Optimal Solutions of Oracle Defined Optimization Problems

Laszlo Csirmaz

Central European University, Budapest

Set optimization meets finance II

Bruneck-Brunico, Italy, Sept 8–12, 2014

Outline

Vector optimization problem – our version

- The **problem space** is $\mathbb{R}^n$; $n \approx 100$.
- The linear **constraints** are given by the $m \times n$ matrix $A \in \mathbb{R}^{m\times n}$; $m \approx 1000$.
- The **feasible set** of the problem (to be optimized over) is

$$\{x \in \mathbb{R}^n \ : \ Ax \geq 0\,\}.$$

- There are p **objectives** to be **minimized**, given by the $p \times n$ matrix $P \in \mathbb{R}^{p\times n}$; $p \approx 10$.
- The **objective space** is $\mathbb{R}^p$.
- The **optimization problem** is to find all extremal solutions of

$$\min\,\{\,Px \in \mathbb{R}^p \ : \ x \in \mathbb{R}^n,\, Ax \geq 0\,\}.$$

$\Rightarrow$

Observe: $\{\,Px \ : \ Ax \geq 0\,\}$ is a **projection** of this polyhedron:

Vector optimization problem – our version

⇒ Vector optimization problem:

$$\min \{ Px \in \mathbb{R}^p : x \in \mathbb{R}^n, Ax \ge 0 \}. \qquad (*)$$

Define

$$\mathcal{Q} = \{ Px + z \in \mathbb{R}^p : \langle x, z \rangle \in \mathbb{R}^{n+p}, \; z \ge 0, Ax \ge 0 \};$$

this is a p-dimensional affine projection of the $n+p$-dimensional polyhedron

$$\mathcal{S} = \{ \langle x, z \rangle \in \mathbb{R}^{n+p} : (A, I)\langle x, z \rangle \ge 0 \}.$$

Proposition*

Under some mild assumptions, the extremal solutions of the vector optimization problem $(*)$ are exactly the vertices of the p-dimensional polyhedron $\mathcal{Q}$. □

*See also A. Löhne: *Projection of polyhedral cones*, ArXiv 1406.1708

~~Vector optimization~~ Vertex enumeration problem

Moving z to the problem space along with x, and increasing n by p, solving the vector optimization problem reduces to

Vertex enumeration problem for projections

Given the m facets of the n-dimensional polyhedron $\mathcal{S}$, that is, $\mathcal{S} = \{x \in \mathbb{R}^n : Ax \geq 0\}$ for a given $m \times n$ matrix A,
enumerate all vertices of its p-dimensional projection $\mathcal{Q} = \{Px : x \in \mathcal{S}\}$ for a given $p \times n$ matrix P.

~~Vector optimization~~ Vertex enumeration problem

Moving z to the problem space along with x, and increasing n by p, solving the vector optimization problem reduces to

Vertex enumeration problem for projections

Given the m facets of the n-dimensional polyhedron $\mathcal{S}$, that is, $\mathcal{S} = \{x \in \mathbb{R}^n : Ax \geq 0\}$ for a given $m \times n$ matrix A, **enumerate** all vertices of its p-dimensional projection $\mathcal{Q} = \{Px : x \in \mathcal{S}\}$ for a given $p \times n$ matrix P.

Some remarks:

- The same reduction works for arbitrary ordering cone C (not only for the non-negative orthant $C = \mathbb{R}^p_+$).
- $\mathcal{Q}$ is not bounded, has vertices **and** extremal rays (typically coming from the ordering cone). Both cases can be handled uniformly by using **homogeneous** coordinates.

Outline

An important fact on scalar linear programming

Observation on duality

When solving a scalar LP problem, we not only get the **solution**, but also a **proof** of the correctness of the solution (dual).

An important fact on scalar linear programming

Observation on duality

When solving a scalar LP problem, we not only get the **solution**, but also a **proof** of the correctness of the solution (dual).

Example:

$$\mathcal{Q} = \{ Px \,:\, x \in \mathbb{R}^n,\ Ax \ge 0 \}.$$

⇒ Given points $i \in \mathcal{Q}$ and $o \notin \mathcal{Q}$, find the intersection $[i,o] \cap \mathcal{Q}$. (This problem is from the inner loop of the *outer approximation algorithm*.) To get the boundary point of $\mathcal{Q}$ on $[i,o]$,

⇒ Solve the scalar LP $\max_{\lambda,x}\{\lambda :\ i+\lambda(o-i) = Px,\ Ax \ge 0,\ \lambda \ge 0\}$.

⇒ The solution gives the boundary point $b = i + \hat{\lambda}(o-i) \in \mathcal{Q}$ (primal), **and** a supporting hyperplane to $\mathcal{Q}$ at b proving that b is indeed optimal (coming from the dual).

Introducing the Separation Oracle

Distilling from the example: the polyhedron $\mathcal{Q} \subseteq \mathbb{R}^p$ is hidden, and can only be reached through questions to the

Facet Separation Oracle, *FSO*

Q: a point $y \in \mathbb{R}^p$.
A: if $y \in \mathcal{Q}$, then the answer is **inside**.
if $y \notin \mathcal{Q}$, then the answer is (the equation of) a **facet** of $\mathcal{Q}$ such that y is on the negative side of the facet.

