INSTITUTE FOR POLICY SCIENCE RESEARCH REPORT NO. 87-B-1

A BISECTION METHOD FOR SOLVING SEMI-INFINITE PROGRAMS

by

KAORU TONE

A BISECTION METHOD FOR SOLVING SEMI-INFINITE PROGRAMS

by

Kaoru Tone

Graduate School for Policy Science, Saitama University
Urawa, Saitama 338, Japan
(February 20, 1987)

ABSTRACT

Semi-infinite programs will be solved by bisections within the framework of LPs. No gradient information is needed, contrary to the usual Newton-Raphson type methods for solving semi-infinite programs. The rate of convergence is linear. The method has a stable convergence feature derived from the bisection rule.

1. Problem

We consider general linear optimization problems with infinitely many constraints.

(P) $\min c^T y$

subject to

$$\sum_{r=1}^{n} a_r(s) y_r \geqq b(s) \qquad \text{for all } s \in S$$

where c is an n-dimensional constant vector,

y is an n-dimensional variable vector,

S is a nonempty compact subset of $R^m (m \geqq 1)$ and

$a_1, \ldots, a_n$ and b are continuous real functions on S.

The linear program dual to (P) is:

(D) $\max \sum_{s \in S} b(s) x(s)$

subject to

$$\sum_{s\in S} a_r(s)x(s)=c_r \qquad r=1,\ldots,n$$

$x(s)\geqq 0$ for all $s\in S$ and $x(s)=0$ except for a finite subset of S.

2. Outline of the Method

The method consists of three main parts: initial discretization, deletion and subdivision. The discretized problems are solved by the simplex method throughout the iterations.

Step 0. (Initial discretization)

The dual pair (P)-(D) is discretized,i.e. the infinite index set S is replaced by a finite set. Let the finite set be $\{s^1,\ldots,s^k\}$. We call such sets grid.

Solve the resulting dual pair of linear programs (P_0)-(D_0) by means of the simplex method.

(P_0) min $c^T y$

subject to

$$\sum_{r=1}^{n} a_r(s^i)y_r \geqq b(s^i) \qquad i=1,\ldots,k$$

(D_0) max $\sum_{i=1}^{k} b(s^i)x_i$

subject to

$$\sum_{i=1}^{k} a_r(s^i)x_i=c_r \qquad r=1,\ldots,n$$

$$x_i \geqq 0 \qquad i=1,\ldots,k.$$

Let the optimal solutions to (P_0) and (D_0) be

$y=(y_1,\ldots,y_n)^T$

and

$$x=(x_1,\ldots,x_k)^T.$$

<u>Step 1.</u> (Deletion)

Apply the 'Deletion rule' as explained later in Sections 3 and 4 to the grid $\{s^i\}$.

<u>Step 2.</u> (Subdivision or bisection)

Apply the 'Subdivision (bisection) rule' as explained in Sections 3 and 4 to the grid.

<u>Step 3.</u> (New (P_0)-(D_0))

Formulate new dual LPs (P_0)-(D_0) by deleting/augmenting constraints/variables to (P_0)-(D_0).

Solve them by the simplex method.

<u>Step 4.</u> (Convergence check)

Stop the process if the subdivision parameter as explained in Sections 3 and 4 becomes less than the tolerance.

Otherwise go back to Step 1.

3. Details of the Method When S is One-Dimensional.

In this section we will show details of the method in case S is one-dimensional. Cases with dim(S)>1 will be discussed in Section 4.

3.1 Initial Discretization and Subdivision Parameter

Let the set S be $[a,b]\subset R$ and arrange the grid $s_0,\ldots,s_k$ as

$$a=s_0<s_1<\ldots<s_k=b \qquad (3.1)$$

where

$$s_i=s_0+i(b-a)/k \qquad (i=0,\ldots,k) \qquad (3.2)$$

We define the <u>subdivision parameter</u> (or mesh size) T to be

$$T=(b-a)/k \quad \text{(the length of an interval)} \qquad (3.3)$$

3.2 Solving (D_0)

We solve the dual program (D_0) by means of the simplex method. The reason for dealing with the dual program will be clarified later on. The optimal information related to the primal program is easilly obtained from the optimal basis of (D_0).

