

A NUMERICAL APPROACH TO INFINITE-DIMENSIONAL LINEAR PROGRAMMING IN L_1 SPACES

SATOSHI ITO

Department of Mathematical Analysis and Statistical Inference
Institute of Statistical Mathematics
Research Organization of Information and Systems, Tokyo 106-8569, Japan

SOON-YI WU

Department of Mathematics, National Cheng Kung University, Tainan, Taiwan
National Center for Theoretical Science, Tainan, Taiwan

TING-JANG SHIU

Department of Mathematics, National Cheng Kung University, Tainan, Taiwan

KOK LAY TEO

Department of Mathematics and Statistics, Curtin University of Technology, Perth
Western Australia, Australia

(Communicated by Song Wang)

ABSTRACT. An infinite-dimensional linear programming formulated on L_1 spaces, problem (P), is studied in this paper. A related optimization problem, general capacity problem (GCAP), is also mentioned in this paper. But we find that the optimal solution does not exist in problem (P). Thus, we approach the optimal value for problem (P) via solving the problem (GCAP). A proposed algorithm is shown that we solve a sequence of semi-infinite subproblems to approach the optimal value of problem (P). The error bound for the difference between the optimal value for problem (P) and optimal value for semi-infinite subproblem is also given in this paper. Finally, numerical examples are implemented and compared with discretization method to show our computational efficiency.

1. Introduction. Let X and Y be compact subsets of some Euclidean spaces, and let $C(X)$ and $C(Y)$ denote the spaces of continuous real-valued functions on X and Y , respectively. Let $L_1(X)$ denote the space of Lebesgue integrable functions on X . We consider the following infinite-dimensional linear programming problem:

$$\begin{aligned} & \inf_{h \in L_1(X)} \int_X f(x) h(x) dx \\ & \text{subject to } \int_X \varphi(x, y) h(x) dx \geq g(y), \quad \forall y \in Y, \\ & \quad \quad \quad h(x) \geq 0 \quad \text{a.e. on } X, \end{aligned} \tag{P}$$

2000 *Mathematics Subject Classification.* 90C05; 90C34; 90C59; 65K05.

Key words and phrases. Infinite-dimensional linear programming, semi-infinite programming, cutting plane method, numerical approach.

The first author was partially supported by JSPS Grant-in-Aid for Scientific Research (C) #12680459 and #16540131.

where $f \in C(X)$, $g \in C(Y)$, $\varphi \in C(X \times Y)$ are given functions. Let $L_1^+(X) \subseteq L_1(X)$ be the set consisting all nonnegative functions almost everywhere on X . $L_1^+(X)$ is the closed convex cone of $L_1(X)$. The second constraint of problem (P) is sometimes written as $h \geq 0$ or $h \in L_1^+(X)$.

A related optimization problem called a general capacity problem is formulated as follows.

$$\begin{aligned} & \min_{\mu \in M(X)} \int_X f(x) d\mu(x) \\ & \text{subject to} \quad \int_X \varphi(x, y) d\mu(x) \geq g(y), \quad \forall y \in Y, \\ & \quad \quad \quad \mu \geq 0, \end{aligned} \tag{GCAP}$$

where $M(X)$ denotes the space of signed regular Borel measures on X , and $M^+(X)$ denotes the closed convex cone in $M(X)$ consisting of nonnegative measures.

A dual form of (GCAP) is given by

$$\begin{aligned} & \max_{\nu \in M(Y)} \int_Y g(y) d\nu(y) \\ & \text{subject to} \quad \int_Y \varphi(x, y) d\nu(y) \leq f(x), \quad \forall x \in X, \\ & \quad \quad \quad \nu \geq 0. \end{aligned} \tag{DGCAP}$$

The capacity problem is the origin in electrostatics which studies in determining the capacity of a conducting body. Anderson and Nash [1] (see also [2]) pointed out the electrostatics capacity problem is related with potential theory. Some early studies on the capacity problem can consult Choquet [3] and Fugled [8]. Yamasaki [22] and Ohtsuka [15] recognized the capacity problem as a general linear program problem. A related duality theory and characterization of the extreme points of the feasible domain and the optimal solution can be found in Lai and Wu [13] and references therein. Gabriel and Hernandez-Lerma [9] discussed the strong duality of general capacity problem in metric spaces. In Lai and Wu [13] and Wu, etc [19], authors developed some methods to solve the general capacity problem. We extend problem (GCAP) to problem (P) as considering in L_1 spaces. Some optimal control problems can, in fact, be viewed as infinite-dimensional programming. See, for example, Teo, Goh and Wong [16], Finlay, Gaitsgory and Lebedev [7], Wu and Teo [18], Gerdtts and Kunkel [10], Gong and Xiang [11], Dahleh and Pearson [4] and Dahleh and Diaz-Bobillo [5]. Recently Vanderbei [17] investigated an optimization problem for the best high-contrast apodization, this is an infinite-dimensional linear programming problem in which the decision variable has a lower bound and an upper bound.

In the next section, we will give a numerical approach to get the approximate optimal solution for problem (P). Our approach is based on the method, which was discussed in [13], to solve problem (GCAP). We find the optimal value for problem (GCAP) to approximate the optimal value for problem (P). Now, we assume that the following condition is satisfied throughout the paper.

Assumption 1. Problem (P) is bounded from below, and there exists a $h_0 \in L_1^+(X)$ such that

$$\int_X \varphi(x, y) h_0(x) dx > g(y), \quad \forall y \in Y.$$

Under Assumption 1, we can construct μ_0 be a signed regular Borel measure on X which satisfies $\int h_0(x) dx = \int d\mu_0$. Theorem 2.1 in [13] provided that problems (GCAP) and (DGCAP) are both feasible, and the strong duality holds for this primal-dual pair, i.e., $V(\text{GCAP}) = V(\text{DGCAP})$, where $V(\text{GCAP})$ and $V(\text{DGCAP})$ denote the optimal objective values of problems (GCAP) and (DGCAP).

