EXPRESSING COMBINATORIAL OPTIMIZATION PROBLEMS BY LINEAR PROGRAMS

(Extended Abstract)

Mihalis Yannakakis

AT&T Bell Laboratories
Murray Hill, NJ 07974

1. INTRODUCTION

Many combinatorial optimization problems call for the optimization of a linear function $c'x$ over a discrete set S of solution vectors. For example, in the case of the Traveling Salesman Problem (TSP), $x=(x_{ij})$ is an $\binom{n}{2}$-dimensional variable vector whose coordinates correspond to the edges of the complete graph K_n on n nodes, c is the vector of inter-city distances, and $S \subseteq \{0,1\}^{\binom{n}{2}}$ is the set of characteristic vectors of the tours of n cities (considered as subsets of the edges of K_n). In the case of the weighted (perfect) matching problem, S is the set of characteristic vectors of the perfect matchings of K_n (n even). These problems are equivalent to: min(max)$c'x$ subject to $x \in$ *convex hull*(S). The convex hull of the solutions is a polytope, which takes its name from the corresponding problem: the *TSP* (resp. *matching*) *polytope.* Analogous polytopes have been defined and studied extensively for other common problems: (weighted) bipartite perfect matching (*assignment* polytope), maximum independent set and clique problem (*vertex-packing* and *clique* polytopes), etc.

Optimizing a linear function over a polytope is a Linear Programming problem. Typically however, polytopes associated with most combinatorial problems (the assignment polytope is one of the exceptions) have an exponential number of facets. Thus, any linear programming formulation in the variables x that defines the polytope has exponential size, and one cannot apply an LP algorithm directly. However, it may be possible (and in some cases true) that the size can be drastically reduced if extra variables and constraints are used.

Let P be a polytope in the set of variables (coordinates) x. We say that a set of linear constraints $C(x,y)$ in the variables x plus new variables y *expresses* P if the projection of the feasible space of $C(x,y)$ on x is equal to P; i.e., $P = \{x :$ there is a y such that (x,y) satisfies $C\}$. Equivalently, C expresses the polytope P iff optimizing any linear function $c'x$ over P is equivalent to optimizing $c'x$ subject to C. We are interested in the question of whether particular polytopes can be expressed by small LP's.

Since Linear Programming is in P, if one could construct a small (polynomial size) LP expressing the polytope of an NP-complete problem, such as the TSP, then it would follow that P= NP. Actually, if anybody believes that P= NP, it is a reasonable approach to try to prove it using Linear Programming, given the fact that the polynomial algorithms for LP are quite hard, and for a long time LP was thought to be outside P. In fact, a recent report was claiming a proof of P= NP this way [S]. The proposed LP for the TSP polytope had n^8 (n^{10} in a revision) variables and constraints. With LP's of this size, it is hard to tell

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.

© 1988 ACM-0-89791-264-0/88/0005/0223 $1.50

what they do or do not express, and clearly, some methodology is needed.

Besides ruling out a possible approach to P= NP, there are several reasons for examining the question of the LP size. There are some problems which we know now to be theoretically in P, but the only method known uses Khachian's Ellipsoid algorithm. The Ellipsoid algorithm for LP has the advantage that it does not require a complete listing of all the (often exponentially many) constraints. It suffices to have a *separation* algorithm: a polynomial time algorithm which, given a point, decides whether it is feasible, and if it is not produces a violated constraint. Given the impracticality of the Ellipsoid algorithm, it would be desirable to replace it with Simplex or Karmakar's algorithm; however, these algorithms need a complete listing of the constraints.

Another use of separation algorithms is in procedures for the TSP based on the polyhedral approach [GP]. Recent progress in this area has increased dramatically the size of instances that can be solved optimally in reasonable time [PR]. The programs are based on a sophisticated combination of branch-and-bound and cutting planes. For cutting planes they use some known simple and "well-behaved" classes of facets. Expressing these facets with a small LP would not imply anything unexpected in theory (such as P= NP), but would permit one to shorcut the repeated generation of cutting planes and solution of the resulting Linear Programs by solving a single LP.

There may be reason also to look into problems that already have good algorithms: It is reported that implementations of Karmakar's algorithm outperform standard procedures for the assignment problem that are based on the Hungarian method, already at about 100 nodes [J]. In view of this, it would be especially interesting to know whether we can express succinctly also the general (nonbipartite) matching polytope.

