The Complexity of the Cutting Plane Method

Karthekeyan Chandrasekaran
Mentor: Santosh Vempala

1 Introduction

The cutting plane algorithm is a commonly used method for optimization in the field of integer programming. An integer programming problem is specified by

$$\begin{aligned} \max\ c^T X & \text{ subject to} \\ AX &\leq b \\ X &\in \mathbb{Z}^n. \end{aligned}$$

The cutting plane method solves a linear relaxation of the problem obtained by dropping the integrality constraint. Given an integer programming problem, it proceeds as follows: If the optimum solution found is not integral, then a cut inequality separating the optimum from all integer solutions is added to the LP and the LP is solved again; this two-step procedure is repeated till an integral solution is reached. Gomory showed that cuts of the form

$$\lfloor \lambda^T A \rfloor X \leq \lfloor \lambda^T b \rfloor \text{ for some } \lambda \geq 0$$

always exist if the current optimum is not integral and one such cut can be found in polynomial time [2]. We will denote the cuts that are generated by this method as Gomory cuts.

There exist input instances for which the number of Gomory cuts needed to identify an integer optimum using the cutting plane algorithm is arbitrarily large [5]. Yet, cutting plane algorithms are widely-used in practice in popular solvers to solve integer programming problems. Through my research, I would like to explore the complexity of cutting plane algorithms for two candidate problems: (1) the minimum cost perfect matching problem and (2) probabilistic instances of the integer programming problem.

2 The minimum-cost perfect matching problem

Given a graph G, a set F of pairwise non-adjacent edges in G is called a perfect matching if for every vertex $v \in V$, there exists an edge $f \in F$ that is adjacent to v. The input to a minimum cost perfect matching problem consists of a graph $G = (V, E)$ with edge costs $c : E \to \mathbb{R}$. The objective is to find a minimum-cost perfect matching, if one exists. Edmonds showed that the perfect matching polytope P (describing the convex hull of characteristic vectors $\psi \in \{0,1\}^{|E|}$ of perfect matchings in G) is specified by the following set of inequalities [1]:

$$\begin{aligned} X_e &\geq 0\ \forall e \in E && \text{(non-negativity constraints)} \\ X(\delta(v)) &= 1\ \forall v \in V && \text{(degree constraints)} \\ X(\delta(U)) &\geq 1\ \forall U \subseteq V, |U| \geq 3, |U| \text{ odd} && \text{(blossom inequalities).} \end{aligned}$$

The LP to solve the minimum-cost perfect matching problem is thus specified by $\min \sum_e c(e) X_e : X \in P$. Even though the LP is exponential in size, there exists an efficient separation oracle for the matching polytope [4] and hence the LP can be solved in polynomial time (by the Ellipsoid algorithm [4]). We denote the polytope specified by only the non-negativity and degree constraints as the *fractional matching polytope* $FP(G)$.

Grötschel and Holland [3] suggested a cutting plane algorithm for this problem: solve the LP $\min \sum_e c(e) X_e : X \in FP(G)$. If the solution is not integral, add a blossom inequality as a cutting plane and repeat. They showed experimental evidence to suggest that the number of cuts needed to arrive at an integral solution for random edge weights is small. My goal is to show that this cutting plane algorithm terminates after the addition of a polynomial($|V|$) number of cuts. In order to show a polynomial bound on the number of cuts, we need to show that the LP optimum makes progress towards an integer solution.

Any extreme point solution to $FP(G)$ is half-integral and thus its support consists of disjoint edges and disjoint odd cycles. The natural cuts to add are (a subset of) the cycles obtained. In work with L. Vegh and S. Vempala, I have shown that adding an arbitrary subset of these cycles keeps the next solution half-integral. I would like to show that this process can be continued (with a natural extension in later steps) so that (a) the half-integrality property is maintained and (b) the number of odd cycles in the support is nondecreasing and decreases every $O(n)$ steps. Proving this would give the first rigorous bound on an efficient cutting-plane method for a well-known problem.