2. Cutting plane algorithm. Since the optimal solution for the problem (P) does not exist, we will approximate the optimal value of problem (P) via finding the optimal value of problem (GCAP). Let us apply the cutting plane strategy for solving problem (GCAP). At each cutting plane iterate, when the index space Y is relaxed to its finite subset $Y_k := \{y_1^k, y_2^k, \dots, y_{n_k}^k\} \subset Y$, the resulting problem gives a semi-infinite linear programming problem:

$$\begin{aligned} & \min_{\mu \in M(X)} \int_X f(x) d\mu(x) \\ & \text{subject to} \quad \int_X \varphi(x, y) d\mu(x) \geq g(y), \quad \forall y \in Y_k, \\ & \quad \quad \quad \mu \geq 0. \end{aligned} \quad (\text{SIP}(Y_k)):$$

From the theory of semi-infinite programming [12], there exists a discrete measure μ^k concentrated on $x_1^k, x_2^k, \dots, x_{m_k}^k$, where $m_k \leq n_k$, such that μ^k is optimal for problem (SIP(Y_k)). Note that a dual form of problem (SIP(Y_k)) is given by

$$\begin{aligned} & \max_{\nu \in R^{n_k}} \sum_{i=1}^{n_k} g(y_i^k) \nu_i \\ & \text{subject to} \quad \sum_{i=1}^{n_k} \varphi(x, y_i^k) \nu_i \leq f(x), \quad \forall x \in X, \\ & \quad \quad \quad \nu \geq 0, \end{aligned} \quad (\text{DSIP}(Y_k)):$$

where the strong duality holds for this primal-dual pair of semi-infinite programming under Assumption 1.

Now the basic cutting plane algorithm is described as follows.

Algorithm 2.1. Basic cutting plane algorithm.

Step 1: Let $\varepsilon > 0$ be a sufficiently small number. Let $Y_1 = \{y_1^1, y_2^1, \dots, y_{n_1}^1\}$ be a predefined finite subset of Y , where $n_1 = |Y_1|$. Let $k = 1$.

Step 2: Find the optimal solution $\mu^k \in M(X)$ of problem (SIP(Y_k)) and the corresponding solution $\nu^k = (\nu_1^k, \nu_2^k, \dots, \nu_{n_k}^k) \in R^{n_k}$ of problem (DSIP(Y_k)), where μ^k is a discrete measure concentrated on $x_1^k, x_2^k, \dots, x_{m_k}^k$.

Step 3: Calculate

$$\delta(\mu^k) := \min_{y \in Y} \left\{ \int_X \varphi(x, y) d\mu^k(x) - g(y) \right\},$$

and denote by \bar{y}^k the minimizing argument.

Step 4: If $0 > \delta(\mu^k) > -\varepsilon$, then stop and output $V(\text{SIP}(Y_k))$ as an approximate optimal objective value of problem (P). If $\delta(\mu^k) \geq 0$, then stop and output $V(\text{SIP}(Y_k))$ as the optimal value $V(\text{P})$.

Step 5: Set $Y_{k+1} = Y_k \cup \{\bar{y}^k\} = \{y_1^{k+1}, y_2^{k+1}, \dots, y_{n_{k+1}}^{k+1}\}$, $n_{k+1} = n_k + 1$, and go to Step 2 with $k := k + 1$.

To establish the convergence of Algorithm 2.1, we need several lemmas, some of which are from [13].

Lemma 2.1. *Suppose that a solution sequence $\{\mu^k\}$ of $SIP(Y_k)$ generated by Algorithm 2.1 is bounded, i.e., there exists an $M > 0$ such that $\|\mu^k\| \leq M$ for all k , where $\|\cdot\|$ denote the total variation norm on $M(X)$. Then, there exists a subsequence of the sequence $\{\mu^k\}$ converging to an optimal solution of problem (GCAP).*

Proof. See the proof of Theorem 5.1 of [13]. \square

Remark 1. The boundedness assumption in Lemma 2.1 can be technically ensured, as it is done in the beginning of Section 5 of [13]. More specifically, it is achieved by considering a compact set $\tilde{Y} := Y \cup \{\tilde{y}\}$, where \tilde{y} is a real number we choose with

$$\tilde{y} \notin Y, \quad \varphi(x, \tilde{y}) = -1, \quad g(\tilde{y}) = -M,$$

where $M > 0$ is chosen sufficiently large. According to this fact, we can imply that $\|\mu^k\| \leq M$.

Lemma 2.2. *Let*

$$\delta(\mu^k) := \min_{y \in Y} \left\{ \int_X \varphi(x, y) d\mu^k(x) - g(y) \right\}.$$

Then

$$\delta(\mu^k) \rightarrow \delta(\mu^*) \quad \text{as } k \rightarrow \infty,$$

where μ^ is an optimal solution of problem (GCAP).*

Proof. Define

$$G(y, \mu) := \int_X \varphi(x, y) d\mu(x) - g(y).$$

By Lemma 2.1, there exists a subsequence $\{\mu^{k_l}\} \subset M(X)$ converging to μ^* . Let $y^{k_l} \in Y$ denote the optimizer for each μ^{k_l} , i.e.,

$$G(y^{k_l}, \mu^{k_l}) \leq G(y, \mu^{k_l}), \quad \forall y \in Y.$$

Since Y is compact, we may assume that $y^{k_l} \rightarrow y^*$ for some $y^* \in Y$. Letting $l \rightarrow \infty$ in the inequality above, we have

$$G(y^*, \mu^*) \leq G(y, \mu^*), \quad \forall y \in Y.$$

This, in turn, implies that

$$\delta(\mu^{k_l}) = G(y^{k_l}, \mu^{k_l}) \rightarrow G(y^*, \mu^*) = \min_{y \in Y} G(y, \mu^*) = \delta(\mu^*).$$