Introducing the Separation Oracle

Distilling from the example: the polyhedron $\mathcal{Q} \subseteq \mathbb{R}^p$ is hidden, and can only be reached through questions to the

Facet Separation Oracle, $\boldsymbol{FSO}$

Q: a point $y \in \mathbb{R}^p$.
A: if $y \in \mathcal{Q}$, then the answer is **inside**.
if $y \notin \mathcal{Q}$, then the answer is (the equation of) a **facet** of $\mathcal{Q}$ such that y is on the negative side of the facet.

Vertex enumeration problem using $\boldsymbol{FSO}$

Given an $\boldsymbol{FSO}$ for the polyhedron $\mathcal{Q} \subseteq \mathbb{R}^p$ (and some initial data), **enumerate** all vertices and extremal rays of $\mathcal{Q}$.

Getting an answer from the Oracle is typically expensive. The main complexity measure is the number of Oracle questions.

Introducing the Separation Oracle

Vertex enumeration using Facet Separation Oracle

Vertex enumeration for projection (restated)

Given the $m \times n$ matrix A and the $p \times n$ matrix P,
enumerate all vertices of $\mathcal{Q} = \{ Px : x \in \mathbb{R}^n, Ax \ge 0 \} \subset \mathbb{R}^p$.

Claim

Solving linear multiobjective optimization can be reduced to vertex enumeration using $\boldsymbol{FSO}$.

Proof.

Implement $\boldsymbol{FSO}$ on the query $y \in \mathbb{R}^p$ as follows:

1. Pick a random inner point $i \in \mathcal{Q}$ (could be fixed!).
2. Solve the LP $\max_{\lambda,x}\{\lambda : i + \lambda(y - i) = Px, Ax \ge 0, \lambda \ge 0 \}$.
3. If $\hat{\lambda} \ge 1$ then return "y is an inner point."
4. Otherwise return the supporting hyperplane to $\mathcal{Q}$ as the facet.

(It will be a facet!) □

Outline

Vertex / facet enumeration

Classical vertex enumeration problem

Given all facets of a polyhedron,
enumerate all of its vertices.

Several vertex enumeration algorithms are known:*

- pivot algorithms – ranked, recursive, reversed
- incremental – double description method
- primal/dual
- backtrack algorithms

All complexity questions in this area are open.

Research problem

Which known method adapts best to our case? Other algorithms?

*K. Fukuda: *Vertex enumeration for polyhedra: algorithms and open problems*.

Outer approximation using double description

Double Description method for vertex enumeration with $\boldsymbol{FSO}$

To enumerate vertices of $\mathcal{Q}$ generate the *approximating sequence* $\mathcal{Q}_j \supseteq \mathcal{Q}_{j+1} \supseteq \cdots \supseteq \mathcal{Q}$ maintaining in each step

1. all vertices and facets of $\mathcal{Q}_j$,
2. for each vertex of $\mathcal{Q}_j$ whether it is known to be a vertex of $\mathcal{Q}$.

To get $\mathcal{Q}_{j+1}$ from $\mathcal{Q}_j$ pick a vertex y of $\mathcal{Q}_j$ which is **not known** to be a vertex of $\mathcal{Q}$. Call the $\boldsymbol{FSO}$ with y.

1. If the answer is "**inner**", mark y as a vertex of $\mathcal{Q}$.
2. Otherwise intersect $\mathcal{Q}_j$ with the facet of $\mathcal{Q}$ returned.

Stop when all vertices of $\mathcal{Q}_j$ are vertices of $\mathcal{Q}$; you are done.

The number of oracle calls is

$$\text{number of vertices of } \mathcal{Q} + \text{number of facets of } \mathcal{Q}.$$

This seems to be optimal. But is it?

In summary

Vector optimization problem

Find all extremal solutions of $\min\{ Px \in \mathbb{R}^p \,:\, x \in \mathbb{R}^n,\ Ax \geq 0 \}$.

$\Downarrow$

Vertex enumeration problem using *FSO*

Given an Facet Separation Oracle representation of $\mathcal{Q} \subseteq \mathbb{R}^p$, **enumerate** all vertices (and extremal rays) of $\mathcal{Q}$.

$\Downarrow$

Double Description method for vertex enumeration with *FSO*

Use the outer approximation method, learning in each step

- either a new vertex of $\mathcal{Q}$,
- or a new facet of $\mathcal{Q}$.

until the whole $\mathcal{Q}$ is known.

It works!

