there exists a positive integer N_1 such that $x_k \in R_{\varepsilon_k}$ for all $k > N_1$. Notice that $\varepsilon_k \to 0$, so for any $\varepsilon > 0$, there exists a positive integer N_2 such that $\varepsilon_k < \varepsilon$ for all $k > N_2$. Therefore, when $k > \max\{N_1, N_2\}$, we have $x_k \in R_{\varepsilon_k} \subseteq R_\varepsilon$.

Case 2. When $k \to +\infty$, $\rho_k \to +\infty$. We suppose that the conclusion does not hold. Then for $\bar{\varepsilon} > 0$, there exists an infinite subsequence $K \subseteq N = \{1, 2, 3, \ldots\}$ such that $x_k \notin R_{\bar{\varepsilon}}$ for all $k \in K$, that is,

$$\int_\Omega [g(x_k, s)]_+ d\mu(s) > \bar{\varepsilon}. \tag{3.1}$$

Since $\varepsilon_k \to 0$, for the above $\bar{\varepsilon} > 0$, there exists a positive integer N_3 such that $\varepsilon_k < \bar{\varepsilon}$ for all $k > N_3$. Then using (2.10) in Algorithm 2.1, we have

$$f_{r_k}(x_k, \lambda_k, \rho_k) \leq \inf_{x \in R^n} f_{r_k}(x, \lambda_k, \rho_k) + \bar{\varepsilon}. \tag{3.2}$$

Therefore by (3.1), (3.2), and $\phi(t)$ satisfying (a)-(b), for any $k \in K$, $k > N_3$, we derive that

$$\inf_{x \in R^n} f_{r_k}(x, \lambda_k, \rho_k) + \bar{\varepsilon} \geq f_{r_k}(x_k, \lambda_k, \rho_k)$$

$$= f(x_k) + \frac{\rho_k}{2}\left[r_k \int_\Omega \phi\left(\frac{g(x_k,s)}{r_k}\right)d\mu(s) + \frac{\lambda_k}{\rho_k}\right]^2 - \frac{\lambda_k^2}{2\rho_k}$$

$$\geq f(x_k) + \frac{\rho_k}{2}\left[r_k \int_{\Omega_+(x_k)} \phi\left(\frac{g(x_k,s)}{r_k}\right)d\mu(s) + \frac{\lambda_k}{\rho_k}\right]^2 - \frac{\lambda_k^2}{2\rho_k}$$

$$\geq f(x_k) + \frac{\rho_k}{2}\left[\int_{\Omega_+(x_k)} g(x_k,s)d\mu(s) + \frac{\lambda_k}{\rho_k}\right]^2 - \frac{\lambda_k^2}{2\rho_k} \tag{3.3}$$

$$\geq f(x_k) + \frac{\rho_k}{2}\left[\bar{\varepsilon} + \frac{\lambda_k}{\rho_k}\right]^2 - \frac{\lambda_k^2}{2\rho_k}$$

$$\geq f(x_k) + \frac{\rho_k}{2}\bar{\varepsilon}^2.$$

Note that $\{f(x_k)\}$ is bounded below and $\rho_k \to +\infty (k \to +\infty)$, then we can obtain that

$$f(x_k) + \frac{\rho_k}{2}\bar{\varepsilon}^2 \longrightarrow +\infty, \quad (k \in K, k \longrightarrow +\infty), \tag{3.4}$$

that is, $\inf_{x \in R^n} f_{r_k}(x, \lambda_k, \rho_k) \to +\infty (k \in K, k \to +\infty)$. However, on the other hand, since $R_0 \neq \emptyset$, we can choose $\bar{x} \in R_0$; by the choice of r_k, ρ_k, λ_k in Algorithm 2.1 and the properties of ϕ, we obtain that

$$\inf_{x \in R^n} f_{r_k}(x, \lambda_k, \rho_k) \leq f_{r_k}(\bar{x}, \lambda_k, \rho_k)$$

$$= f(\bar{x}) + \frac{\rho_k}{2}\left[r_k \int_\Omega \phi\left(\frac{g(\bar{x},s)}{r_k}\right)d\mu(s) + \frac{\lambda_k}{\rho_k}\right]^2 - \frac{\lambda_k^2}{2\rho_k}$$

$$\leq f(\bar{x}) + \frac{\rho_k}{2}\left[r_k \int_\Omega \phi(0)d\mu(s) + \frac{\lambda_k}{\rho_k}\right]^2 - \frac{\lambda_k^2}{2\rho_k} \tag{3.5}$$

$$= f(\bar{x}) + \frac{\rho_k}{2}r_k^2(\phi(0)\mu(\Omega))^2 + \lambda_k r_k \phi(0)\mu(\Omega)$$

$$\leq f(\bar{x}) + \rho_0 r_0^2(\phi(0)\mu(\Omega))^2 + \sqrt{r_0}\phi(0)\mu(\Omega).$$