\square

Lemma 2.3. *Let μ^k be an optimal solution for problem $(SIP(Y_k))$. If there exists a $\bar{\mu} \in M^+(X)$ satisfying*

$$\int_X \varphi(x, y) d\bar{\mu}(x) > 1, \quad \forall y \in Y,$$

then

$$|V(\text{GCAP}) - V(\text{SIP}(Y_k))| \leq \left| \delta(\mu^k) \int_X f(x) d\bar{\mu}(x) \right|.$$

Proof. See the proof of Theorem 6.5 of [13], which is given for the case when $\delta(\mu^k) < 0$. Since μ^k is feasible for problem (GCAP) with $V(\text{GCAP})=V(\text{SIP}(Y_k))$, the result follows readily. \square

To continue, we need the following assumption.

Assumption 2. There exists an absolutely continuous measure (with respect to the Lebesgue measure) $\bar{\mu} \in M^+(X)$ satisfying

$$\int_X \varphi(x, y) d\bar{\mu}(x) > 1, \quad \forall y \in Y. \quad (1)$$

Remark 2. $\bar{\mu}$ is found easily to achieve Assumption 2. Without loss of generality, we assume that the set $X^+ := \{x \in X \mid \varphi(x, y) > 0, \text{ for each } y \in Y\}$ be nonempty. We define $\bar{\mu}(x)$ as follows

$$\bar{\mu}(x) = \begin{cases} \bar{M} & x \in X^+, \\ 0 & \text{elsewhere,} \end{cases}$$

where \bar{M} is a sufficiently large real number such that $\bar{M} \times \varphi(x, y) > 1$, for $(x, y) \in X^+ \times Y$. Then $\bar{\mu} \in M^+(X)$ is an absolutely continuous measure which satisfies Assumption 2.

Base on Assumption 2, we have the following result on the error bound for the difference between the optimal value of problem (SIP(Y_k)) and problem (P).

Theorem 2.4. Let μ^k be an optimal solution to problem (SIP(Y_k)), and let Assumption 2 be satisfied. If $\delta(\mu^k) < 0$, then

$$|V(P) - V(\text{SIP}(Y_k))| \leq \left| \delta(\mu^k) \int_X f(x) d\bar{\mu}(x) \right|.$$

Proof. For notational simplicity, we may, without loss of generality, take X to be a compact interval. Let $h_l^k \in L_1(X)$, for $l > 0$, be a step function defined by

$$h_l^k(x) = \begin{cases} a_i^k l & \text{on } \left[x_i^k - \frac{1}{2l}, x_i^k + \frac{1}{2l} \right], \quad i = 1, 2, \dots, m_k, \\ 0 & \text{elsewhere,} \end{cases}$$

where $a_i^k = \mu^k(x_i^k)$, $i = 1, 2, \dots, m_k$, with necessary modifications if some x_i^k s happen to be end points of the interval, and l be a sufficiently large such that $\bigcap_{i=1}^{m_k} [x_i^k - \frac{1}{2l}, x_i^k + \frac{1}{2l}]$ is empty. Let $\mu_l^k \in M(X)$ be the corresponding absolutely continuous measure with the density function h_l^k . Since, for any $p \in C(X)$, we have

$$\begin{aligned} \int_X p(x) d\mu_l^k(x) &= \int_X p(x) h_l^k(x) dx \\ &= \sum_{i=1}^{m_k} \frac{p(\bar{x}_i^k)}{l} a_i^k l, \quad \text{where } \bar{x}_i^k \in \left[x_i^k - \frac{1}{2l}, x_i^k + \frac{1}{2l} \right], \\ &\rightarrow \sum_{i=1}^{m_k} p(x_i^k) a_i^k = \int_X p(x) d\mu^k(x) \quad \text{as } l \rightarrow \infty. \end{aligned}$$

Thus, μ_l^k is weak* convergent to μ^k when $l \rightarrow \infty$.

Let $\bar{\mu}_l^k := \mu_l^k - \delta(\mu^k)\bar{\mu}$ and $\bar{\mu}^k := \mu^k - \delta(\mu^k)\bar{\mu}$, where $\bar{\mu} \in M^+(X)$ is the absolutely continuous measure which satisfies Assumption 2. Then, it holds that

$$\begin{aligned} \int_X \varphi(x, y) d\bar{\mu}^k(x) - g(y) &= \int_X \varphi(x, y) d\mu^k(x) - g(y) - \delta(\mu^k) \int_X \varphi(x, y) d\bar{\mu}(x) \\ &> \delta(\mu^k) - \delta(\mu^k) = 0 \end{aligned}$$

for all $y \in Y$, this shows that there exists a $\delta > 0$ such that

$$\int_X \varphi(x, y) d\bar{\mu}^k(x) - g(y) \geq \delta.$$

Note that the left-hand side above is continuous in y and hence attains a positive minimum.

