MC.R. Vol. 14, No. 2, May 1989
Printed in U.S.A.

MAXIMIZING CLASSES OF TWO-PARAMETER OBJECTIVES OVER MATROIDS*

R. HASSIN† AND A. TAMIR†‡

Let $M = (N, \mathscr{F})$ be a matroid. Suppose that each element i in N is associated with an ordered pair of rational numbers (a_i, b_i). For each subset S in $\mathscr{F}$ define $A(S) = \sum_{i \in S} a_i$ and $B(S) = \sum_{i \in S} b_i$. Let g be a real convex function defined on R^2. Consider the problem of maximizing $g(A(S), B(S))$ over all bases S of M. We present a polynomial algorithm for this problem when g is a general polynomial. This algorithm is strongly polynomial when the degree of g is at most cubic. Using the latter result we apply appropriate transformations to obtain strongly polynomial algorithms for some cases when g is not polynomial. In particular, we find in strongly polynomial time the minimal cost reliability ratio spanning tree of an undirected graph.

1. Introduction. Let $M = (N, \mathscr{F})$ be a matroid, where N is the ground set and $\mathscr{F}$ is the collection of independent sets. Define $n = |N|$. Suppose that each element $i \in N$ is associated with an ordered pair of rational numbers (a_i, b_i). Let β be the set of bases of M.

Consider, for example, the following maximization problem over β.

$$\max_{S \in \beta} \left\{ \left(\sum_{i \in S} a_i \right)^2 + \left(\sum_{i \in S} b_i \right)^2 \right\}. \tag{1.1}$$

We assume that the problem is specified by an oracle which represents the matroid, and the set of rationals $\{(a_i, b_i)\}$, $i \in N$. Thus, the size of the problem is the sum of the "size" of the oracle and the size of the encoding of the numerical data.

In general, of course, (1.1) cannot be solved by a greedy scheme. However, the following simple procedure can be applied. For each base $S \in \beta$, define $A(S) = \sum_{i \in S} a_i$, $B(S) = \sum_{i \in S} b_i$. Viewed as a function of A and B, the objective in (1.1) is (strictly) convex. Therefore, a maximizer of (1.1) is a base, say S^*, for which $(A(S^*), B(S^*))$ is an extreme point of the convex hull of the set of two dimensional points, $(A(S), B(S))$, $S \in \beta$. S^* can be obtained by generating all the extreme points of the above hull. Such a procedure is polynomial if we can efficiently produce the extreme bases or at least the potential candidates for optimality. An extreme point (A, B) is characterized by some real γ, such that

$$\max_{S \in \beta} \{ A(S) + \gamma B(S) \} = A + \gamma B, \quad \text{or} \quad \max_{S \in \beta} \{ -A(S) + \gamma B(S) \} = -A + \gamma B$$

In general, if p is an upper bound on the number of extreme points, then all these points can be generated by applying the greedy algorithm $O(p)$ times, [CAN, G]. It is

*Received October 2, 1986; revised January 21, 1988.
AMS 1980 subject classification. Primary: 05B35. Secondary: 90C35.
IAOR 1973 subject classification. Main: Graphs. Cross references: Networks.
OR/MS Index 1978 subject classification. Primary: 484 Networks/Graphs/Flow Algorithms.
Key words. Matroids, strongly polynomial algorithms.
†Tel Aviv University.
‡New York University.

0364-765X/89/1402/0362$01.25

well known, [C], that there are only $O(n^2)$ critical values of γ. They are the solutions to the equations

$$\pm a_i + \gamma b_i = \pm a_j + \gamma b_j, \qquad i, j \in N, \quad i \neq j.$$

Thus, one can identify a set of $O(n^2)$ bases, that will include the solution to (1.1), by applying the greedy algorithm $O(n^2)$ times. In fact a sharper bound, $O(\sqrt{m} \cdot n)$, on the cardinality of the set of extreme points has been established in [G] where m is the rank of M. This bound implies that just $O(\sqrt{m} \cdot n)$ applications of the greedy algorithm suffice for generating all extreme bases. If a (strongly) polynomial time oracle is available for testing independence in the matroid then the above approach will solve (1.1) in (strongly) polynomial time.

