Fall 2014 Randomized Algorithms Oct 15, 2014

Lecture 4

Prof. Friedrich Eisenbrand *Scribes: Ashkan Norouzi-Fard*

In this lecture, first we discuss about linear programming and we introduce a randomized linear time algorithm for solving an LP. After that we present an upper bound on the diameter of a polyhedron.

Linear Programming

In this section we present **Clarkson(1)** [1] algorithm using the **Clarkson(2)** [1] algorithm introduced in previous lecture. First let us review **Clarkson(2)** algorithm that we saw.

Clarkson(2) algorithm

We can now present the **Clarkson(2)** algorithm

Algorithm 1 Clarkson(2)

Input: A multiset H containing the n constraints of the LP
Output: The solution x^* to the LP

```
Set r = 6 · d^2
repeat
    Pick R ∈ (H choose r) uniformly at random
    Compute x*_R and V_R = {h ∈ H : x*_R violates h}
    if |V_R| ≤ (1/(3d)) μ(H) then
        H = H + V_R (double the occurrence of each h ∈ V_R in H.
    end if
until V_R = ∅
return x*_R
```

Set $r = 6 \cdot d^2$
repeat
  Pick $R \in \binom{H}{r}$ uniformly at random
  Compute x^*_R and $V_R = \{h \in H \ : \ x^*_R \text{ violates } h\}$
  if $|V_R| \leq \frac{1}{3d}\mu(H)$ **then**
    $H = H + V_R$ (double the occurrence of each $h \in V_R$ in H.
  end if
until $V_R = \emptyset$
return x^*_R

Clarkson(1) algorithm

Clarkson(1) is similar to the previous algorithm. We are given an LP with n constraints and in d-dimensions. We assume that the given LP is finite and has a unique feasible solution. These two assumptions can be achieved without loss of generality. Let us now state the algorithm.

Algorithm 2 Clarkson(1)

Input: A multiset H containing the n constraints of the LP
Output: The solution x^* to the LP

if $|H| \leq 9d^2$ **then**
return Clarkson(1)
end if
Set $r \leftarrow \lfloor d\sqrt{n} \rfloor$, $G \leftarrow \emptyset$
repeat
  Pick $R \in \binom{H}{r}$ uniformly at random
  $r \leftarrow$ Clarkson(1)$(R \cup G)$
  $V \leftarrow \{h \in H|$ h violates r $\}$
  if $|V| \leq 2\sqrt{n}$ **then**
    $G \leftarrow V \cup G$
  end if
until $V = \emptyset$
return r

Analysis of Clarkson(1)

According to sampling lemma (discussed in previous lecture), the number of violated constraints is $\sqrt{n}$ in expectation in each iteration. Therefore, the "if" condition becomes true with probability at least 1/2 and one of the constraints of the optimal basis enters G. Moreover, $|V| \leq 2\sqrt{n}$ and the number of times enter "if" loop is exactly d, therefore, $|R \cup G| \leq d \cdot \sqrt{n} + 2d \cdot \sqrt{n} = 3d \cdot \sqrt{n}$. The running time would be:

$$T_1(n) \leq 2d[O(dn) + T_2(3d\sqrt{n})] = O(d^2 n) + d^4 \sqrt{n} \log(n) + d^{O(d)} \log(n)$$

Note that $d^{O(d)}$ can be improved to $e^{\sqrt{d \log(d)}}$ by using an different algorithm to compute x_R^* in **Clarkson(2)**. For details please refer to [2]. Currently we assume this is computed using the simplex algorithm.

In the next part of the lecture, we introduce LP type problem and use Clarkson(1) algorithm to solve them.

LP type problem

Assume we are given n points in d dimensions. How can we find the smallest ball including all of them? We can not use LP to solve this problem, instead we will introduce LP type problem and us the to solve this problem. In LP type problem, we are given a set of elements S and a function $w : 2^S \rightarrow R \cup \{\pm\infty\}$ satisfying the following two axioms:

- Monotonicity: for any $A \subseteq B \subseteq S$, then $w(A) \leq w(B)$
- Locality: for any $A \subseteq B \subseteq S$ and $h \in S$ and $w(B + h) > w(B)$, then $w(A + h) > w(A)$.