On the other hand, since $\bar{\mu}_l^k$ is weak* convergent to $\bar{\mu}^k$, it can be shown that there exists a subsequence of the sequence $\{\bar{\mu}_l^k\}$, denoted by the original sequence, such that

$$\max_{y \in Y} \left| \int_X \varphi(x, y) d\bar{\mu}_l^k - \int_X \varphi(x, y) d\bar{\mu}^k \right| \rightarrow 0, \quad \text{as } l \rightarrow \infty. \quad (2)$$

For each fixed l , there exists a y_l , which minimizes the left-hand side. Furthermore, since Y is compact, the sequence $\{y_l\}$ contains a subsequence converging to a point $y^* \in Y$. Again, the subsequence is denoted by the original sequence $\{y_l\}$. Then, we have

$$\begin{aligned} \max_{y \in Y} \left| \int_X \varphi(x, y) d(\bar{\mu}_l^k - \bar{\mu}^k) \right| &= \left| \int_X \varphi(x, y_l) d(\bar{\mu}_l^k - \bar{\mu}^k) \right| \\ &\leq \left| \int_X (\varphi(x, y_l) - \varphi(x, y^*)) d\bar{\mu}_l^k \right| \\ &\quad + \left| \int_X \varphi(x, y^*) d(\bar{\mu}_l^k - \bar{\mu}^k) \right| \\ &\quad + \left| \int_X (\varphi(x, y^*) - \varphi(x, y_l)) d\bar{\mu}^k \right|, \end{aligned}$$

which implies (2).

Now we choose an ε such that $\delta \geq \varepsilon > 0$. For this ε , there exists a l^* such that if $l \geq l^*$ then

$$\left| \int_X \varphi(x, y) d\bar{\mu}_l^k(x) - g(y) - \left(\int_X \varphi(x, y) d\bar{\mu}^k(x) - g(y) \right) \right| \leq \varepsilon, \quad \forall y \in Y.$$

Thus, it follows that

$$0 \leq \delta - \varepsilon \leq \int_X \varphi(x, y) d\bar{\mu}^k(x) - g(y) - \varepsilon \leq \int_X \varphi(x, y) d\bar{\mu}_l^k(x) - g(y), \quad \forall y \in Y.$$

Hence, for $l \geq l^*$, $\bar{\mu}_l^k$ is feasible for problem (P) and we have

$$\int_X f(x) d\mu^k(x) = V(\text{SIP}(Y_k)) \leq V(\text{GCAP}) \leq V(P) \leq \int_X f(x) d\bar{\mu}_l^k(x),$$

which implies

$$\begin{aligned} |V(P) - V(\text{SIP}(Y_k))| &\leq \left| \int_X f(x) d\bar{\mu}_l^k(x) - \int_X f(x) d\mu^k(x) \right| \\ &= \left| \int_X f(x) d\bar{\mu}_l^k(x) - \delta(\mu^k) \int_X f(x) d\bar{\mu}(x) - \int_X f(x) d\mu^k(x) \right| \\ &\leq \left| \int_X f(x) d\bar{\mu}_l^k(x) - \int_X f(x) d\mu^k(x) \right| + \left| \delta(\mu^k) \int_X f(x) d\bar{\mu}(x) \right| \\ &\rightarrow \left| \delta(\mu^k) \int_X f(x) d\bar{\mu}(x) \right|, \quad \text{as } l \rightarrow \infty. \end{aligned}$$

□

Theorem 2.5. *Let μ^k be an optimal solution to problem $(SIP(Y_k))$, and let Assumption 2 be satisfied. If $\delta(\mu^k) \geq 0$, then $V(P) = V(SIP(Y_k))$.*

Proof. Similar to Theorem 1. □

For each iterate k in Algorithm 2.1, we add a new constraint to problem $(SIP(Y_k))$ to get new subproblem $(SIP(Y_{k+1}))$. It means that the number of constraints is increasing after one iteration in Algorithm 2.1. For efficiency, we keep the active constraints and drop the inactive constraint for each iterates. The modified algorithm will reduce the dimension of problem. We modify the last step of Algorithm 2.1 as follows.

Algorithm 2.2. Modified cutting plane algorithm.

Steps 1–4: Same as in Algorithm 2.1.

Step 5: Let

$$\begin{aligned} A_k &= \{y_i^k \in Y_k \mid \nu_i^k > 0\}, \\ n_{k+1} &= |A_k| + 1, \\ Y_{k+1} &= A_k \cup \{\bar{y}^k\} = \{y_1^{k+1}, y_2^{k+1}, \dots, y_{n_{k+1}}^{k+1}\}, \end{aligned}$$

and go to Step 2 with $k := k + 1$.

Remark 3. In [13], it was proved that there exists a subsequence of $\{\mu^k\}$ converging to an optimal solution of problem (GCAP) under an additional condition that $\bar{y}^k \in A_{k+1}$ for all k .

3. Relaxed cutting plane method for solving SIP subproblem. In Step 2 of the proposed cutting plane algorithm, we need to solve a linear semi-infinite programming problem given in the form:

$$\begin{aligned} \max_{\nu \in R^n} \quad & \sum_{j=1}^n g(y_j) \nu_j \\ \text{subject to} \quad & \sum_{j=1}^n \varphi(x, y_j) \nu_j \leq f(x), \quad \forall x \in X, \\ & \nu \geq 0, \end{aligned} \tag{DSIP}$$

or equivalently, in the form:

$$\begin{aligned} \min_{\mu \in M(X)} \quad & \int_X f(x) d\mu(x) \\ \text{subject to} \quad & \int_X \varphi(x, y_j) d\mu(x) \geq g(y_j), \quad j = 1, 2, \dots, n, \\ & \mu \geq 0, \end{aligned} \tag{SIP}$$

where $y_j \in Y$, $j = 1, 2, \dots, n$, are given at each iterate. The iteration counter k is suppressed for simplicity throughout this section. In the literature on semi-infinite programming, the form given here as (DSIP) is usually regarded as in the primal

form, where the decision variable is finite-dimensional and the number of inequality constraints is infinite. Letting

$$b = \begin{bmatrix} g(y_1) \\ g(y_2) \\ \vdots \\ g(y_n) \end{bmatrix} \in R^n, \quad a \in C(X), \quad a(x) = \begin{bmatrix} \varphi(x, y_1) \\ \varphi(x, y_2) \\ \vdots \\ \varphi(x, y_n) \end{bmatrix} \in R^n,$$

we have the standard-form primal–dual pair:

$$\begin{aligned} & \max_{\nu \in R^n} b^T \nu \\ & \text{subject to } a(x)^T \nu \leq f(x), \quad \forall x \in X, \\ & \nu \geq 0, \end{aligned} \quad (\text{DSIP}')$$

and

$$\begin{aligned} & \min_{\mu \in \mathcal{M}(X)} \int_X f(x) d\mu(x) \\ & \text{subject to } \int_X a(x) d\mu(x) \geq b, \\ & \mu \geq 0. \end{aligned} \quad (\text{SIP}')$$

