MA4701. Optimización Combinatorial. 2013.
Profesor: José Soto
Escriba(s): Sebastián Urzúa y Hugo Carrillo.
Fecha: 18 de Noviembre de 2013.

Cátedra 23

1. Optimización sobre envolturas convexa de vértices del cubo.

En esta clase se verá una aplicación del método de la elipsoide.

Teorema 1. *Si $S \subseteq \{0,1\}^d$, $P = conv(S)$, P es de dimensión completa y w es función de pesos ($w \in \mathbb{Z}^d$), entonces podemos minimizar $w^\top x$; $x \in P$ con $O(d^3 \log(d \cdot w_{max}))$ llamadas a un oráculo de separación para P. Aquí se anotó $w_{max} =$* $\max |w_x|$.

Demostración. Veamos primero que revisar la factibilidad en P es polinomial:
• $P \subseteq [0,1]^d \subseteq B^* := B\Big(\Big(\frac{1}{2}, \frac{1}{2}, \ldots, \frac{1}{2}\Big), \frac{\sqrt{d}}{2}\Big)$, con $\Big(\frac{1}{2}, \frac{1}{2}, \ldots, \frac{1}{2}\Big) \in \mathbb{R}^d$.
• P es de dimensión completa, luego debe contener $d+1$ vértices afínmente independientes (es decir, que no están en el mismo hiperplano), digamos, $\text{conv}(v_0, v_1, \ldots, v_d) \subseteq P$, luego

$$\text{Vol}(\text{conv}(v_0, \ldots, v_d)) = \frac{1}{d!}\text{Vol}\Big(\text{Paralelógramo generado por } (v_i - v_o):\ i = 1, \ldots, d\Big) = \frac{1}{d!} \cdot 1$$

El método elipsoidal encuentra un punto factible en P usando

$$O\left(d \log\left(\frac{\text{Vol}(B^*)}{\text{Vol}(P)}\right)\right) = O(d \log(d^{d/2} \cdot d!)) = O(d^3 \log(d)),$$

llamadas al oráculo de separación de P, pues $\log(d!) = O(d \log(d))$.

Veamos ahora como optimizar sobre P. Para todo $k \in \mathbb{Z}$, definamos $P_k = P \cap \{x;\ w^\top x \leq k + 1/2\}$. Si x^* es vértice de P, entonces $x^* \in \{0,1\}^d$, y por lo tanto $|w^t x| \leq w_{max} d$. Queremos encontrar el menor k tal que $P_k \neq \emptyset$. El término "$+1/2$" nos asegura que si $P_k \neq \emptyset$, entonces P_k es de dimensión completa.

Notemos primero que el valor buscado, $k \in [-w_{max} d, w_{max} d]$. Luego podemos encontrar dicho valor usando búsqueda binaria. Esto toma a lo más $\lceil log(2w_{max} d + 1\rceil = O(log(w_{max} d))$ chequeos de factibilidad en poliedros P_k.

Para concluir, debemos estimar el tiempo que le toma al método elipsoidal revisar factibilidad en P_k. Para esto necesitamos cotas superiores e inferiores en el volumen de P_k.

Cota superior de volumen. $P_k \subseteq B^* := B\Big(\Big(\frac{1}{2}, \frac{1}{2}, \ldots, \frac{1}{2}\Big), \frac{\sqrt{d}}{2}\Big)$.

Cota inferior. A diferencia de P, P_k no necesariamente contiene un simplex (envoltura convexa de $d+1$ vectores afinmente independientes) con vértices enteros. Sin embargo, si $P_k \neq \emptyset$, entonces contiene al menos 1 punto de coordenadas enteras. Sea v_0 dicho punto. Sean $v_1, \ldots, v_d \in S$ tales que $(v_0, \ldots, v_d)$ son afínmente independientes (es decir, no están todos en un mismo hiperplano, o dicho de otro modo, $(v_i - v_0)_{i=1}^d$ es l.i.).

Para $\varepsilon = \dfrac{1}{4w_{max} d}$, definimos

$$u_i = \begin{cases} v_i & \text{si } v_i \in P_k, \\ v_0 + \varepsilon(v_i - v_0) & \text{si } v_i \notin P_k, \end{cases}$$

Recordemos que si $v_i \notin P_k$, entonces $w^\top u_i > k + \dfrac{1}{2}$. Eso implica que $w^\top u_i = w^\top v_0 + \varepsilon w^\top (v_i - v_0) \leq k + \frac{1}{4w_{max} \cdot d} \cdot 2d \cdot w_{max} \leq k + \frac{1}{2}$, ya que $(v_i - v_0) \in \mathbb{Z}^d \cap [-2d \cdot w_{max}, 2d \cdot w_{max}]$, y luego para todo i, $u_i \in P_k$.

Por lo tanto, $\text{Vol}(P_k) \geq \text{Vol}(\text{conv}(u_0, \ldots, u_d)) = \dfrac{1}{d!} Vol\Big(\text{paralelepípedo } (\{u_i - u_0; i = 1, \ldots, d\})\Big)$. Además, $u_i - u_0 = \alpha(v_i - v_0)$, con $\alpha_i \in \{1, \varepsilon\}$. De esto se concluye que $\text{Vol}(P_k) \geq \dfrac{1}{d!}\varepsilon^d = \dfrac{1}{d!} \cdot \left(\dfrac{1}{4w_{max} d}\right)^d$.

De lo anterior, se concluye que el número de iteraciones del método elipsoidal necesarias para determinar factibilidad en P_k es

$$O\left(d\log\left(\frac{\mathrm{Vol}(B^*)}{\mathrm{Vol}(P_k)}\right)\right) = O(d\log(d^{d/2}\cdot d!\cdot 4^d\cdot w_{max}^d\cdot d^d)) = O(d^2\log(d\cdot w_{max})).$$

Juntando lo anterior tenemos que para encontrar el mínimo k tal que $P_k \neq \emptyset$, debemos usar $O(d^2\log^2(d\cdot w_{max}))$.

Pero esto sólo nos devuelve el valor del mínimo k y un vector en P_k. Para encontrar el verdadero punto x^* óptimo, lo que hacemos es recursión en $P\cap\{x\in\mathbb{R}^d\colon w^\top x = k\}$. Usando que $w^T x = k$ podemos despejar una coordenada en función de las demás, luego la dimensión se reduce en 1. En el problema reducido es posible revisar que la dimensión es completa o, nuevamente encontrar una coordenada que se escriba en función de las anteriores (detalles omitidos). Una vez que estamos en el caso de dimensión completa, repetimos el argumento. Repitiendo esto d veces llegamos a un vértice de P óptimo.

En conclusión, el tiempo total que requiere el algoritmo es de $O(d^3\log^2(d\cdot w_{max}))$ llamadas al oráculo.

$\square$