Next consider the following maximization problem.

$$\text{(1.2)} \qquad \operatorname*{Max}_{S \in \beta} \left\{ \sum_{i \in S} a_i + \prod_{i \in S} b_i \right\}.$$

Suppose that $b_i > 0$ for all $i \in N$. We then could mimic the above approach by replacing (1.2) by

$$\text{(1.3)} \qquad \operatorname*{Max}_{S \in \beta} \left\{ \sum_{i \in S} a_i + \exp\left(\sum_{i \in S} \log b_i \right) \right\}.$$

Define $b_i' = \log b_i$, $i \in N$, and setting $B'(S) = \sum_{i \in \beta} b_i'$, $S \in \beta$, we note that the objective in (1.3) is a convex function of $A(S)$ and $B'(S)$. Therefore a maximizer of (1.3) is a base S^*, for which $(A(S^*), B'(S^*))$ is an extreme point of the convex hull of the points $(A(S), B'(S))$, $S \in \beta$.

Using the above approach to find the extreme points of this hull will require—(if a greedy algorithm is applied)—determining the signs of expressions of the form

$$\text{(1.4)} \qquad x \log y - z \log w$$

where x, y, z, w are positive rationals.

It has been shown in [CAT] that this task can be done in time which is polynomial in the input lengths of x, y, z and w. This has led to a polynomial (but not strongly polynomial) algorithm to solve (1.2) [CAT]. Note that the problem of minimizing the cost/reliability ratio over all spanning trees of a graph [CAN] can be modeled in terms of (1.2).

As a last example consider the following model,

$$\text{(1.5)} \qquad \operatorname*{Max}_{S \in \beta} \left\{ \prod_{i \in S} a_i + \prod_{i \in S} b_i \right\},$$

where $a_i, b_i > 0$, $i \in N$.

Defining $a_i' = \log a_i$, $i \in N$, and setting $A'(S) = \sum_{i \in S} a_i'$, leads to

$$\text{(1.6)} \qquad \operatorname*{Max}_{S \in \beta} \{ \exp(A'(S)) + \exp(B'(S)) \}.$$

Again, due to convexity an optimal base will correspond to an extreme point of the convex hull of $(A'(S), B'(S))$, $S \in \beta$. One could attempt to solve (1.6) by adapting the approach taken in [CAT] for finding the extreme points. However, such an approach will require a polynomial routine for testing the signs of expressions of the form $(\log x)(\log y) - (\log z)(\log w)$ where x, y, z and w are positive rationals. We are not aware of any polynomial routine for this test, or of any polynomial procedure to solve (1.5).

Motivated by the above examples and others, we develop in this paper a different and unified solution approach which yields strongly polynomial algorithms for a class of two parameter maximization problems defined on matroids that includes the above three problems. The reader is referred to [GLS] for the definitions of polynomial and strongly polynomial algorithms.

To illustrate our general approach, consider again problem (1.1). Call a base S a *local solution* if there is no pair of elements $i, j \in N$, such that $i \in S$, $j \notin S$ and $S' = S - i + j$ is a base in β which yields a larger objective value than S.

We will introduce a subdivision of the (A, B) plane into a polynomial (in n) number of cells. This subdivision will induce a partition of the points $(A(S), B(S))$, $S \in \beta$. Furthermore, we will show that each one of the cells corresponds to at most one base which is a local solution. An optimal solution base to (1.1), S^*, is one of these local solutions. It will be contributed by that unique cell of the plane containing $(A(S^*), B(S^*))$.

In the case of model (1.1) the subdivision of the plane will be the arrangement induced by a set H of $O(n^2)$ lines. The set H dissects the (A, B) plane into a collection of open convex regions, each bounded by edges, which are segments of the lines. The boundaries of the segments, vertices, are intersection points of lines in H. Taking the edges relatively open with respect to their corresponding lines, we obtain a partition of the plane into cells, i.e., regions, edges, and vertices. This is the arrangement induced by H.