In this section, we propose a method to solve this pair of semi-infinite programming. There are various ways to solve semi-infinite programming problem, such as [14]. Here we use a powerful method, relaxed cutting plane method, to solve the semi-infinite programming problem. In view of a review article [12], the so-called cutting plane method or implicit exchange method is one of the key solution techniques. This statement is valid not only for linear semi-infinite programming problem, but also a good numerical technique for quadratic and convex programming problem [6, 20]. Basically, this approach finds a sequence of optimal solutions of the corresponding regular linear programs in a systematic way and shows that the sequence converges to an optimal solution of (DSIP'). We briefly present here the relaxed cutting plane method for the (DSIP').

Algorithm 3.1. Relaxed cutting plane algorithm for (DSIP').

Step 1: Let $m = 1$, choose a finite set $\{x_1, x_2, \dots, x_s\} \subset X$ and δ be an arbitrary small real number. Set $X_1 = \{x_1, x_2, \dots, x_s\}$ and $s = |X_1|$.

Step 2: Solve problem (LP(X_m)), where LP(X_m) is a linear programming problem defined as follows:

$$\begin{aligned} & \max_{\nu \in R^n} b^T \nu \\ & \text{subject to } a(x_i)^T \nu \leq f(x_i), \quad \text{for } i = 1, \dots, s + m - 1, \\ & \nu \geq 0, \end{aligned} \quad (\text{LP}(X_m)):$$

Let $\nu^m = \{\nu_1^m, \nu_2^m, \dots, \nu_n^m\}$ be an optimal solution.

Define $\phi_m(x) = \sum_{j=1}^n a(x)^T \nu_j^m - f(x)$.

Step 3: Find any $x_{s+m} \in X$ such that $\phi_m(x_{s+m}) > \delta$. If such x_{s+m} does not exist, stop and output ν^m as the solution. Otherwise, set $X_{m+1} = X_m \cup \{x_{s+m}\}$.

Step 4: Update $m := m + 1$, then go to step 2.

Let $(\text{DLP}(X_m))$ be the dual problem of $(\text{LP}(X_m))$, and $\mu^m = (\mu_1^m, \dots, \mu_{s+m-1}^m)^T$ be the optimal solution of $(\text{DLP}(X_m))$. We define a discrete measure $\tilde{\mu}^m$ on X such that

$$\tilde{\mu}^m(x) = \begin{cases} \mu_i^m, & \text{if } x = x_i \in X_m; \\ 0, & \text{if } x \notin X_m. \end{cases}$$

Furthermore, let $B'_m = \{i_1^m, \dots, i_{k_m}^m\}$ and $B_m = \{j_1^m, \dots, j_{p_m}^m\}$ denote the index sets defined by $B'_m = \{i | \tilde{\mu}^m(x_i) > 0\}$ and $B_m = \{j | \nu_j^m > 0\}$, respectively. Recalling the definition of $\phi_m(x)$, we define $\phi_j^m = \sum_{i \in B'_m} a_j(x_i) \mu_i^m - b_j$, for $j = 1, \dots, n$. Thus, we have the following convergence properties with respect to the sequence of the solutions generated by the cutting plane algorithm.

Theorem 3.1. *Given any $\delta > 0$, assume in each iteration, that:*

- (A1) $(\text{LP}(X_m))$ has a bounded feasible domain;
- (A2) $(\text{DLP}(X_m))$ is nondegenerate.

Moreover, there exists a $\bar{\delta} > 0$ such that:

- (A3) $\tilde{\mu}^m(x_i) \geq \bar{\delta}$, $\forall i \in B'_m$.
- (A4) $\phi_j^m < -\bar{\delta}$, $\forall i \notin B_m$.
- (A5) M_m^T has a square submatrix D_m with rank $p_m (= |B_m|)$ and $|\det(D_m)| > \bar{\delta}$, where M_m be a $p_m \times k_m$ matrix with its j th row vector is $(a_j(t_{i_1^m}), \dots, a_j(t_{i_{k_m}^m}))$, for $j \in B_m$.

Then, the proposed scheme terminates in a finite number of iterations.

Proof. The proof follows from similar arguments as these given for Theorem 2.1 and 2.3 of [19], respectively. \square

4. Numerical examples. In this section, Algorithm 2.1 and Algorithm 2.2 are used to solve two examples given below. In Step 2 of Algorithm 2.1 and Algorithm 2.2, we need to solve the semi-infinite programming subproblems $(\text{SIP}(Y_k))$ and $(\text{DSIP}(Y_k))$ for problem (P). In the following numerical experiment, for efficiency, we perform the relaxed cutting plane scheme, which stated in Section 3, to solve $(\text{SIP}(Y_k))$ and $(\text{DSIP}(Y_k))$ simultaneously. For comparison, we have implemented Algorithm 2.1, Algorithm 2.2 and the traditional discretization method. In discretization method, we discretize the set X into a finite subset $X_N = \{x_0, x_1, \dots, x_N\} \subset X$. Problem (P) becomes a linear semi-infinite programming problem (P_N) listed as follows:

$$\begin{aligned} & \min \sum_{i=1}^N f(x_i) z_i \Delta x_i \\ & \text{subject to} \quad \sum_{i=1}^N \varphi(x_i, y) z_i \Delta x_i \geq g(y), \quad \forall y \in Y, \\ & \quad \quad \quad z_i \geq 0 \quad \text{for } i = 1, 2, \dots, N. \end{aligned} \tag{P}_N$$