The algorithm was used successfully for combinatorial optimization problems with $p = 10$ (**ten**!) objectives. Some representative results:

$m \times n$	Vertices	Facets	Time
4055×370	19	58	1:10:10
4009×370	40	103	3:24:37
3891×358	30	102	3:34:31
3963×362	167	235	9:20:19
4007×370	318	356	13:20:08
4007×370	318	356	14:34:42
4007×370	297	648	22:02:39
3913×362	779	1269	37:15:33
3987×362	4510	7966	427:43:30
3893×362	10387	13397	716:36:32

Outline

Vertex Separation Oracle

Suppose the projection $\mathcal{Q}$ can be reached by inquiring the

Vertex Separation Oracle, $\boldsymbol{VSO}$

Q: (the equation of) a closed halfspace $H \subseteq \mathbb{R}^p$.
A: if $\mathcal{Q} \subseteq H$ then the answer is **inside**.
if $\mathcal{Q} \not\subseteq H$, then the answer is a **vertex** of $\mathcal{Q}$ not in H.

Accessing $\mathcal{Q}$ through a Vertex Separation Oracle $\boldsymbol{VSO}$ has some attractive properties:

- If the computation has to be aborted, we have several vertices (extremal solutions).
- The order of half-spaces submitted to the Oracle can be chosen (prioritized) so as to give a general view of the Pareto set, and refine it where necessary.

Inner approximation using double description

Double Description method for vertex enumeration with $\boldsymbol{VSO}$

To enumerate the vertices of $\mathcal{Q}$ generate the *approximating sequence* $\mathcal{Q}_j \subseteq \mathcal{Q}_{j+1} \subseteq \cdots \subseteq \mathcal{Q}$ maintaining in each step

1. all vertices and facets of $\mathcal{Q}_j$,
2. for each *facet* of $\mathcal{Q}_j$ whether it is know to be a facet of $\mathcal{Q}$.

To get $\mathcal{Q}_{j+1}$ form $\mathcal{Q}_j$ pick a facet f of $\mathcal{Q}_j$ which is not known to be a facet of $\mathcal{Q}$. Call the $\boldsymbol{VSO}$ with the half-space $f \geq 0$.

1. If the answer is **inside**, mark f as a facet of $\mathcal{Q}$, and continue.
2. Otherwise let $\mathcal{Q}_{j+1}$ be the convex hull of $\mathcal{Q}_j$ and the vertex returned.

Stop when all facets of $\mathcal{Q}_j$ are facets of $\mathcal{Q}$; you are done.

The number of Oracle calls this algorithm makes is also

$$\text{number of vertices of } \mathcal{Q} + \text{number of facets of } \mathcal{Q}.$$

How to implement the Vertex Separation Oracle?

Vertex enumeration for projection (restated)

Given the $m \times n$ matrix A and the $p \times n$ matrix P,
enumerate all vertices of $\mathcal{Q} = \{ Px \,:\, x \in \mathbb{R}^n,\, Ax \geq 0 \} \subset \mathbb{R}^p$.

Given the closed halfplane $\{y \in \mathbb{R}^p : \langle b, y\rangle \geq d\}$, the naïve idea is

1. Solve the LP $\hat{d} = \min_x \{ \langle b, Px\rangle \,:\, Ax \geq 0 \}$.
2. If $d \leq \hat{d}$, then return "$\mathcal{Q}$ is inside."
3. Otherwise return the point $\hat{y} \in \mathcal{Q}$, where the minimum $\hat{d} = \langle b, \hat{y}\rangle$ is taken.

How to implement the Vertex Separation Oracle?

Vertex enumeration for projection (restated)

Given the $m \times n$ matrix A and the $p \times n$ matrix P,
enumerate all vertices of $\mathcal{Q} = \{\, Px \,:\, x \in \mathbb{R}^n,\ Ax \geq 0 \,\} \subset \mathbb{R}^p$.

Given the closed halfplane $\{y \in \mathbb{R}^p : \langle b, y \rangle \geq d\}$, the naïve idea is

1. Solve the LP $\hat{d} = \min_x \{ \langle b, Px \rangle \,:\, Ax \geq 0 \,\}$.
2. If $d \leq \hat{d}$, then return "$\mathcal{Q}$ is inside."
3. Otherwise return the point $\hat{y} \in \mathcal{Q}$, where the minimum $\hat{d} = \langle b, \hat{y} \rangle$ is taken.

The problem is that $\hat{y}$ is a boundary point of $\mathcal{Q}$, but it is not necessarily a vertex if the normal b is not in "general position."

How to implement the Vertex Separation Oracle?

Vertex enumeration for projection (restated)

Given the $m \times n$ matrix A and the $p \times n$ matrix P,
enumerate all vertices of $\mathcal{Q} = \{ Px \,:\, x \in \mathbb{R}^n,\ Ax \ge 0 \} \subset \mathbb{R}^p$.

Given the closed halfplane $\{y \in \mathbb{R}^p : \langle b, y\rangle \ge d\}$, the naïve idea is

1. Solve the LP $\hat{d} = \min_x \{ \langle b, Px\rangle \,:\, Ax \ge 0 \}$.
2. If $d \le \hat{d}$, then return "$\mathcal{Q}$ is inside."
3. Otherwise return the point $\hat{y} \in \mathcal{Q}$, where the minimum $\hat{d} = \langle b, \hat{y}\rangle$ is taken.

A correct implementation would be to take a projective image of $\mathcal{Q}$ first, where the ideal hyperplane is in general position (there are others). The problem and solution closely relate to the geometric duality of F. Heyde and A. Löhne.

Thank you for your attention