Let the optimal solutions to (P_0) and (D_0) be

$$y=(y_1,\ldots,y_n)^T \qquad (3.4)$$

and

$$x=(x_1,\ldots,x_k)^T. \qquad (3.5)$$

3.3 Deletion / Subdivision Rules

Since the optimal solutions (3.4)-(3.5) solve the discretized problems, we have, at grid point s_i,

$$\sum_{r=1}^{n} a_r(s_i)y_r=b(s_i) \qquad \text{if } x_i>0 \qquad (3.6)$$

and

$$\sum_{r=1}^{n} a_r(s_i)y_r \geqq b(s_i) \qquad \text{if } x_i=0. \qquad (3.7)$$

However, it is not certain if the relations

$$\sum_{r=1}^{n} a_r(s)y_r \geqq b(s) \qquad (3.8)$$

for every $s \in S$.

Let $\phi(s)=\sum_{r=1}^{n} a_r(s)y_r - b(s)$. (3.9)

The <u>discrepancy</u> $\delta(y)$ of y is defined as

$$\delta(y) = \min_{s \in [a,b]} \phi(s). \qquad (3.10)$$

A lower bound to $\delta(y)$ is given by

$$-\Delta = -(FM_1 + M_2)T^2/8 \qquad (3.11)$$

(Kortanek [2]),

where $F > |y_r| \quad (r=1,\ldots,n)$

$$M_1 = \max_{s \in S} \sum_{r=1}^{n} |a_r''(s)|$$

$$M_2 = \max_{s \in S} |b''(s)|$$

and

T is defined by (3.3).

It is easy to see that if at two successive grid points s_i and s_{i+1}, we have

$$\phi(s_i) > \Delta \text{ and } \phi(s_{i+1}) > \Delta$$

then it follows that

$$\phi(s) > 0 \quad \text{for every } s \in [s_i, s_{i+1}].$$

Thus, we have the deletion rule for grid points.

[Deletion Rule]

If at three successive grid points s_i, s_{i+1} and s_{i+2}, we have

$$\phi(s_i) > \Delta,\ \phi(s_{i+1}) > \Delta \text{ and } \phi(s_{i+2}) > \Delta \qquad (3.13)$$

then we delete s_{i+1} and hence the whole interval (s_i, s_{i+2}) from further consideration. Notice that the rule needs to be changed a little at the boundary points.

[Subdivision Rule]

We subdivide the remaining intervals by introducing a new grid at the mid-point of each interval.

Thus, we have

$$\text{new } T = T/2 \tag{3.14}$$

and

$$\text{new } \Delta = \Delta/4. \tag{3.15}$$

[Remark 1] Usually it is not easy to determine Δ as defined by (3.11). In such a case, Δ should be taken to be a threshold for deleting grid points. A smaller Δ deletes more grid points. If $\phi(s)$ is well approximated by a quadratic curve at a local maximum, the relation (3.15) will generally hold after the subdivision.

