Prof. F. Eisenbrand EPFL - DISOPT

Combinatorial Optimization

Adrian Bock Fall 2011

Sheet 3 October 20, 2011

General remark:
In order to obtain a bonus for the final grading, you may hand in written solutions to the exercise marked with a star at the beginning of the exercise session on November 1.

Exercise 1

Let $t(x) = Ax + b$ be the mapping that transforms the 2-dimensional unit ball into the ellipsoid $E(A, b)$. Draw $E(A, b)$ for $A = \begin{pmatrix} 1 & 3 \\ 2 & 5 \end{pmatrix}$ and $b = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$.

Solution

The first idea is to consider some points in the unit ball and to apply the transformation to it in order to obtain corresponding points in the ellipse (e.g. choose unit vectors).
Second, we can rewrite the description of $E(A, b)$ as follows:

$$E(A, b) = \{y \in \mathbb{R}^2 \colon \|A^{-1}(y - b)\| \leq 1\} = \{y \in \mathbb{R}^2 \colon (y - b)^T B^{-1}(y - b) \leq 1\}$$

where $B^{-1} = A^{-T} A^{-1}$. Note that B is symmetric and positive definite. We have from linear algebra that the eigenvectors of B define the principal directions of the ellipsoid and the square root of the corresponding eigenvalues determines their length, e.g. the equatorial radii.
In our example, we obtain $v_1 \approx \begin{pmatrix} 3.17 \\ 5.38 \end{pmatrix}$, $v_2 \approx \begin{pmatrix} -0.14 \\ 0.08 \end{pmatrix}$, the principal directions that are already scaled by the eigenvalues $\lambda_1 \approx 39$ and $\lambda_2 \approx 0.026$.
For an exact drawing or the exact eigenvectors and eigenvalues, consult your favourite computer algebra system.

Exercise 2

(i) Show that the unit simplex $\Delta = \text{conv}\{0, e_1, \ldots, e_n\} \subset \mathbb{R}^n$, where e_j are the standard unit vectors, has volume $\frac{1}{n!}$.

(ii) Show that the volume of the simplex $\text{conv}\{v_0, v_1, \ldots, v_n\} \subset \mathbb{R}^n$ is $\frac{|\det(v_1 - v_0, \ldots, v_n - v_0)|}{n!}$.

Solution

(i) The cases $n = 1, 2, 3$ are easy to verify. Prove the general formula by induction on n:
Consider the $(n-1)$-dimensional unit simplex defined by $0, e_1, \dots, e_{n-1}$ that lies on the boundary of Δ_n. Any point $x = (x_1, \dots, x_n) \in \Delta_n$ satisfies $\sum_{i=1}^n x_i \leq 1$. The idea consists in integrating over the last component $x_n \in [0, 1]$ while scaling the $(n-1)$-dimensional unit simplex by $(1-x_n)$. In other words, if we fix the coordinate $x_n = \beta$, we get an $(n-1)$-dimensional unit simplex defined by $0, (1-\beta)e_1, \dots, (1-\beta)e_{n-1}$ shifted by βe_n. Because of symmetry, this is the same as integrating over a scaling factor $\alpha \in [0, 1]$:

$$\mathrm{vol}_n(\Delta_n) = \int_0^1 \mathrm{vol}_{n-1}(\alpha\cdot\Delta_{n-1})d\alpha = \int_0^1 \alpha^{n-1}\mathrm{vol}_{n-1}(\cdot\Delta_{n-1})d\alpha = \int_0^1 \frac{\alpha^{n-1}}{(n-1)!}d\alpha = \frac{1}{n!}$$

(ii) Note that the volume of the simplex does not change by shifting it by $-v_0$. Thus we consider the simplex $\mathrm{conv}\{0, v_1 - v_0, \dots, v_n - v_0\} \subset \mathbb{R}^n$. Note that this simplex can be obtained from the unit simplex by applying the linear transformation $t(x) = Ax$ to it, where $A = \big(v_1 - v_0, \dots, v_n - v_0\big)$. From part i) and linear algebra we get

$$\begin{aligned}\mathrm{vol}(\mathrm{conv}\{v_0, v_1, \dots, v_n\}) &= \mathrm{vol}(\mathrm{conv}\{0, v_1 - v_0, \dots, v_n - v_0\}) \\ &= |\det(v_1 - v_0, \dots, v_n - v_0)| \cdot \mathrm{vol}(\Delta) = \frac{|\det(v_1 - v_0, \dots, v_n - v_0)|}{n!}\end{aligned}$$