This subdivision has $O(n^4)$ cells and therefore there will be $O(n^4)$ local solution bases.

In the general model we will allow subdivisions which are induced by polynomials, i.e., algebraic varieties.

In §2 we introduce our main tool: tournament graphs defined on matroids. We prove a uniqueness result on an optimal base with respect to a given tournament.

In §3 we define the optimization model and present a strongly polynomial algorithm to find an optimal base. In §4 we discuss extensions to multiparameter objectives, and optimization over nonmatroidal systems.

2. Matroids and tournament graphs.

DEFINITIONS. Let $G = (N, E)$ be a directed graph with vertex set N. G is a *tournament* if for every pair of distinct elements $x, y \in N$ there exist at least one and at most two arcs in E connecting x and y. If there exist exactly two such arcs they are oppositely directed. Note that the common definition of a tournament requires that there is exactly one directed arc connecting x and y [MO].

If x and y are two elements in N, and there exists an arc directed from y to x, then *x dominates y*.

Let $M = (N, \mathscr{F})$ be a matroid with $|N| = n$, and let $G = (N, E)$ be a tournament. Notice that the vertex set of G is the ground set of M. G is called a tournament on N.

Consider two distinct bases S_1 and S_2, of the matroid M. *S_2 improves upon S_1* if there exist an element $y \in S_1$ and an element $x \in N$ such that x dominates y and $S_2 = S_1 - y + x$, i.e., S_2 is obtained from S_1 by a single *swap*.

A basis S of M is *G-optimal* if there exists no base S' which improves upon S. Such a basis will be denoted $S(G)$. Note that $S(G)$ is well defined only if M has a G-optimal base. However, for the sake of convenience, if M has no G-optimal base we will say that $S(G)$ does not exist.

An *independence oracle* for $M = (N, \mathscr{F})$ is a procedure that determines whether a given set $S \subseteq N$ is in $\mathscr{F}$.

A *rank oracle* for $M = (N, \mathscr{F})$ is a procedure that determines for a given set $S \subseteq N$, a subset $A \subseteq S$, $A \in \mathscr{F}$, of largest cardinality. The rank of S is then $\text{rank}(S) = |A|$.

A rank oracle can clearly serve as an independence oracle. $S \subseteq N$ is independent if and only if $\text{rank}(S) = |S|$. On the other hand a rank oracle can be constructed from an independence oracle since $\text{rank}(S)$ can be computed by calling the independence oracle $|S|$ times.

THEOREM 2.1. *Let $M = (N, \mathscr{F})$ be a matroid and let $G = (N, E)$ be a tournament graph on N. Then there exists at most one base of M which is G-optimal.*

PROOF. Suppose that S_1 and S_2 are two distinct G-optimal bases. Let $x \in S_1 - S_2$. By the Symmetric Swap Axiom, [B], there exists $y \in S_2 - S_1$ such that both $S_1 - x + y$ and $S_2 + x - y$ are bases. Since G is a tournament we conclude that either $S_1 - x + y$ improves upon S_1 or $S_2 + x - y$ improves upon S_2. Thus, S_1 and S_2 cannot both be G-optimal. ■

THEOREM 2.2. *Let G be a tournament on N, and suppose that $M = (N, \mathscr{F})$ has a G-optimal base, $S(G)$. Let $x \in N$. Define*

$$A_x = \{ y \in N \mid y \textit{ dominates } x, \textit{ and } x \textit{ does not dominate } y \}$$

$$= \{ y \in N \mid y \neq x \textit{ and } x \textit{ does not dominate } y \}.$$

$$\bar{A}_x = \{ y \in N \mid y \textit{ dominates } x \}.$$

(1) *If $x \in S(G)$ then $\bar{A}_x$ does not span x.*
(2) *If $x \notin S(G)$ then A_x spans x.*

PROOF. Suppose first that $x \in S(G)$, and x depends on $\bar{A}_x$, i.e., $\bar{A}_x$ spans x. Then the set $(S(G)) \cup \bar{A}_x - x$ contains a base, S. Since $x \notin \bar{A}_x$ the bases S and $S(G)$ are distinct. By the Symmetric Swap Axiom there exists $y \in S - S(G)$ such that $S(G) - x + y$ is a base. Since $y \in \bar{A}_x$, the latter base improves upon $S(G)$—thus contradicting the G-optimality of $S(G)$.