In Section 2 we show that the matching and the TSP polytopes cannot be expressed by polynomial size *symmetric* LP's. Informally, "symmetric" means that the nodes of the complete graph are treated the same way; see Section 2 for a formal definition. It is not clear what could be gained by treating one node different than another, but of course this still requires a proof. In Section 3 we reduce the minimum size necessary to express a polytope to a concrete combinatorial problem, and point out a relation to communication complexity theory. In Section 4 we examine the vertex packing polytopes of classes of graphs on which the optimization problem can be solved via the Ellipsoid algorithm. We use a result from communication complexity to show that vertex packing polytopes of perfect graphs can be expressed by subexponential Linear Programs.

2. THE MATCHING AND THE TSP POLYTOPES

Although we usually speak of *the* TSP (or matching) polytope, there is one for every size n. Thus also, when we say that an LP expresses such a polytope, we mean again one LP for every n.

A complete description of the matching polytope was found by Edmonds [E]: $\sum_j x_{ij} = 1$ for all i; $\sum_{i\in S,\, j\notin S} x_{ij} \geq 1$ for all odd subsets S of nodes; and $x_{ij} \geq 0$.

A complete description of the TSP polytope is not known (and probably will never be). This polytope has many complex facets [GP, PW, PY]. However, several simple and useful classes of facets have been identified. First are the obvious constraints: $0 \leq x_{ij} \leq 1$, $\sum_j x_{ij} = 2$ for all i. Another easy class are the *subtour elimination constraints* (SECs): $\sum_{i\in S,\, j\notin S} x_{ij} \geq 2$ for all nonempty proper subsets S of nodes. Next come the 2− *matching constraints* followed by further generalizations (see [GP] for a comprehensive treatment).

The subtour elimination constraints can be easily expressed by a polynomial size LP. One way is based on the separation algorithm for these constraints. Viewing the x_{ij}'s as capacities on the edges, the SECs state that the minimum cut in the graph has capacity at least 2. From the max flow - min cut theorem, this can be expressed in the obvious way introducing appropriate flow variables and constraints. A different and less obvious way uses a small portion of the LP described in [S]. Introduce variables v_{ij} and y_{rijk} with the following intended meaning for a Hamilton tour: Orient the tour in one of the two ways. Variable v_{ij} (where i,j is now an *ordered* pair of nodes) has value 1 if the tour traverses the directed edge (i,j), and 0 otherwise; y_{rijk} is 1 if the directed edge (i,j) is the

kth edge of the tour starting from node r, and 0 otherwise. Some constraints that are clearly consistent with this interpretation are: $x_{ij} = v_{ij} + v_{ji}$; $\sum_i v_{ij} = \sum_i v_{ji} = 1$, $\sum_r y_{rijk} = \sum_k y_{rijk} = v_{ij}$; $\sum_i y_{rijk} = \sum_i y_{rji,k+1} = \sum_i y_{jri,n-k}$; $y_{rijk} \geq 0$. It is not clear what exactly these variables and constraints accomplish, but it can be shown that they too imply the SEC's (and in fact make deeper cuts). Although both of these LP's have polynomial size, they are rather large. It would be of interest if there are alternative smaller LP's, as the SECs provide usually good lower bounds, within few percentage points of the true integer optimum [J].

The constraints of the matching polytope look similar to the SECs, apart from the parity of S. In fact, their separation algorithm is a minor modification of the one for the SEC's (the proof is nontrivial - see [PRa]). However, they are apparently harder to express.

A permutation (relabelling) π of the nodes of the complete graph induces also a permutation of the edges and their corresponding variables: x_{ij} is mapped to $x_{\pi(i)\pi(j)}$. It also maps one perfect matching (or tour) into another, and induces a rotation of the coordinates that leaves the matching and the TSP polytopes invariant. We say that a polytope $P(x,y)$ over variables $x = (x_{ij})$ and new variables y is *symmetric* if every permutation π of the nodes can be also extended to the new variables y so that P remains invariant. A LP (set of linear constraints) is called symmetric if its feasible space is. Clearly, if a set of constraints "looks" symmetric (permutation of the variables gives the same LP) then so is its feasible space, but not conversely; a LP that does not look symmetric may describe a symmetric polytope. The assumption of symmetry is a natural one.