Exercise 3 ($\star$)

The purpose of this exercise is to review the steps of the ellipsoid method in greater detail for the case where we want to use it to compute a max-weight matching of $G = (V, E)$ with positive integer weights $w \in \mathbb{Z}_+^{|E|}$.
Let μ_w be the maximum weight of a matching and let $P_G = \mathrm{conv}\{\chi^M : M \text{ matching}\}$ be the matching polytope.
Show that

(i) $\mu_w = z^* + 1$, where z^* is the largest integer such that $P_G \cap (w^T x \geq z + \frac{1}{2})$ has positive volume.

(ii) if z is an integer such that $P_G \cap (w^T x \geq z + \frac{1}{2})$ has positive volume, then $\mathrm{vol}\left(P_G \cap (w^T x \geq z + \frac{1}{2})\right) \geq \frac{1}{n!(2\|w\|)^n \cdot n^{n/2}}$

(iii) the ellipsoid-method needs a polynomial (in encoding length) number of iterations, to find out whether $P_G \cap (w^T x \geq z)$ has positive volume for some integer z. (Find a suitable initial ellipsoid and estimate number of iterations)

(iv) Complete the description of an algorithm using the ellipsoid method that computes μ_w (binary search) and a point in $P_G \cap (w^T x \geq \mu_w - \frac{1}{2})$. You can assume an oracle for the separation problem.

Solution

(i) Note first that $P_G \cap (w^T x \geq z + \frac{1}{2})$ obviously has zero volume for $z = \mu_w$. On the other hand, the volume must be positive in the case of $z = \mu_w - 1$, since P_G is integral.

(ii) Clearly, we can concentrate on a lower bound on the volume in the case of $z = z^*$, since otherwise the volume is only bigger. Observe first that the height of the strip between the two hyperplanes $w^T x \geq z^* + \frac{1}{2}$ and $w^T x \geq \mu_w$ is exactly $\frac{1}{2\|w\|}$. Observe further that $P_G \cap (w^T x \geq z^* + \frac{1}{2})$ is a simplex that is clearly contained in the ball of radius $\sqrt{n}$. Thus if we scale the original simplex P_G by the height of the strip and the radius of the ball, we obtain a simplex contained in $P_G \cap (w^T x \geq z^* + \frac{1}{2})$. The unit simplex is definitely contained in the original simplex P_G. In total we obtain

$$\text{vol}\left(P_G \cap (w^T x \geq z + \frac{1}{2})\right) \geq \text{vol}\left(\frac{1}{2\|w\|} \cdot \frac{1}{\sqrt{n}} P_G\right) = \frac{\text{vol}(P_G)}{(2\|w\|)^n \cdot n^{n/2}} \geq \frac{1}{n!(2\|w\|)^n \cdot n^{n/2}}$$

(iii) As mentioned in the previous point, the ball of radius $\sqrt{n}$ around the origin certainly contains the unit cube and thus P_G. Its volume is $\text{vol}(E_{init}) = \text{vol}(\sqrt{n} B_n)$. We choose as a stopping criterion $L = \frac{1}{n!(2\|w\|)^n \cdot n^{n/2}}$. From the lecture we know that the number of iterations of the Ellipsoid method is bounded by

$$n \cdot \log\left(\frac{\text{vol}(E_{init})}{L}\right) \leq n \cdot \log\left(\frac{\frac{1}{\pi n n^{n/2}}(\frac{2\pi e}{n})^{n/2}}{\frac{1}{n!(2\|w\|)^n \cdot n^{n/2}}}\right) = n \cdot \log\left(\frac{n!}{\pi n}\left(2\|w\|\sqrt{\frac{n}{2\pi e}}\right)^n\right)$$

Since $\log n! = O(n \log n)$, we obtain that the number of iterations is $O\left(n^2 \log n (\log n + \log w_{max})\right)$, thus polynomial.