Next suppose that $x \notin S(G)$. Let C_x be the (unique) circuit in $S(G) + x$. If x dominates some $y \in C_x$, $y \neq x$, then the base $S(G) - y + x$ would contradict the G-optimality of $S(G)$. Thus, x does not dominate any other element of C_x, and $C_x - x \subseteq A_x$. Therefore, x depends on A_x, i.e., A_x spans x. ■

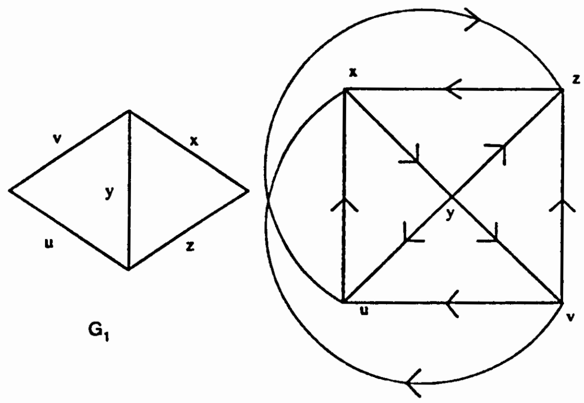
The above theorem suggests an algorithm to find the G-optimal base, $S(G)$, or verify that it does not exist. An immediate corollary of Theorem 2.2 is that if there is some x in N which is spanned by $\bar{A}_x$ but not by A_x, then $S(G)$ does not exist. Note that this situation might indeed occur since, in particular, A_x is a subset of $\bar{A}_x$. Furthermore, if M has a G-optimal base, then it is given by the set

$$X_1 = \{ x \in N \mid \bar{A}_x \text{ does not span } x \}. \tag{2.1}$$

Therefore, a necessary condition for the existence of a G-optimal base is that X_1 is a base of the matroid M. It should be noted, however, that it is possible that X_1 is a base of the matroid and $S(G)$ still does not exist. This is demonstrated by the following example.

EXAMPLE 2.1.

Consider the graphic matroid defined by the graph G_1 with the associated tournament G_2. $X_1 = \{x, z, u\}$. However $S(G)$ does not exist since X_1 is improved upon by $\{y, z, u\}$. ■



G_1 G_2

Given that X_1 is a base, testing for the existence of a G-optimal base now amounts to checking whether X_1 can be improved upon by a single swap of a pair of elements.

Algorithm 1. Finding $S(G)$

Step 1. Compute A_x and $\bar{A}_x$ for all $x \in N$.

Step 2. Set $X_1 = \phi$.

Step 3. For each $x \in N$ do the following:

Determine whether $\bar{A}_x$ spans x by computing and comparing the rank of $\bar{A}_x$ and the rank of $\bar{A}_x + x$. If x is not spanned by $\bar{A}_x$ set $X_1 \leftarrow X_1 + x$. Otherwise, determine whether A_x spans x by computing and comparing the rank of A_x and the rank of $A_x + x$. If x is not spanned by A_x, stop and conclude that $S(G)$ does not exist.

Step 4. If X_1 is not a base, stop and conclude that $S(G)$ does not exist.

Step 5. Test whether there exists $x \in X_1$ and $y \in A_x$, $y \notin X_1$, such that $X_1 - x + y$ is a base. If none exists $S(G) = X_1$. Otherwise, stop and conclude that $S(G)$ does not exist.

To estimate the complexity of Algorithm 1, we assume the existence of an independence oracle or a rank oracle.

Step 1 can be performed in $O(n^2)$ time by using the tournament graphs. Step 3 amounts to $O(n)$ rank evaluations. Step 4 will need 1 rank evaluation, while Step 5 requires $O(m(n - m))$ such evaluations, where m is the rank of the underlying matroid.