3 A Randomized Cutting Plane Algorithm

Gomory's cuts are obtained by taking a particular linear combination of the facets defining the current basic feasible solution and rounding down the coefficients of the resulting hyperplane. Gomory showed that the number of cuts needed in the cutting plane algorithm implementing Gomory cuts is at most 2^n for arbitrary binary integer programs [2].

A valid cutting plane (including Gomory's cuts) can be obtained by considering a hyperplane that is tangential to the given polytope at the current extreme point solution and rounding down the coefficients of this hyperplane. This immediately suggests a random choice for the cutting planes: take a random combination of the facets defining the current basic feasible solution and apply the round-down technique. I would like to investigate if the complexity of the cutting plane algorithm improves if it implements random cuts.

Formally, the random cuts are obtained as follows: the current basic feasible fractional solution X^* is specified by n facets. Let I denote the set of indices of the facets of the polytope that define X^*. Then, $X^*_j = ([A]_I^{-1}[b]_I)_j$ for $j \in I$ and $X^*_j = 0$ for $j \notin I$. Consider the lattice $L^* = \{y \in \mathbb{Q}^n : y^T[A]_I \in \mathbb{Z}^n\}$. Let $S = \{y \in \mathbb{Q}^n : y^T[b]_I \notin \mathbb{Z}\}$. After obtaining $y \in L^* \cap S$, it is straightforward to obtain a cut that separates X^* from all integer solutions by the round-down technique. The set L^* is the dual lattice of the lattice generated by the columns of $[A]_I$. If X^* is not integral, then for every basis $\{y_1, \cdots, y_n\}$ of L^*, at least one of $y_i^T b \notin \mathbb{Z}$ among $i \in \{1, \cdots, n\}$. Further, if λ_i is chosen uniformly in $\{0, 1\}$ for each $i \in \{1, \cdots, n\}$, then with probability at least $1/2$, $\sum_{i=1}^{n} \lambda_i y_i \in L^* \cap S$. Thus a random $\{0, 1\}$-combination of the rows of $([A]_I^{-1})^T$ leads to a cut separating the current LP optimum with probability at least $1/2$. (Instead of a random $\{0, 1\}$-combination, if one considers the basis vector y_i such that $y_i^T b \notin \mathbb{Z}$, then it is equivalent to the Gomory cut).

I would like to address the following questions about the cutting plane algorithm using random cuts.

1. If the cutting plane algorithm uses random cuts, then what is the expected number of cuts needed to identify an integer solution for arbitrary binary integer programs? Can we show a sub-exponential bound?

2. Is there a natural distribution on the input instances of the integer program for which the cutting plane algorithm using random cuts converges to an integer solution using polynomial number of cuts in expectation?

3. Consider the random polytope determined by m random tangential hyperplanes to a n-dimensional sphere of radius R centered around an arbitrary point $X_0 \in \mathbb{R}^n$. In joint work with S. Vempala, I have shown that for $R = \Omega(\sqrt{\log(2m/n)})$, such a random polytope contains an integer point with high probability. For a fixed direction c, what is the expected number of cuts needed in the cutting plane algorithm using random cuts to find an integer optimum in the random polytope? For $R = O(\sqrt{\log(2m/n)})$ and a fixed direction c, what is the expected number of cuts to find an optimum integer point along c or to certify that the polytope does not contain an integer point?

References

[1] J. Edmonds. Maximum matching and a polyhedron with 0,1-vertices. *Journal of Research of the National Bureau of Standards(B)*, 69:125–130, 1965.

[2] R. Gomory. An algorithm for integer solutions to linear programs. In *Recent advances in Mathematical Programming*, pages 269–302, New York, NY, USA, 1963. McGrawHill.

[3] M. Grötschel and O. Holland. Solving matching problems with linear programming. *Mathematical Programming*, 33:243–259, 1985. 10.1007/BF01584376.

[4] M. Grötschel, L. Lovász, and A. Schrijver. The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica*, 1:169–197, 1981. 10.1007/BF02579273.

[5] A. Schrijver. *Theory of linear and integer programming*. Wiley-Interscience series in discrete mathematics and optimization. Wiley, 1998.