Theorem 1. The matching polytope cannot be expressed by a polynomial size symmetric LP. □

We outline the main steps of the proof: First we transform to an LP in standard form (equality constraints plus nonnegativity of variables) which is also symmetric in a bigger space. Then we pick appropriate feasible solutions for the perfect matchings, and show that (as far as these solutions are concerned) every variable "depends" on a fixed number of nodes. Finally, we show that a standard form LP in which every variable depends on a fixed number of nodes (actually, less than a constant fraction) does not express the matching polytope; in fact, there is a feasible solution such that the subgraph formed by the edges (i,j) whose corresponding variable x_{ij} is nonzero does not even have a perfect matching.

Consider the LP's we described for the SEC's of the TSP. It is intuitively clear (from the indices) that for a variable y_{rijk} only the nodes r, i and j are important. Also, a flow variable in the other formulation refers only to the source, the sink, and the two nodes of the corresponding edge. The notion of "depends" above is a formalization of this intuition.

Theorem 2. The TSP polytope cannot be expressed by a polynomial size symmetric LP.

Proof: Consider a graph G with $6n$ nodes which are partitioned into three equal size sets $L = \{l_1, \ldots, l_{2n}\}$, $M = \{m_1, \ldots, m_{2n}\}$, $R = \{r_1, \ldots, r_{2n}\}$. The subsets L, R induce complete subgraphs, and in addition each node m_i is connected to l_i and r_i. Think of the complete graph induced by L as an instance for the matching polytope, and suppose we have a symmetric LP C for the TSP on $6n$ nodes. A permutation of L induces an automorphism of G. Therefore, setting in C all variables x_{ij} to 0 for the missing edges (i,j) we get another LP C' which is symmetric with respect to L. Since the nodes m_i have degree 2, a Hamilton circuit of G consists of their incident edges and perfect matchings from L and R. It follows that C' expresses the matching polytope on $2n$ nodes. □

It follows that a polynomial size symmetric LP cannot imply the 2-matching constraints; also it will have a feasible solution such that the subgraph defined by the edges with nonzero x_{ij} is not Hamiltonian.

A comment on the relationship to the P=NP question. While the construction of a polynomial size LP for the TSP would imply that P= NP, the converse implication does not seem to hold. Rather, P= NP? is equivalent to a weaker requirement of the LP, in some sense reflecting the difference between decision and optimization problems. Say that a polytope Q in variables z_{ij} ($1 \leq i,j \leq n$) is a *Hamilton Circuit* (HC) polytope if it includes the characteristic vectors of Hamiltonian graphs and excludes nonHamiltonian (considered again as subsets of the edges of the complete graph). For example, if an LP expresses the TSP polytope in variables x_{ij}, then the polytope obtained by adding constraints $x_{ij} \leq z_{ij}$ and then projecting the feasible space on z is a HC polytope.

Proposition. NP has polynomial size circuits (respectively, P= NP) if and only if (for every n) there is a polynomial size LP (resp., that can be constructed efficiently) which expresses a HC polytope. □

3. A COMBINATORIAL PARAMETER

When no new variables are allowed there is very little flexibility in expressing a polytope by an LP. If the polytope is full-dimensional, a non-redundant LP (one in which no constraint can be thrown away) must contain exactly one inequality per facet; furthermore, the inequality for each facet is unique up to scalar multiplication. For a lower dimensional polytope, a minimal LP must contain as many equations as the deficit from full dimension, and exactly one inequality per facet; inequalities that define the same facet may differ more substantially though in this case. However, if one may use any new variables and constraints one wishes, there is an unlimited number of possibilities. We provide a combinatorial characterization to get some handle on this problem.

Let P be a polytope in n-dimensional space with f facets and v vertices. Define a matrix SM (for *slack matrix*) for P whose rows correspond to the facets, and the columns correspond to the vertices. Pick an inequality (anyone) for each facet. The ijth entry of SM is the slack of the jth vertex in the inequality corresponding to the ith facet. That is, if the inequality is $a_i' x \leq b_i$ and the jth vertex is x^j,then $SM[i,j] = b_i - a_i' x^j$. Note that SM is a nonnegative matrix.

Theorem 3. Let m be the smallest number such that SM can be written as the product of two nonnegative matrices of dimensions $f \times m$ and $m \times v$. The minimum of the number of variables plus number of constraints over all LP's expressing P is $\Theta(m+n)$. □

For example, the trivial decomposition $SM = I \cdot SM$ (I the $f \times f$ identity matrix) corresponds to the LP that does not use new variables. The theorem remains true if the matrix is augmented with additional rows corresponding to any valid constraints, and additional columns corresponding to any feasible points. In particular, to get a lower bound we may use any valid constraints, and do not need to know a full description of the polytope.