3. The optimization model. Given the matroid $M = (N, \mathscr{F})$, suppose that each element $i \in N$ is associated with an ordered pair of real numbers (a_i, b_i). To simplify the presentation we now make the following assumption which is later removed in §4.1.

Nondegeneracy Assumption. If i and j are two distinct elements of N then $(a_i, b_i) \neq (a_j, b_j)$.

For each independent set S define $A(S) = \sum_{i \in S} a_i$ and $B(S) = \sum_{i \in S} b_i$.

Let g be a real strictly convex function defined on R^2, and consider the following optimization problem

$$\max_{S \in \beta} g(A(S), B(S)). \tag{3.1}$$

To solve (3.1) we will apply the machinery developed above. Each point (A, B) in R^2 will be associated with a tournament graph $G(A, B)$ defined on N. Furthermore, if S^* is a base solving (3.1) and $(A, B) = (A(S^*), B(S^*))$, then S^* will be shown to be the (unique) $G(A, B)$-optimal base.

Our solution procedure will then be based upon identifying the different tournament graphs that are induced by *all* points in R^2, and then computing their respective optimal bases. Clearly each such tournament will produce at most one base, i.e., its optimal base. A maximum solution to (3.1) will then be one of these optimal bases. Our focus in this paper is on maximization problems that give rise to a polynomial number of distinct tournaments and hence to as many optimal bases.

We start with the definition of the tournaments.

Let (A, B) be a point in R^2. Define a directed graph $G(A, B)$ with a vertex set N, as follows: Let i, j be distinct elements in N. Nodes i and j are connected with an arc directed from i to j if and only if

$$g(A - a_i + a_j, B - b_i + b_j) > g(A, B). \tag{3.2}$$

Furthermore, i and j are connected by an arc directed from j to i if and only if

$$g(A + a_i - a_j, B + b_i - b_j) > g(A, B). \tag{3.3}$$

Thus, if both (3.2) and (3.3) are satisfied then there are two oppositely directed arcs connecting i and j.

THEOREM 3.1. *For every (A, B) in R^2, $G(A, B)$ is a tournament graph.*

PROOF. Let i and j be two distinct elements in N. Due to the nondegeneracy assumption the points (A, B), $(A - a_i + a_j, B - b_i + b_j)$ and $(A + a_i - a_j, B + b_i - b_j)$ are three distinct points in R^2. The point (A, B) is the mean point of the other two. Thus, the strict convexity of g implies $(g(A - a_i + a_j, B - b_i + b_j) + g(A + a_i - a_j, B + b_i - b_j))/2 > g(A, B)$. Therefore,

$$\begin{aligned} g(A - a_i + a_j, B - b_i + b_j) - g(A, B) \\ > g(A, B) - g(A + a_i - a_j, B + b_i - b_j). \end{aligned} \tag{3.4}$$

If (3.3) does not hold, then the right-hand side of (3.4) is nonnegative. Therefore the left-hand side is positive and (3.2) is satisfied. This completes the proof. ∎

Our next task is to exhibit the relevance of the $G(A, B)$ optimal base.

THEOREM 3.2. *Let S^* be a maximum solution to* (3.1). *Then S^* is the (unique) $G(A(S^*), B(S^*))$-optimal base.*

PROOF. Suppose that S^* is not the $G(A(S^*), B(S^*))$-optimal base. Thus there exist $i \in S^*$ and $j \in N$, such that j dominates i with respect to $G(A(S^*), B(S^*))$, and $S = S^* - i + j$ is a base. From (3.2) it then follows that $g(A(S), B(S)) > g(A(S^*), B(S^*))$, which in turn contradicts the maximality of S^*. ∎

As a consequence of the last theorem, S^* is found when one detects all possible distinct tournaments, and then locates and compares their optimal bases. To follow this solution strategy we partition R^2 into a finite number of equivalence classes. Let $\{G^t\}$, $t = 1, \ldots, W$ denote the set of distinct tournaments defined on N. It follows directly from the definition of a tournament that $W = O(3^{\binom{n}{2}})$. Define the tth class

Q' by

$$Q' = \{(A, B) \mid (A, B) \in R^2, G(A, B) = G'\}.$$

We say that Q' is *solid* if there exists a base of the matroid, S, such that $(A(S), B(S)) \in Q'$. For our purposes it will suffice to identify only the tournament graphs corresponding to solid equivalence classes. To ensure that the total number of those is not exponential we will next impose some additional conditions on the objective function g, which so far has only been assumed to be strictly convex. The specialized conditions that we consider will be general enough to include several models in the literature. Our approach will also improve upon existing procedures for these models.

