Ellipsoid Algorithm

Siddharth Bhat
20161105

***Abstract*—We motivate the Ellipsoid algorithm, discuss its original theoretical importance, and remark on its practical efficiency.**

***Index Terms*—component, formatting, style, styling, insert**

I. Introduction

The ellipsoid algorithm was discovered in Naum Z. Shor. Later, Leonid Genrikhovich Khachiyan proved that the algorithm runs in polynomial time. This was a breakthrough in the theory of linear programming, which proved that solving LP's is in `PTIME`.

II. Description the algorithm: Checking for non-emptiness

III. Using non-emptiness to solve LP's

So far, all we can do using the ellipsoid algorithm is to check if some system of equations $Ax \leq b$ is *non-empty*. In other words, we can check the non-emptiness of a given polyhedron. Here, we will describe how to use this to solve *optimisation problems*.

Consider a linear program and its dual:

$$x, c \in \mathbb{R}^{n \times 1} \quad A \in \mathbb{R}^{m \times n} \quad b, y \in \mathbb{R}^{m \times 1}$$
$$P_{primal} \equiv \underset{x}{\text{maximise}}\ c^T x \text{ subject to } Ax = b$$
$$P_{dual} \equiv \underset{y}{\text{minimise}}\ b^T y \text{ subject to } A^T y \geq c$$

Let $x^\star$ be the optimal value of x for P_{primal}, and $y^\star$ be the optimal value of y for P_{dual}. From strong duality, we know that the value of $c^T x^\star = b^T y^\star$.

So, we can create a *combined* linear program, whose feasibility will force us to provide a point such that $c^T x = b^T y$. That is, we create a new polyhedra Q defined by the equations:

$$Ax \leq b \quad A^T y \geq c \quad c^T x = b^T y$$

Now, if a feasible point $(x_0, y_0) \in Q$, then it must be the case that $Ax_0 = b$, $A^T y_0 \geq c$, and $c^T x_0 = b^T y_0$. At this point, strong duality tells us that $(x_0, y_0) = (x^\star, y^\star)$.

Hence, we can find the optimal value of the linear program by evaluating $c^T x_0$.

Thus, the ellipsoid algorithm can be used to solve for the optimality of a linear program, by starting from a non-emptiness check! This is beautiful, and proves a deep result of LP's: A certificate of non-emptiness is as good as a ceritificate of optimality.