Let us call the smallest number m of the theorem, the *positive rank* of the matrix SM. There are two parts in the definition of this parameter: the linear algebra part and the nonnegativity restriction. If we ignore the second restriction we get simply the rank of the matrix SM. Although typically SM has an exponential number of rows and columns, its rank is always small, less than n. If we ignore the linear algebra part, and just look at the zero-nonzero structure of the matrix, we can view the problem as one of communication complexity. Let FV be the predicate which is 1 (true) of a facet f_i and a vertex v_j if v_j is not on f_i, and 0 (false) otherwise. Consider the following communication problem: there are two sides, one knows a facet f_i and the other a vertex v_j and want to compute FV. The complexity of a protocol is the number of bits exchanged. In a nondeterministic protocol guesses are allowed, and the requirement is that there be at least one successful guess if the predicate is 1. (See [AUY, MS, PS, Y] for more background.)

Corollary. The nondeterministic communication complexity of the predicate FV is a lower bound on the logarithm of the minimum size of an LP expressing the polytope. □

For the matching polytope there is an obvious $4 \log n$ nondeterministic protocol: just guess two edges of the matching that cross the partition of the facet. It is plausible that this may be the best that can be done, but we have not examined this yet. A n^4 lower bound is probably (we believe) far from tight, but would be enough to imply that the direct application of a LP algorithm to general matching could not compete with present combinatorial algorithms.

4. VERTEX PACKING POLYTOPES

The *vertex packing polytope* $VP(G)$ of a graph G is the convex hull of the characteristic vectors of its independent sets of nodes. Note: there is one polytope for every graph. As expected, we do not know full descriptions of these polytopes. However, for some classes of graphs certain simple and natural constraints suffice.

First there are the obvious constraints: $0 \leq x_i \leq 1$ for every node i, and $x_i + x_j \leq 1$ for every edge (i,j) of the graph (1). These constraints describe the polytope $VP(G)$ iff G is bipartite. Another set of constraints follows from the fact that a cycle C with an odd number $2k+1$ of nodes can contain at most k nodes of an independent set: $\sum_{i \in C} x_i \leq (|C|-1)/2$ for all odd cycles C of the graph (2). The constraints (1), (2) describe the vertex packing polytope for so-called $t-$*perfect* graphs. Although there is in general an exponential number of constraints of type (2), there is a good separation algorithm for them, and thus the optimization problem over the polytope defined by (1) and (2) can be solved in polynomial time using the ellipsoid algorithm (see [L]). It is not too hard to show also:

Theorem 4. The polytope defined by constraints (1) and (2) can be expressed by a polynomial size LP (n^2 variables, n^3 constraints). □

Another set of constraints follows from the fact that an independent set can contain at most one node from a clique: $\sum_{i \in K} x_i \leq 1$ for every clique K of the graph (3). Together with the nonnegativity constraints $x_i \geq 0$, these constraints are sufficient to describe the vertex packing polytope of *perfect* graphs. This is a well-studied, rich class of graphs; it includes several natural subclasses (for example, chordal and comparability graphs and their complements). Some basic properties are: the chromatic number is equal to the maximum clique size; every induced subgraph of a perfect graph is also perfect; the complement of a perfect graph is also perfect (see [BC, G] for more information). The maximum (weight) independent set problem can be solved on perfect graphs through a very complex and deep application of the Ellipsoid algorithm [GLS]. It is an important open problem in computational graph theory to find a better algorithm for this problem.

The slack matrix in this case is 0-1, and coincides with the predicate FV: it is 1 (true) of a clique K and an independent set I if $K \cap I = \emptyset$, and 0 otherwise. There is an obvious nondeterministic protocol for the *complementary* predicate $\overline{FV}$: guess the (in fact unique) node in the intersection of K and I. But we do not see any obvious protocol for FV itself (verifying disjointness). Is the nondeterministic communication complexity of the predicate $K \cap I = \emptyset$ superlogarithmic? If this is the case for any particular graphs (not necessarily perfect), then we can deduce a superpolynomial lower bound for the size of LP's expressing Vertex Packing polytopes in general, and then through reducibilities, for the TSP polytope.

The obvious protocol we mentioned for $\overline{FV}$ is *unambiguous*: if $\overline{FV}$ is true, then exactly one guess is successfull.