Suppose that the function g is a 2-variable polynomial. Let $i, j, i \neq j$ be two elements in N. Define the polynomial $g_{ij}(A, B)$ by

$$g_{ij}(A, B) = g(A - a_i + a_j, B - b_i + b_j) - g(A, B).$$

The number of distinct equivalence classes in R^2 is clearly bounded by the number of topological components induced on R^2 by the set of $n(n-1)$ polynomials $\{g_{ij}(A, B)\}$. (It should be noted that we are considering here all topological components, i.e., of dimension 0, 1 and 2.) It is well known from the general theorems on algebraic manifolds, [MI, W], that the latter number is $O(d^2n^4)$, when d is the degree of the polynomial g. Hence, for this case the number of distinct tournament graphs corresponding to solid equivalence classes $\{Q'\}$ is $O(n^4)$.

Suppose that we can determine the signs of all the polynomials $\{g_{ij}(A, B)\}$ at each one of the $O(n^4)$ topological components induced by these polynomials. Then we can clearly find the $O(n^4)$ tournament graphs associated with these topological components. From Theorems 2.1 and 3.2, each such tournament contributes at most one base, i.e., its optimal base, provided it exists. The maximum solution base to the optimization problem is one of these $O(n^4)$ bases.

To illustrate this approach consider model (1.1) where $g(A, B) = A^2 + B^2$. Then, $g_{ij}(A, B)$ is a linear function, $g_{ij}(A, B) = 2(a_j - a_i)A + 2(b_j - b_i)B + (a_j - a_i)^2 + (b_j - b_i)^2$. To find the $O(n^4)$ tournament graphs it is sufficient to compute the planar (subdivision) graph induced by the arrangement of the $n(n-1)$ lines $\{g_{ij}\}$. See the introduction for the definition of the arrangement. This task can be performed in $O(n^4)$ time and space by the recent algorithms reported in [CGL], [EOS], and [EG]. When $g(A, B)$ is not quadratic, as in (1.1), the task of finding the topological cells, i.e., the set of maximal connected sets, in each of which all polynomials g_{ij} are sign invariant, is not that simple. The algorithms in [CO, BKR, KY] can be used to obtain the $O(n^4)$ different sign patterns of $\{g_{ij}\}$ corresponding to the topological cells. In general, these algorithms are polynomial but not strongly polynomial for two variable polynomials. The number of steps will depend (polynomially) on the length of the binary description of the input, i.e., the rationals $\{(a_i, b_i)\}$, $i \in N$, and the rational coefficients of the polynomial g. However, if the degree of each g_{ij} is at most 2, the procedure in [CO] can be implemented in strongly polynomial time. Since the degree of any polynomial g_{ij} is one less than the degree of g, we conclude that the $O(n^4)$ different sign patterns can be obtained in strongly polynomial time whenever the degree of g is at most 3. The key element of the above procedure is the computation and representation of the topological cells of dimension 0, i.e., isolated intersection points of pairs of functions $\{g_{ij}\}$. When these functions are quadratic, each one of the two components of an intersection point is the solution of a one-dimensional quartic polynomial. It is well known that roots of polynomials of degree at most 4 have "closed form" representations. Using these representations the procedure in [CO] becomes strongly polynomial when the functions $\{g_{ij}\}$ are quadratic. We also note that from Galois' Theory no "closed form" representations exist for a general polynomial of degree greater than 4. Therefore, we cannot extend the above approach to obtain a strongly polynomial scheme when some polynomials $\{g_{ij}\}$ are of degree greater than 2.