Here $\Delta x_i = x_i - x_{i-1}$ and let $z_i = h(x_i)$ for $i = 1, 2, \dots, N$. After solving Problem (P_N) , we consider the optimal solution h for problem (P) as a step function by setting $h(x) = z_i$, for $x \in [x_{i-1}, x_i)$, $i = 1, 2, \dots, N$. First, we consider the following example:

Example 1.

$$\begin{aligned} & \inf_{h \in L_1(X)} \int_{-1}^1 h(x) dx \\ & \text{subject to } \int_{-1}^1 ((x-y)^2 - 2)^2 h(x) dx \geq 1, \quad \forall y \in Y = [-1, 1], \\ & \quad h(x) \geq 0 \quad \text{a.e. on } X = [-1, 1], \end{aligned}$$

where $X = Y = [-1, 1]$, $f(x) = 1$, $\varphi(x, y) = ((x - y)^2 - 2)^2$ and $g(y) = 1$. The corresponding (GCAP) problem to Example 1 is given below:

$$\begin{aligned} & \min \int_{-1}^1 d\mu(x) \\ & \text{subject to } \int_{-1}^1 ((x-y)^2 - 2)^2 d\mu(x) \geq 1, \quad \forall y \in Y = [-1, 1], \quad (\text{GCAP 1}) \\ & \quad \mu \in M^+([-1, 1]). \end{aligned}$$

From [1], we know that the optimal solution for (GCAP 1) is given by a three-point measure $\mu(\cdot)$ with $\mu(-1) = \mu(1) = 1/9$, and $\mu(0) = 2/9$. The optimal objective value is $4/9 \approx 0.444444$. Although there exists an optimal solution for (GCAP 1), but we cannot find the optimal solution for Example 1. For instance, if we discretize X into 1000 equal partitions, i.e. $\Delta x_i = 1/500$ for $i = 1, 2, \dots, 1000$. We get 0.444221 be the optimal objective value and the optimal solution for problem (P₁₀₀₀) is (see Figure 1)

$$h^*(x) = \begin{cases} 56.4313 & \text{if } x \in [-1, -0.998]; \\ 64.8334 & \text{if } x \in [0, 0.010]; \\ 47.5400 & \text{if } x \in [0.010, 0.012]; \\ 53.4519 & \text{if } x \in [0.998, 1]; \\ 0 & \text{elsewhere.} \end{cases}$$

Since $\int_{-1}^1 d\mu(x) = \int_{-1}^1 h(x)dx$, the values of $h(-1)$, $h(0)$ and $h(1)$ will go to infinity as the norm of partition on X goes to 0. Hence our attention is to investigate the infimum of the objective function value. Figure 2 shows the inequality constraint $\int_{-1}^1 ((x-y)^2 - 2)^2 h^*(x) dx - 1$, for $y \in [0, 1]$. The inequality constraint becomes active at two points near ± 0.7 as shown in Figure 2.

In this implementation, we also use the relaxed cutting plane method which introduce in Section 3 to solve problem (P_N) when we apply discretization method. In Algorithm 1 and Algorithm 2, we set $\varepsilon = 10^{-4}$ as a terminating condition in Step 4, and choose the initial points $\{y_1^1, \dots, y_{n_1}^1\} \subset Y_1$ arbitrarily to start Algorithm 2.1 and Algorithm 2.2. We use MATLAB (version 7.0) which was installed on a Pentium 4, 3.40 GHZ personal computer to implement these three algorithms. The

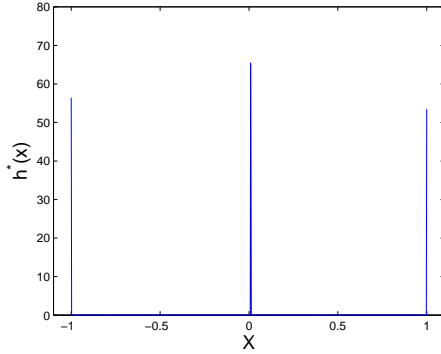


FIGURE 1. Optimal solution for P_{1000}

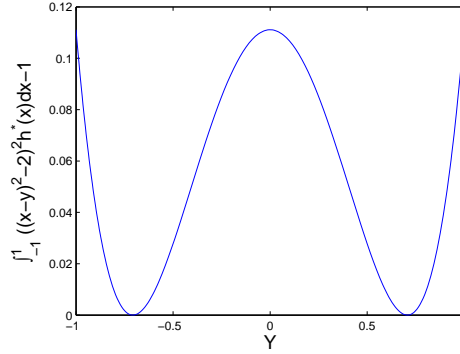


FIGURE 2. Inequality Constraint in P_{1000}

numerical results obtained are listed in Table 1.