3.4 Solving the New LP

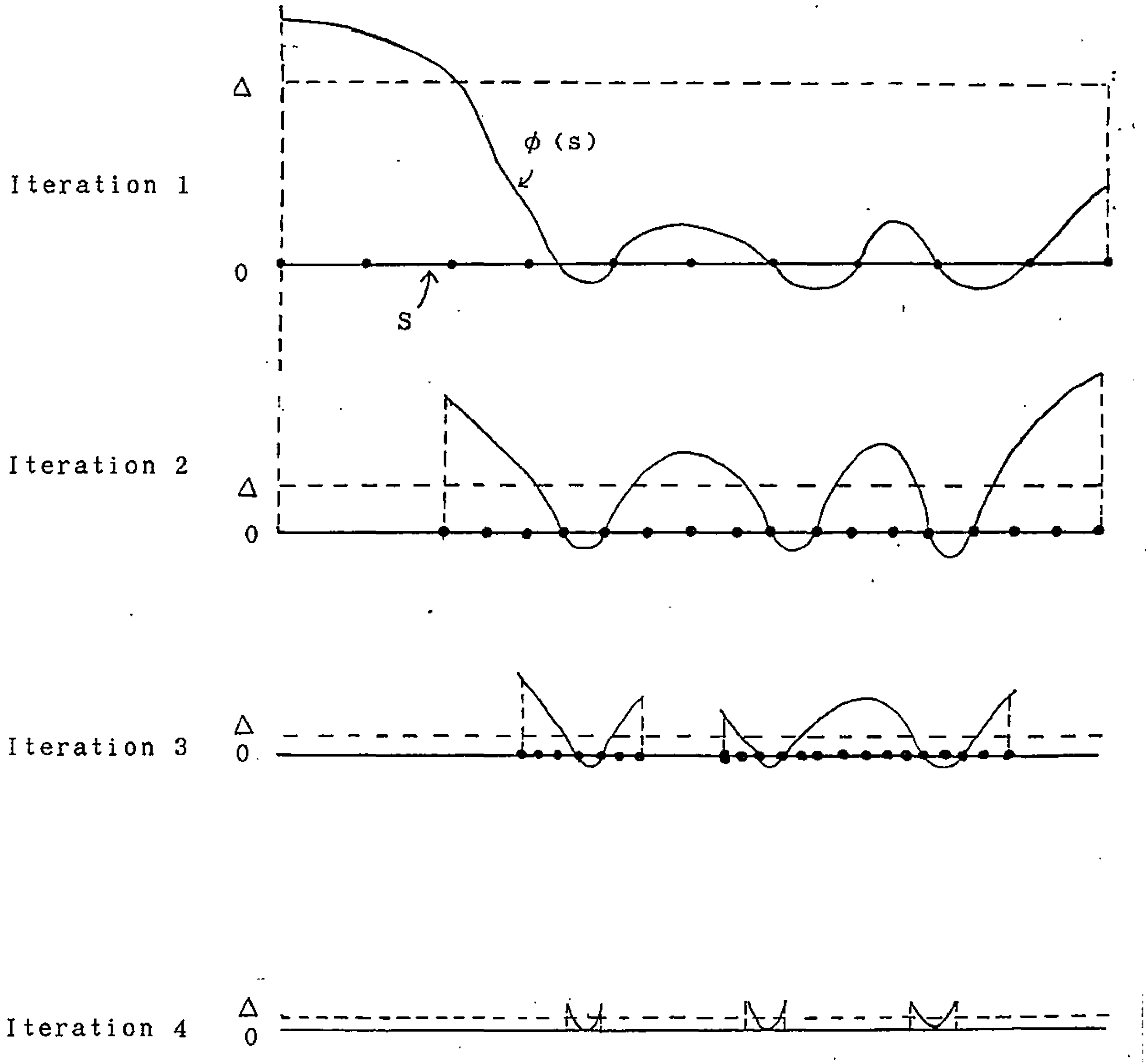
We delete the columns corresponding to the deleted grid points from the dual tableau and introduce new columns corresponding to the new grid points to the tableau. The new columns will be priced out by using the optimal dual basis of the preceding iteration and the primal simplex method will determine the new optimal solution.

3.5 Convergence Check

We stop the iterations if T comes to satisfy for some tolerance T_{tol},

$$T < T_{tol}. \tag{3.16}$$

[Remark 2] A typical process of subdivision (or bisection) is sketched in Fig. 1, where the curves represent $\phi(s)$ with s as abscissa and the tolerance Δ for each iteration is given by the dashed line.



Iteration 1
Δ
φ (s)
0
S
Iteration 2
Δ
0
Iteration 3
Δ
0
Iteration 4
Δ
0

Fig. 1

4. General Case.

In this section we will deal with the dual pair of problems (P)-(D) when S is a compact set with L=dim(S)>1.

4.1 Initial Discretization

We discretize S by using L-dimensional cubes with edge length T. The mesh points are the initial grid points $\{s^1,\dots,s^k\}$. The grid points are used to formulate (P_0) and (D_0), which are solved by the simplex meyhod. Let the optimal solutions be y and x.

4.2 Deletion and Subdivision (Bisection) Rules

Every grid has at most 2L neighbors.