This indicates that $\inf_{x \in R^n} f_{r_k}(x, \lambda_k, \rho_k)$ has an upper bound. It is in contradiction with $\inf_{x \in R^n} f_{r_k}(x, \lambda_k, \rho_k) \to +\infty (k \in K, k \to +\infty)$. $\qquad\square$

By using Lemma 3.1, we have the following Lemma 3.2.

Lemma 3.2. *Suppose that the point sequence $\{x_k\}$ is generated by Algorithm 2.1. Then for every accumulation point x^* of $\{x_k\}$, one has $x^* \in R_0$.*

Theorem 3.3. *Suppose that the sequence $\{x_k\}$ is generated by Algorithm 2.1, then*

(i) $\lim_{k \to \infty} f(x_k) = \lim_{\varepsilon \to 0^+} \theta_f(\varepsilon)$;

(ii) $\lim_{k \to \infty} f_{r_k}(x_k, \lambda_k, \rho_k) = \lim_{\varepsilon \to 0^+} \theta_f(\varepsilon)$;

(iii) $\lim_{k \to \infty} (\rho_k/2)[r_k \int_\Omega \phi(g(x_k, s)/r_k) d\mu(s) + \lambda_k/\rho_k]^2 - \lambda_k^2/2\rho_k = 0$.

Proof. Since $\theta_f(\varepsilon)$ is monotonically decreasing with respect to $\varepsilon > 0$ and has below bound, we know $\lim_{\varepsilon \to 0^+} \theta_f(\varepsilon)$ exists and is finite. By Algorithm 2.1, we have $\varepsilon_k \downarrow 0$. Then

$$\lim_{k \to +\infty} \theta_f(\varepsilon_k) = \lim_{\varepsilon \to 0^+} \theta_f(\varepsilon). \tag{3.6}$$

Taking $\delta_k > 0$ and $\delta_k \to 0 (k \to +\infty)$, by the definition of infimum, there exists $z_k \in R_{\varepsilon_k}$ such that

$$f(z_k) \leq \theta_f(\varepsilon_k) + \delta_k. \tag{3.7}$$

Since $z_k \in R_{\varepsilon_k}$, that is, $\int_\Omega [g(z_k, s)]_+ d\mu(s) \leq \varepsilon_k$.
    On the other hand, by Lemma 3.1, for any $\varepsilon > 0$, when k is sufficiently large, we have

$$x_k \in R_\varepsilon. \tag{3.8}$$

Since $\phi(t)$ satisfies conditions (a) and (c), we obtain

$$\lim_{t \to +\infty} \frac{\phi(t) - \phi(0)}{t} = \lim_{t \to +\infty} \frac{\phi(t)}{t} = 1, \qquad \lim_{t \to 0^+} \frac{\phi(t) - \phi(0)}{t} = \lim_{t \to 0^+} \phi'(t) = \phi'(0). \tag{3.9}$$

Therefore, there exists $M > 0$ such that

$$\phi(t) \leq Mt + \phi(0), \tag{3.10}$$

for any $t \geq 0$. As stated previously, by the choice of r_k, ε_k, ρ_k, and λ_k in Algorithm 2.1, (3.7), (3.8), and (3.10) derive that for any $\varepsilon > 0$,

$$
\theta_f(\varepsilon) \leq f(x_k)
$$

$$
\leq f(x_k) + \frac{\rho_k}{2}\left[r_k \int_\Omega \phi\left(\frac{g(x_k,s)}{r_k}\right)d\mu(s) + \frac{\lambda_k}{\rho_k}\right]^2 - \frac{\lambda_k^2}{2\rho_k}
$$