	Y_1	optval	cpu	iter	$\delta(\mu^k)$
Discretizaion	-	0.444221	25.9961	-	-
Algorithm 1	$\{0.5\}^*$	0.444221	3.7447	10	-6.8430e-5
Algorithm 2	$\{0.5\}$	0.444221	3.5868	10	-6.8430e-5
Algorithm 1	$\{-1/\sqrt{2}, 1/\sqrt{2}\}$	0.444444	1.2879	5	-1.4283e-7
Algorithm 2	$\{-1/\sqrt{2}, 1/\sqrt{2}\}$	0.444444	1.2055	5	-1.4283e-7

*: 0.5 is the initial point

Table 1: Numerical results for Example 1

Here **optval** means optimal value, **iter** denotes the iteration number in our algorithms and **cpu** means the computer time spent. Notice that the value of $\delta(\mu^k)$ is not greater than 0, it says that we get an approximate optimal objective value by applying Algorithm 2.1 and Algorithm 2.2. From Theorem 2.4, we can evaluate the error bound between $V(P)$ and $V(\text{SIP}(Y_k))$ by setting a discrete measure $\bar{\mu}$ on $X = [-1, 1]$ defined by

$$\bar{\mu}(0) = 1 \quad \text{and} \quad \bar{\mu}(x) = 0 \quad \text{if } x \neq 0.$$

Then $\bar{\mu}$ satisfies the Equation 1 in Assumption 2. So we have $\int_X d\bar{\mu}(x) = 1$ and $|V(P) - V(\text{SIP}(Y_k))| \leq |\delta(\mu^k)|$. Observing the numerical results in Table 1, our algorithms perform very well when compared with the traditional discretization method. Also, we find that the results obtained are getting better when the initial points chosen are close to the active points. We implement Example 2 for further numerical experience.

Example 2.

$$\begin{aligned} & \inf_{h \in L_1(X)} \int_{-\pi/2}^{\pi/2} h(x) dx \\ & \text{subject to} \quad \int_{-\pi/2}^{\pi/2} 2 \sin((x - y)^2) h(x) dx \geq 1, \quad \forall y \in Y = [-\pi/2, \pi/2], \\ & \quad \quad \quad h(x) \geq 0 \quad \text{a.e. on } X = [-\pi/2, \pi/2], \end{aligned}$$

where $X = Y = [-\pi/2, \pi/2]$, $f(x) = 1$, $\varphi(x, y) = 2 \sin((x - y)^2)$ and $g(y) = 1$. The same as result in Example 1, we cannot find the optimal solution (see Figure 3). We implement Example 2 by discretization method with $N = 1000$, we obtain the optimal objective value 1.412739 and approximate optimal solution is

$$h^*(x) = \begin{cases} 14.6972 & \text{if } x \in [-1.0995, -1.0963]; \\ 38.8489 & \text{if } x \in [-1.0963, -1.0921]; \\ 14.8452 & \text{if } x \in [-1.0921, -1.0889]; \\ 10.6013 & \text{if } x \in [-0.3173, -0.3151]; \\ 76.0138 & \text{if } x \in [-0.3151, -0.3119]; \\ 69.8384 & \text{if } x \in [-0.3119, -0.3087]; \\ 69.7603 & \text{if } x \in [0.3110, 0.3142]; \\ 76.0692 & \text{if } x \in [0.3142, 0.3174]; \\ 10.6235 & \text{if } x \in [0.3174, 0.3206]; \\ 13.5457 & \text{if } x \in [1.0889, 1.0921]; \\ 41.4194 & \text{if } x \in [1.0995, 1.1027]; \\ 13.4259 & \text{if } x \in [1.1027, 1.1059]; \\ 0 & \text{elsewhere,} \end{cases}$$

From Figure 4, the inequality constraint $\int_{-\pi/2}^{\pi/2} 2 \sin((x - y)^2) h^*(x) dx - 1$, for $y \in [-\pi/2, \pi/2]$, becomes active near 0, ± 0.9 and $\pm 0.5\pi$.

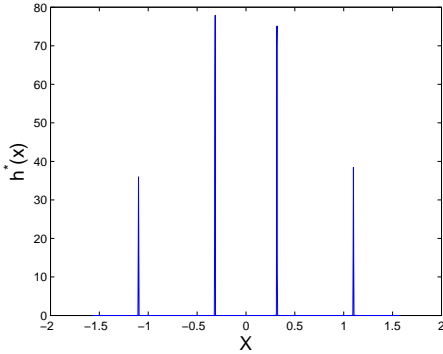


FIGURE 3. Optimal solution for P_{1000}

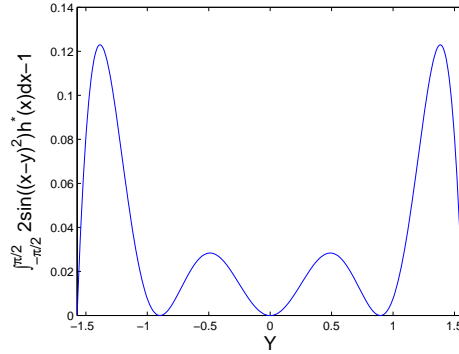


FIGURE 4. Inequality Constraint in P_{1000}

Finally, we list the numerical results obtained in Table 2. Since $\delta(\mu^k)$ is negative, we obtain an approximate value for Example 2. Defining a discrete measure $\bar{\mu}$ on $X = [-\pi/2, \pi/2]$ as follows:

$$\bar{\mu}\left(-\frac{\pi}{2}\right) = \bar{\mu}\left(\frac{\pi}{2}\right) = 1, \bar{\mu}(0) = 3 \text{ and } \bar{\mu}(x) = 0 \text{ if } x \neq \pm\frac{\pi}{2}, 0,$$

then we have $\int_X d\bar{\mu}(x) = 5$ and from Theorem 2.4, $|V(P) - V(\text{SIP}(Y_k))| \leq |5\delta(\mu^k)|$. In Table 2, our algorithms also perform much better than discretization method. Even though the difference of CPU time, which record in Table 1 and Table 2,

between Algorithm 2.1 and Algorithm 2.2 is small, we believe that Algorithm 2.2 will be much efficiency when the set of active points is large.