The polynomiality assumption on g is not satisfied for models (1.3) and (1.6). However, we will next extend the above approach to include these models and others as well.

Let $i, j \in N$, $i \neq j$. Define

$$R_{ij} = \{(A, B) \mid g(A - a_i + a_j, B - b_i + b_j) > g(A, B)\}.$$

Let $f\colon R^2 \to R^k$ be a mapping of R^2 into R^k. Further, let $\{T_{ij}\}$, $i, j \in N$, $i \neq j$ be a collection of subsets in R^k such that $(A, B) \in R_{ij}$ if and only if $f(A, B) \in T_{ij}$, for $i, j \in N$, $i \neq j$.

Suppose that each subset T_{ij} is an (open) algebraic manifold, i.e., there exists a polynomial $h_{ij}(x_1, \dots, x_k)$ such that

$$T_{ij} = \{(x_1, \dots, x_k) \mid h_{ij}(x_1, \dots, x_k) > 0\}.$$

Then the number of distinct equivalence classes i.e., the number of tournaments induced by the sets $\{R_{ij}\}$, is bounded by the number of topological components induced on R^k by the set of polynomials $\{h_{ij}\}$. If d, the maximum degree of $\{h_{ij}\}$, and k, the dimension of R^k, are constant and independent of n, then the number of components will still be polynomial in n [MI, W]. In this case we are still able to implement the above approach to find an optimal basis.

For illustration purposes consider the model (1.5)-(1.6). Then

$$\begin{aligned} R_{ij} &= \{(A', B') \mid \exp(A' - \log a_i + \log a_j) + \exp(B' - \log b_i + \log b_j) \\ &\qquad > \exp(A') + \exp(B')\} \\ &= \left\{(A', B') \mid \left(\frac{a_j}{a_i} - 1\right)\exp(A') > \left(1 - \frac{b_j}{b_i}\right)\exp(B')\right\}. \end{aligned}$$

Define $f\colon R^2 \to R^1$, by $f(A', B') = \exp(A' - B')$. Also let

$$T_{ij} = \left\{x \mid x\left(\frac{a_j}{a_i} - 1\right) > \left(1 - \frac{b_j}{b_i}\right)\right\}.$$

The set of tournaments induced by $\{R_{ij}\}$ is the same as the set induced by $\{T_{ij}\}$. The latter set is of $O(n^2)$ cardinality, and it corresponds to the partition of the real line induced by the rational points $\{d_{ij}\}$,

$$d_{ij} = \left(1 - \frac{b_j}{b_i}\right)\Big/\left(\frac{a_j}{a_i} - 1\right), \qquad i, j \in N, \quad i \neq j.$$

We note in passing that for model (1.2)-(1.3)

$$R_{ij} = \left\{(A, B') \mid -a_i + a_j > \left(1 - \frac{b_j}{b_i}\right)\exp(B')\right\}.$$

Thus, define $f\colon R^2 \to R^1$ by $f(A, B') = \exp(B')$, and set $T_{ij} = \{x \mid -a_i + a_j > x(1 - b_j/b_i)\}$. The number of tournaments that should be considered is again $O(n^2)$. They correspond to the partition of the line induced by the points $b_i(a_j - a_i)/(b_i - b_j)$, $i, j \in N$, $i \neq j$.

As a final example consider the problem of minimizing the cost/reliability ratio over all spanning trees of a graph [CAN]. This problem has been solved in [CAT] by a polynomial (but not strongly polynomial) algorithm. We now exhibit a strongly polynomial algorithm.