The goal is to compute a minimal subset B_S of S, having $w(B_S) = w(S)$. Let us first define the basis and basis of any set $A \subseteq S$.

Definition 1 *$B \subseteq A$ is a basis of $A \subseteq S$, if $w(B') < w(B) = w(A) \quad \forall B' \subset B$.*

Definition 2 *B is a basis if $w(B') < w(B) \quad \forall B' \subset B$.*

We represent an LP type instance using following two oracle (since function w might be exponentially large).

- Violation Oracle: for a given $h \in S$ and $A \subseteq S$, outputs whether $w(A + h) > w(A)$ or not.
- Basis Oracle: for a given $h \in S$, and basis B, output basis of $B \cup \{h\}$

Clarkson(1) works for this problem too, the only difference is that instead of mathematical operation we would have oracle calls. LP type problem enables us to solve minimum enclosing ball problem.

Diameter of Polyhedron

A (d, n)-polyhedron is a polyhedron in d-dimension with n facet. For example, a cube is a $(3, 6)$-polyhedron. Here is a formal definition of (d, n)-polyhedron.

Definition 3 *A polyhedron P defined by $\{H_1, ..., H_n\} = H$ is called (d, n)-polyhedron if P satisfying the following conditions:*

- *full dimensional in $\mathbb{R}^d$*
- *$H' \subseteq H$, polyhedron defined by H' is not equal to P.*

A (d, n)-polyhedron has n facets $F_i = P \cap H_i$ for $i = 1, ..., n$, where each of the facets is $(d-1)$-dimensional and can be expressed using at most $n-1$ constraints. We say $v \in P$ is a vertex of P if there is a half-space H with $P \cap H = \{v\}$. Two vertices v, w are adjacent if the line segment between them forms a face of dimension 1 for P. Define G to be the graph that naturally arises from this description of the polyhedron P: nodes corresponding to vertices in P and edges corresponding to adjacent vertices in P. Let $\delta(u, v)$ be the shortest path between two vertices u, v in G. Moreover, let $\text{diam}(P) = \max_{u,v} \delta(u, v)$ and $\Delta(d, n) = \max_P \text{ diam}(P)$ where P is a (d, n)-polyhedron.

Theorem 4 *(Kalai-Kleitman) [3].* $\Delta(d, n) \leq n^{log(d)+2}$

In order to prove the above theorem, we prove the following lemma. One can see by this lemma proves the theorem by using induction on $n + d$.

Lemma 5 $\Delta(d, n) \leq \Delta(d-1, n-1) + 2\Delta(d, \lfloor n/2 \rfloor) + 2$*, for any* $1 \leq d \leq n$.

Proof Take any polyhedron P and two vertices u, v in it. Define k_v maximal integer such that all paths of length k_v from v reach no more than $\lfloor n/2 \rfloor$ facets. Any path from v with length $k_v + 1$ shares a facet with any path from u with length $k_u + 1$, so distance from u to v is at most $k_u + 1 + k_v + 1 + \Delta(d-1, n-1)$. Know we need to prove that $k_v \leq \Delta(d, \lfloor n/2 \rfloor)$ for any vertex in $v \in P$. Let $P_v \subset P$ be the polyhedron define by the facets reachable from v with a path of length k_v. P_v is a $(d, \lfloor n/2 \rfloor)$-polyhedron so $k_v \leq \Delta(d, \lfloor n/2 \rfloor)$. ■

Moreover, Kalai and Kleitman achieved $\Delta(d, n) \leq n2^d$ [3]. An interesting open problem is to find a polynomial bound or a bound similar to $nd^{\log(d)}$

References

[1] Kenneth L. Clarkson. Las vegas algorithms for linear and integer programming when the dimension is small. *J. ACM*, 42(2):488–499, March 1995.

[2] Bernd Gärtner and Emo Welzl. Linear programmingrandomization and abstract frameworks. In *STACS 96*, pages 667–687. Springer, 1996.

[3] Gil Kalai and Daniel J Kleitman. A quasi-polynomial bound for the diameter of graphs of polyhedra. *Bulletin of the American Mathematical Society*, 26(2):315–316, 1992.