	Y_1	optval	cpu	iter	$\delta(\mu^k)$
Discretizaion	-	1.412739	63.9204	-	-
Algorithm 1	$\{\pm\pi/2\}$	1.412729	3.2798	7	-7.5753e-6
Algorithm 2	$\{\pm\pi/2\}$	1.412797	3.2339	7	-3.8378e-6
Algorithm 1	$\{\pm\pi/2, \pm 0.9, 0\}$	1.412797	1.1998	1	-2.8189e-5
Algorithm 2	$\{\pm\pi/2, \pm 0.9, 0\}$	1.412797	1.0792	1	-2.8189e-5

Table 2: Numerical results for Example 2

5. Conclusion. In this paper, we present two algorithms to solve the infinite-dimensional linear programming in L_1 spaces. This kind problem is related to the general capacity problem. The value of optimal solution $h^*(x)$ for the problem (P) is unbounded for some $x \in X$, that is, the optimal solution does not exist for problem (P), this is the main difference between problem (P) and problem (GCAP). As the optimal solution does not exist, we construct a sequence of approximate solution to approximate the optimal solution of problem (P). The numerical results obtained also show that the optimal solution $h^*(x)$ for problem (P) is not bounded. We have good numerical results, which compare with traditional discretization method, listed in Table 1 and Table 2.

Acknowledgments. The authors thank Dr. Yanqun Liu of RMIT, Australia, for his discussion and comments in the earliest version of this paper, and also thank reviewers' comments and suggestions, which have helped us to improve this paper.

REFERENCES

- [1] E. J. Anderson and P. Nash, "Linear Programming in Infinite-Dimensional Spaces: Theory and Applications," John Wiley & Sons, Chichester, 1987.
- [2] E. J. Anderson, A. S. Lewis and S. Y. Wu, *The capacity problem*, Optimization, **20** (1989), 725–742.
- [3] G. Choquet, *Theory of capacities*, Annales de l'Institut Fourier, **5** (1954), 131–295.
- [4] M. A. Dahleh and J. B. Pearson, Jr., *L^1 -optimal compensators for continuous-time systems*, IEEE Transaction on Automatic Control, **32** (1987), 889–895.
- [5] M. A. Dahleh and I. J. Diaz-Bobillo, "Control of Uncertain Systems: A Linear Programming Approach," Prentice-Hall, Englewood Cliffs, N.J., 1995.
- [6] S. C. Fang, C. J. Lin and S. Y. Wu, *Solving quadratic semi-infinite programming problems by using relaxed cutting-plane scheme*, Journal of Computational and Applied Mathematics, **129** (2001), 89–104.
- [7] L. Finlay, V. Gaitsgory and I. Lebedev, *Linear programming solutions of periodic optimization problems: Approximation of the optimal control*, Journal of Industrial and Management Optimization, **3** (2007), 399–413.
- [8] B. Fugled, *On the theory of potentials in locally compact spaces*, Acta Mathematica, **103** (1960), 139–215.
- [9] J. R. Gabriel and O. Hernandez-Lerma, *Stong duality of the general capacity problem in metric spaces*, Mathematical Methods of Operations Research, **53** (2001), 25–34.
- [10] M. Gerdt and Martin Kunkel, *A nonsmooth Newton's method for discretized optimal control problems with state and control constraints*, Journal of Industrial and Management Optimization, **4** (2008), 247–270.
- [11] Y. Gong and X. Xiang, *A class of optimal control problems of systems governed by the first order linear dynamic equations on time scales*, Journal of Industrial and Management Optimization, **5** (2009), 1–10.
- [12] R. Hettich and K. O. Kortanek, *Semi-infinite programming: Theory, method, and applications*, SIAM Review, **35** (1993), 380–429.

- [13] H. C. Lai and S. Y. Wu, *Extremal points and optimal solutions for general capacity problems*, Mathematical Programming, **54** (1992), 87–113.
- [14] Y. Liu, K. L. Teo and S. Ito, *A dual parameterization approach to linear–quadratic semi-infinite programming problems*, Optimization Methods and Software, **10** (1999), 471–491.
- [15] M. Ohtsuka, *Generalized capacity and duality theorem in linear programming*, Journal of Science, Hiroshima University Series A–I, **30** (1966), 45–56.
- [16] K. L. Teo, C. J. Goh and K. H. Wong, “A Unified Approach to Optimal Control Problems,” Longman Scientific & Technical, England, 1991.
- [17] R. J. Vanderbei, *Extreme optics and the search for earth-like planets*, Mathematical Programming, **112** (2008), 255–272.
- [18] C. Z. Wu and K. L. Teo, *Global impulsive optimal control computation*, Journal of Industrial and Management Optimization, **2** (2006), 435–450.
- [19] S. Y. Wu, S. C. Fang and C. J. Lin, *Relaxd cutting plane method for solving linear semi-infinite programming problems*, Journal of Optimization Theory and Applications, **99** (1998), 759–779.
- [20] S. Y. Wu and S. C. Fang, *Solving convex programs with infinitely many linear constraints by a relaxed cutting plane method*, Computers and Mathematics with Applications, **38** (1999), 23–33.
- [21] S. Y. Wu, S. C. Fang and C. J. Lin, *Solving general capacity problem by relaxed cutting plane approach*, Annals of Operations Research, **103** (2001), 193–211.
- [22] M. Yamasaki, *On a capacity problem raised in connection with linear programming*, Journal of Science, Hiroshima University Series A–I, **30** (1966), 57–73.

Received August 2008; 1st revision January 2009; final revision July 2009.

E-mail address: sito@ism.ac.jp

E-mail address: soonyi@mail.ncku.edu.tw

E-mail address: 11894103@mail.ncku.edu.tw

E-mail address: K.L.Teo@curtin.edu.au