(iv) Here is the algorithm to find an almost optimal point in P_G in polynomial time:

1. Set $L = 0$, $U = \sum_{e \in E} w_e$ and $x^* = 0$.
2. Let $z = \lfloor \frac{U+L}{2} \rfloor$. **If** $U \leq L + 1$ **then** let $\mu_w = U$ and **return** (x^*, μ_w)
3. **If** Ellipsoid-method$(P_G \cap (w^T x \geq z + \frac{1}{2})) = \emptyset$ **then**
   set $U = z$ and **go to** 2.
   else Set x^* to the point output by the Ellipsoid method.
   Set $L = z$ and **go to** 2.

This algorithm obviously needs $\log_2(w_{\max}) + 2\log_2(n)$ many calls of the Ellipsoid method. From part (iii), we know that each call to the ellipsoid method implies at most a polynomial number of iterations. In each iteration, the separation oracle (given as input) is asked once and depending on its answer, a new ellipsoid is computed in polynomial time. In total, this algorithm computes an almost optimal point in the maximum weight matching polytope in polynomial time, using a polynomial number of oracle calls for the separation problem.

Note that the size requirements of the whole procedure can be shown to be polynomially bounded, but at this point, we refer to the literature for a detailed analysis.

Exercise 4
Describe an algorithm that given as input

- an integral polyhedron $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$, where $A \in \mathbb{R}^{m\times n}$ and $b \in \mathbb{R}^m$,
- an objective function vector $c \in \mathbb{Z}^n$,
- the optimal objective function value $z = \max\{c^T x \mid x \in P\}$, and
- a feasible point $x^\star \in P$ with $z - \frac{1}{2} \leq c^T x^\star \leq z$

computes an inclusion-wise minimal optimal face $F = \{x \in \mathbb{R}^n \mid A'x = b'\}$ of P. The running time of your algorithm shall be bounded by a polynomial in n and m.

Solution

Here is an algorithm that finds a minimal optimal face (where $\overline{L} := \{1, \ldots, m\}\setminus L$):

(i) Initialize $L = \emptyset$.
(ii) Compute the solution space $V = \operatorname{span}(v_1, \ldots, v_k)$ of $A_L x = 0$ using Gaussian elimination. Set $V = \{c\}$ if $L = \emptyset$.
(iii) **For** $v \in \{\pm v_1, \ldots, \pm v_k\}$ **do**:
  If $c^T v \geq 0$ **then** compute $\lambda^* = \sup\{\lambda\colon A_{\overline{L}}(x^* + \lambda v) \leq b_{\overline{L}}\}$
    If $\lambda^* < \infty$ **then** $x^* = x^* + \lambda^* v$, $L = L \cup \{j\colon a_j^T x^* = b_j\}$. **Go to** (ii)
(iv) return $F = \{x \in \mathbb{R}^n\colon A_L x = b_L\}$

Show correctness of the algorithm:

a) F is a minimal face.
Since the LP is bounded, we can not reach step (iv) if there is $v \in V$ with $c^T v > 0$, because this implies $\lambda^* < \infty$. In other words, we reach step (iv) only if for all $v \in V$ both $c^T v = 0$ and $\lambda^* = \infty$ holds. It implies $F = \{x^* + v\colon v \in V\} \subseteq P$ and thus F is a minimal face of P.

b) F is an optimal face.
Note that the objective value of x^* is non-decreasing throughout the algorithm and $A_L x^* = b_L\}$ is maintained. Let $\hat{x} \in F$ be an integral point (P is integral!) and write it as $\hat{x} = x^* + \hat{v}$ for some $\hat{v} \in V$. We obtain

$$z \geq c^T \hat{x} = c^T x^* \geq z - \frac{1}{2}$$

and thus $z = c^T \hat{x}$ by the integrality of z, c and $\hat{x}$.

c) The number of iterations of the algorithm is polynomial.
The dimension of V reduces with each jump back to (ii) (except for the first one!) by at least one. Thus the running time is at most $O(n \cdot (n^2 m + n \cdot m \cdot n))$.