$$
\leq \inf_{x \in R^n} f_{r_k}(x, \lambda_k, \rho_k) + \varepsilon_k
$$

$$
\leq f(z_k) + \frac{\rho_k}{2}\left[r_k \int_\Omega \phi\left(\frac{g(z_k,s)}{r_k}\right)d\mu(s) + \frac{\lambda_k}{\rho_k}\right]^2 - \frac{\lambda_k^2}{2\rho_k} + \varepsilon_k
$$

$$
\leq \theta_f(\varepsilon_k) + \delta_k + \frac{\rho_k}{2}\left[r_k \int_\Omega \phi\left(\frac{g(z_k,s)}{r_k}\right)d\mu(s) + \frac{\lambda_k}{\rho_k}\right]^2 - \frac{\lambda_k^2}{2\rho_k} + \varepsilon_k \qquad (3.11)
$$

$$
\leq \theta_f(\varepsilon_k) + \delta_k + \frac{\rho_k}{2}\left[M\varepsilon_k + r_k\phi(0)\mu(\Omega) + \frac{\lambda_k}{\rho_k}\right]^2 - \frac{\lambda_k^2}{2\rho_k} + \varepsilon_k
$$

$$
\leq \theta_f(\varepsilon_k) + \delta_k + \frac{\rho_k}{2}\left(\max\{r_k,\varepsilon_k\}(M + \phi(0)\mu(\Omega))\right)^2
$$

$$
+ \lambda_k \max\{r_k,\varepsilon_k\}(M + \phi(0)\mu(\Omega)) + \varepsilon_k
$$

$$
\leq \theta_f(\varepsilon_k) + \delta_k + \frac{\rho_0}{2}\max\{r_0,\varepsilon_0\}(M + \phi(0)\mu(\Omega))^2 \max\{r_k,\varepsilon_k\}
$$

$$
+ (M + \phi(0)\mu(\Omega))\max\{\sqrt{r_k},\sqrt{\varepsilon_k}\} + \varepsilon_k.
$$

From the above inequalities and (3.6), noticing that $r_k\varepsilon_k \to 0(k \to \infty)$, for any $\varepsilon > 0$, we have

$$
\theta_f(\varepsilon) \leq \liminf_{k \to +\infty} f(x_k)
$$

$$
\leq \limsup_{k \to +\infty} f(x_k)
$$

$$
\leq \limsup_{k \to +\infty}\left\{f(x_k) + \frac{\rho_k}{2}\left[r_k \int_\Omega \phi\left(\frac{g(x_k,s)}{r_k}\right)d\mu(s) + \frac{\lambda_k}{\rho_k}\right]^2 - \frac{\lambda_k^2}{2\rho_k}\right\} \qquad (3.12)
$$

$$
\leq \lim_{k \to +\infty} \theta_f(\varepsilon_k)
$$

$$
= \lim_{\varepsilon \to 0^+} \theta_f(\varepsilon).
$$

Then $\lim_{\varepsilon \to 0^+}\theta_f(\varepsilon) = \lim_{k \to +\infty}f(x_k) = \limsup_{k \to +\infty}\{f(x_k) + \frac{\rho_k}{2}[r_k \int_\Omega \phi(g(x_k,s)/r_k)d\mu(s) + \lambda_k/\rho_k]^2 - \lambda_k^2/2\rho_k\}$. So the conclusions (i)–(iii) hold. $\qquad \square$

Now, we prove the global convergence of Algorithm 2.1.

Corollary 3.4. *Suppose that the point sequence $\{x_k\}$ is generated by Algorithm 2.1. Then every accumulation point of $\{x_k\}$ is the optimal solution of the problem* (1.1).

Proof. Let x^* be an accumulation point of $\{x_k\}$; from Lemma 3.2, we have

$$x^* \in R_0. \tag{3.13}$$

By the conclusion (i) of Theorem 3.3 and (3.13), we obtain

$$\lim_{\varepsilon \to 0^+} \theta_f(\varepsilon) = \lim_{k \to \infty} f(x_k) = f(x^*) \geq \theta_f(0). \tag{3.14}$$

Then we get $f(x^*) = \theta_f(0)$, because (3.14) and $\theta_f(\varepsilon)$ are upper semi-continuous at the point $\varepsilon = 0$. $\qquad\square$

By using Theorem 3.3, we have the following Corollary 3.5.

Corollary 3.5. $\lim_{k \to \infty} f(x_k) = \theta_f(0)$ *if and only if $\theta_f(\varepsilon)$ is lower semi-continuous at the point* $\varepsilon = 0$.