[Deleting Rule]

If the relation

$$\phi(s)= \sum_{r=1}^{n} a_r(s)y_r - b(s) > \Delta \tag{4.1}$$

holds at a grid and its neighbors, then we will call the convex hull of such neighbors as 'a deleted domain'. And in the subdivision process which follow, we neglect the grid points inside the deleted domain. Δ is a threshold similar to (3.11) (see also [Remark 1]). For higher dimensional Ls, it would be difficult to estimate Δ by a formula such as (3.11). A practical way to estimate Δ is as follows:

After the initial LPs are solved, we estimate the discrepancy $\delta(y)$ by sampling s from S. The value will be used as the initial Δ, which will be updated by dividing 4 at each iteration.

[Subdivision Rule]

We introduce new grid points outside the deleted domain by subdividing each edge of the cubes at the mid-point. Thus, we have

$$\text{new } T=T/2 \tag{4.2}$$

and

$$\text{new } \Delta=\Delta/4. \tag{4.3}$$

4.3 Solving New LPs and Checking Convergence

These steps are quite similar to the one-dimensional case as explained in subsections 3.4 and 3.5.

[Remark 3] A typical subdivision process of the two-dimensional S is depicted in Fig.2.

Iteration 1

Iteration 2

Iteration 3

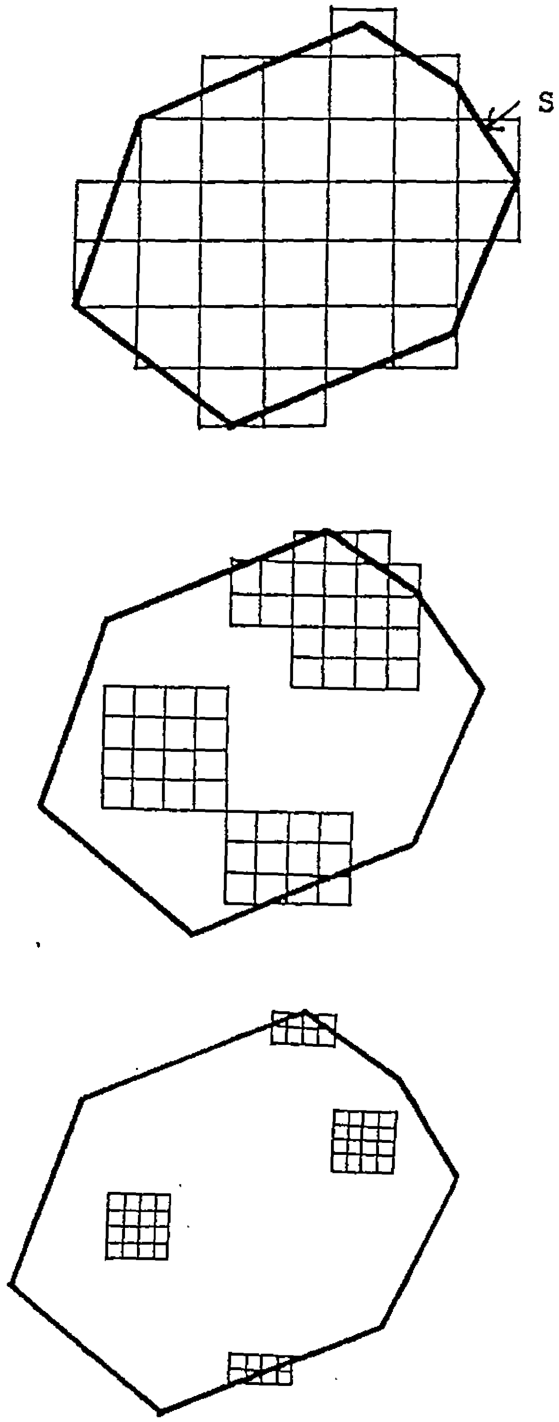


Fig. 2

Acknowledgment

I would like to express my sincere gratitude to Professor Abraham Charnes of the University of Texas at Austin for motivating me to work on the present subject while I was visiting Center for Cybernetic Studies, the University of Texas and doing the research [1] on the data envelopment analysis.

References

[1] Charnes A. and K. Tone,"A Computational Method for Solving DEA Problems with Infinitely Many DMUs," Research Report CCS 561, Center for Cybernetic Studies, the University of Texas at Austin (January 1987).

[2] Kortánek, K.O.,"Interpólation and Error Bounds for Semi-infinite Programs and Solution of Nonlinear Systems of Equations," (1979),Manuscript.