Lemma. If the unambiguous communication complexity of a predicate is c, then its deterministic complexity is at most c^2. □

The proof is similar to the one in [AUY], that if both a predicate and its complement have nondeterministic complexity c, then the deterministic complexity is at most c^2. Now, for a 0-1 matrix FV, the logarithm of the positive rank is no more than the deterministic (in fact, the unambiguous) communication complexity of FV. As the numbers involved are small (0-1), it follows from the lemma and Theorem 3 that:

Theorem 5. The Vertex Packing polytopes of perfect graphs can be expressed by LP's of size $n^{\log n}$. □

Note that, unlike Theorem 4, this does not mean that the polytope defined by the constraints (3) is expressible by such an LP for general (nonperfect) graphs. In fact, optimizing over (3) is in general NP-hard [GLS], and therefore, it is unlikely that (3) can be expressed by an LP of subexponential size.

5. DISCUSSION

This work is a first step towards a systematic study of expressing combinatorial optimization problems with small Linear Programs; there are clearly many open problems. Some of the more immediate ones are:

1) Find techniques for computing or bounding the positive rank of a matrix.

2) We do not think that asymmetry helps much. Thus, prove that the matching and TSP polytopes cannot be expressed by polynomial size LP's without the symmetry assumption.

3) Can we express the Vertex Packing polytopes of perfect graphs with polynomial size LP's? How small?

REFERENCES

[AUY] A. V. Aho, J. D. Ullman, M. Yannakakis, "On Notions of Information Transfer in VLSI Circuits", Proc. 15th Annual ACM Symp. on Theory of Computing, 133-139, (1983).

[BC] C. Berge, V. Chvatal, Topics on Perfect Graphs, Annals of Disc. Math. 21, North-Holland, 1984.

[E] J. Edmonds, "Maximum Matching and a Polyhedron with 0,1 Vertices", J. Res. Natl. Bureau of Standards B 69, 125-130, (1965).

[G] M. Golumbic, Algorithmic Graph Theory and Perfect Graphs, Academic Press, 1980.

[GLS] M. Grotschel, L. Lovasz, A. Schrijver, "The Ellipsoid Mehtod and its Consequences in Combinatorial Optimization", Combinatorica 1, 169-197, (1981).

[GP] M. Grotschel, M. W. Padberg, "Polyhedral Theory", and "Polyhedral Computations" in E. L. Lawler et al, eds., The Traveling Salesman Problem, John Wiley, 251-360, (1985).

[J] D. S. Johnson, personal communication.

[Ka] N. Karmakar, "A New Polynomial Time Algorithm for Linear Programming", Combinatorica 4, 373-395, (1984).

[Kh] L. G. Khachian, "A Polynomial Algorithm in Linear Programming", Dokl. Akad. Nauk SSSR 244, 1093-1096; English translation: Soviet Math. Dokl. 20, 191-194, (1979).

[L] L. Lovasz, "Vertex Packing Algorithms", Proc. Intl. Col. Automata, Languages and Programming, Springer Verlag, 1-14, (1985).

[MS] K. Melhorn, E. M. Schmidt, "Las Vegas is Better than Determinism in VLSI and Distributed Computing", Proc. 14th Annual ACM Symp. on Theory of Computing, 330-337, (1982).

[PRa] M. Padberg, M. R. Rao, "Odd Minimum Cut-Sets and *b*-Matchings", Math. of Operations Res. 7, 67-80, (1982).

[PRi] M. Padberg, G. Rinaldi, "Optimization of a 532-city Symmetric Traveling Salesman Problem by Branch and Cut", Operations Res. Letters 6, 1-7, (1987).

[PS] C. H. Papadimitriou, M. Sipser, "Communication Complexity", Proc. 14th ACM Symp. on Theory of Computing, 196-200, (1982).

[PW] C. H. Papadimitriou, D. Wolfe, "The Complexity of Facets Resolved", Proc. 26th Annual IEEE Symp. on Foundations of Computer science, 74-78, (1985).

[PY] C. H. Papadimitriou, M. Yannakakis, "The Complexity of Facets (and Some Facets of Complexity)", J. Comp. Sys. Sc. 28, 244-259, (1984).

[S] E. R. Swart, "P= NP", Technical Report, University of Guelph, (1986; revision 1987).

[Y] A. C. Yao, "Some Complexity Questions Related to Distributive Computing", Proc. 11th Annual ACM Symp. on Theory of Computing, 209-213, (1979).