4. Numerical Results

To give some insight into the behavior of the algorithm presented in this paper. It is implemented in Matlab 7.0.4 and runs are made on AMD Athlon(tm) 64 × 2 Dual Core Processor 4800+ with CPU 2.50 GHz and 1.87 GB memory. Tables 1 and 2 show the computational results of the corresponding problems with the following items:

$\qquad k$: number of iterations;

$\qquad x_0$: starting point;

$\qquad \phi(t)$: smoothing function;

$\qquad x_k$: the final iteration point;

$\qquad \lambda_k$: the final Lagrangian multiplier;

$\qquad f(x_k)$: the function value of $f(x)$ at the final x_k.

The parameters used in the Algorithm 2.1 are specified as follows:

$$r_0 = 1, \quad \varepsilon_0 = 1, \quad \rho_0 = 1, \quad \lambda_0 = 1,$$

$$\lambda_{k+1} = \min\left\{\lambda_k + \rho_k r_k \int_\Omega \phi\left(\frac{g(x_k, s)}{r_k}\right) d\mu(s), 10^3\right\}. \tag{4.1}$$

Example 4.1 (see [18]). Consider the following:

$$\min \quad 1.21 \exp(x_1) + \exp(x_2)$$

$$\text{s.t.} \quad s - \exp(x_1 + x_2) \leq 0, \quad \forall s \in [0, 1]. \tag{4.2}$$

Table 1: Numerical results of Example 4.1.

k	$\phi(t)$	x_k	λ_k	$f(x_k)$
26	$\log(1 + e^t)$	$(-0.0968, 0.0938)$	367.2453	2.1982
30	$(1/2)(t + \sqrt{t^2 + 4})$	$(-0.0959, 0.0947)$	923.9.40	2.1993
19	e^t	$(-0.0953, 0.0953)$	6.1129	2.1999

Table 2: Numerical results of Example 4.2.

n	k	$\phi(t)$	x_k	λ_k	$f(x_k)$
3	17	$\log(1 + e^t)$	$(0.0839, 0.4494, 1.0105)$	18.5016	0.6472
3	16	$(1/2)(t + \sqrt{t^2 + 4})$	$(0.0839, 0.4494, 1.0108)$	18.5869	0.6472
3	16	e^t	$(0.0873, 0.4248, 1.0460)$	12.3942	0.6484
6	16	$\log(1 + e^t)$	$(0.00, 1.03, -0.25, 1.24, -1.39, 0.94)$	5.3234	0.6161
6	17	$(1/2)(t + \sqrt{t^2 + 4})$	$(-0.00, 1.03, -0.25, 1.23, -1.39, 0.94)$	7.5705	0.6161
6	16	e^t	$(-0.00, 1.02, -0.25, 1.24, -1.41, 0.95)$	4.1742	0.6161
8	17	$\log(1 + e^t)$	$(-0.00, 1.00, -0.06, 0.77, -1.49, 2.77, -2.41, 0.96)$	4.3033	0.6157
8	18	$(1/2)(t + \sqrt{t^2 + 4})$	$(-0.00, 1.00, -0.06, 0.77, -1.47, 2.78, -2.40, 0.96)$	7.0902	0.6157
8	16	e^t	$(0.00, 1.00, -0.06, 0.78, -1.49, 2.78, -2.42, 0.97)$	3.3116	0.6158

We choose the starting point $x_0 = (0,0)$. This example has the optimal solution $x^* = (-\ln 1.1, \ln 1.1)$.

Example 4.2 (see [18]). Consider the following:

$$\min \quad \sum_{i=1}^{n} x_i / i$$

$$\text{s.t.} \quad \tan s - \sum_{i=1}^{n} x_i s^{i-1} \leq 0, \quad \forall s \in [0,1], \tag{4.3}$$

for $n = 3$, 6, and 8. We choose zero vectors as the starting points.

Throughout the computational experiments, we use trust region method for solving an unconstrained optimization subproblem at each step. For the corresponding trust region subproblem, we directly use the trust function in Matlab toolbox. The test results of Example 4.1 are summarized in Table 1. We test the three cases for $\phi(t) = \log(1 + e^t)$, $\phi(t) = (1/2)(t + \sqrt{t^2 + 4})$ and $\phi(t) = e^t$, which are, respectively, used as the smoothing approximation functions. k denotes the number of the iteration, λ_k denotes the approximate Lagrangian multiplier at the final iteration, and x_k and $f(x_k)$ are the approximate solution and the objective function at the final iteration. For Example 4.2, we test the results when $n = 3$, $n = 6$, and $n = 8$ in Table 2. Numerical results demonstrate that augmented Lagrangian

algorithm established in this paper is a practical and effective method for solving semi-infinite programming problem.

References

[1] B. Bhattacharjee, P. Lemonidis, W. H. Green Jr., and P. I. Barton, "Global solution of semi-infinite programs," *Mathematical Programming*, vol. 103, no. 2, pp. 283–307, 2005.

[2] R. Hettich and K. O. Kortanek, "Semi-infinite programming: theory, methods, and applications," *SIAM Review*, vol. 35, no. 3, pp. 380–429, 1993.

[3] G. Still, "Discretization in semi-infinite programming: the rate of convergence," *Mathematical Programming*, vol. 91, no. 1, pp. 53–69, 2001.

[4] Y. Tanaka, M. Fukushima, and T. Ibaraki, "A globally convergent SQP method for semi-infinite nonlinear optimization," *Journal of Computational and Applied Mathematics*, vol. 23, no. 2, pp. 141–153, 1988.

[5] M. R. Hestenes, "Multiplier and gradient methods," *Journal of Optimization Theory and Applications*, vol. 4, pp. 303–320, 1969.

[6] M. J. D. Powell, "A method for nonlinear constraints in minimization problems," in *Optimization*, R. Fletcher, Ed., pp. 283–298, Academic Press, London, UK, 1969.

[7] R. T. Rockafellar, "Augmented Lagrange multiplier functions and duality in nonconvex programming," *SIAM Journal on Control and Optimization*, vol. 12, pp. 268–285, 1974.

[8] X. Q. Yang and K. L. Teo, "Nonlinear Lagrangian functions and applications to semi-infinite programs," *Annals of Operations Research*, vol. 103, pp. 235–250, 2001.

[9] J.-J. Rückmann and A. Shapiro, "Augmented Lagrangians in semi-infinite programming," *Mathematical Programming*, vol. 116, no. 1-2, pp. 499–512, 2009.

[10] D.-H. Li, L. Qi, J. Tam, and S.-Y. Wu, "A smoothing Newton method for semi-infinite programming," *Journal of Global Optimization*, vol. 30, no. 2-3, pp. 169–194, 2004.

[11] L. Qi, C. Ling, X. Tong, and G. Zhou, "A smoothing projected Newton-type algorithm for semi-infinite programming," *Computational Optimization and Applications*, vol. 42, no. 1, pp. 1–30, 2009.

[12] L. Qi, A. Shapiro, and C. Ling, "Differentiability and semismoothness properties of integral functions and their applications," *Mathematical Programming*, vol. 102, no. 2, pp. 223–248, 2005.

[13] L. Qi, S.-Y. Wu, and G. Zhou, "Semismooth Newton methods for solving semi-infinite programming problems," *Journal of Global Optimization*, vol. 27, no. 2-3, pp. 215–232, 2003.

[14] M. Gugat and M. Herty, "The smoothed-penalty algorithm for state constrained optimal control problems for partial differential equations," *Optimization Methods & Software*, vol. 25, no. 4–6, pp. 573–599, 2010.

[15] E. G. Birgin, R. A. Castillo, and J. M. Martínez, "Numerical comparison of augmented Lagrangian algorithms for nonconvex problems," *Computational Optimization and Applications*, vol. 31, no. 1, pp. 31–55, 2005.

[16] A. R. Conn, N. I. M. Gould, and P. L. Toint, "A globally convergent augmented Lagrangian algorithm for optimization with general constraints and simple bounds," *SIAM Journal on Numerical Analysis*, vol. 28, no. 2, pp. 545–572, 1991.

[17] P. Tseng and D. P. Bertsekas, "On the convergence of the exponential multiplier method for convex programming," *Mathematical Programming*, vol. 60, no. 1, pp. 1–19, 1993.

[18] A. R. Conn and N. I. M. Gould, "An exact penalty function for semi-infinite programming," *Mathematical Programming*, vol. 37, no. 1, pp. 19–40, 1987.