Using our notation, the model is

$$\text{(3.5)} \qquad \operatorname*{Max}_{S\in\beta}\left\{ \prod_{i\in S} b_i \Big/ \sum_{i\in S} a_i \right\},$$

where $a_i, b_i > 0$ for $i \in N$. Setting $A(S) = \sum_{i\in S} a_i$, $B'(S) = \sum_{i\in S} \log b_i$ and using the monotonicity of the log function we note that (3.5) is equivalent to

$$\operatorname*{Max}_{S\in\beta}\{B'(S) - \log A(S)\}.$$

Thus,

$$R_{ij} = \{(A, B') \mid Ab_j > b_i(A - a_i + a_j)\}, \quad T_{ij} = \{x \mid xb_j > b_i(x - a_i + a_j)\}, \quad i, j \in N, i \neq j.$$

Therefore, again $O(n^2)$ tournament graphs need to be considered. These tournaments correspond to the partition of the line determined by the points

$$\text{(3.6)} \qquad d_{ij} = \frac{b_i(a_j - a_i)}{b_j - b_i}, \qquad i, j \in N, \quad i \neq j.$$

We can now state a strongly polynomial algorithm for (3.5) when M is a graphic matroid.

Algorithm 2

Step 1. Compute and sort the numbers $\{d_{ij}\}$ defined in (3.6).

Step 2. Associate a tournament graph with each (distinct) positive point d_{ij}, and with the mid-point of each open (positive) interval defined by two consecutive points of the sorted sequence found in Step 1. Thus each tournament is represented by some point on the line. If v is such a point, let G_v be the corresponding tournament.

Step 3. For each v construct G_v as follows: j dominates i in G_v (i.e., there is a directed arc from i to j in G_v) if and only if one of the following conditions is satisfied:

(a) $b_j = b_i$ and $a_j < a_i$.

(b) $b_j > b_i$ and $v > d_{ij}$.

(c) $b_j < b_i$ and $v < d_{ij}$.

Step 4. For each v find the unique optimal G_v base (provided it exists), using Algorithm 1.

Step 5. Compute the objective in (3.5) at each one of the bases generated in (4) to find an optimal solution to (3.5).

To evaluate the complexity of the above procedure, we first find the complexity of Algorithm 1 when the matroid is a graphic matroid. In this case n is the number of edges of the underlying graph and the rank of the matroid is $(k - 1)$, where k is the number of nodes.

For a graphic matroid a rank evaluation can be performed in $O(n)$ time. Therefore, the complexity of Algorithm 1 for a graphic matroid is $O(kn^2)$.

There are $O(n^2)$ tournaments $\{G_v\}$ to consider in Algorithm 2. Therefore, its overall complexity is $O(kn^4)$. Actually, in Step 4 of Algorithm 2 we do not have to verify the existence of the optimal G_v base. It will simply suffice to find the respective base X_1 by Algorithm 1. In particular, when executing Algorithm 1 we can skip Step 5 there. This leads to an overall complexity of $O(n^4)$ for Algorithm 2.

Since our goal was only to exhibit the existence of a strongly polynomial algorithm we have not attempted here to obtain lower order algorithms by utilizing more sophisticated data structures.

4. Extensions and concluding remarks.

4.1. *Relaxing the nondegeneracy assumption.* When the function g is a polynomial the Nondegeneracy Assumption made in §3 can be removed by using perturbation. Consider the perturbation where each a_i, $i \in N$, is now replaced by $a_i + \epsilon^i$, $\epsilon > 0$.

For each $S \in \beta$ the objective value $g(A(S), B(S))$ can now be viewed as a one dimensional polynomial in ϵ. Denote this polynomial by $g^\epsilon(S)$.

For each $\epsilon > 0$ the ϵ-perturbed problem is to find a base, $S^*(\epsilon)$, maximizing the ϵ-perturbed objective, i.e., $g^\epsilon(S^*(\epsilon)) = \text{Max}_{s \in \beta}\{g^\epsilon(S)\}$.

Consider the subcollection consisting only of the distinct polynomials in the collection $\{g^\epsilon(S)\}$, $S \in \beta$. Let $\mathscr{R}$ denote the set of all intersection (crossing) points of pairs of polynomials in this subcollection, i.e., an element of $\mathscr{R}$ is a root of a polynomial of the form $g^\epsilon(S_1) - g^\epsilon(S_2)$, where $S_1, S_2 \in \beta$. $\mathscr{R}$ is clearly a finite set. Let ϵ_1 be the smallest positive element in $\mathscr{R}$. (If none exists, $\epsilon_1 = \infty$.)

Therefore, there is a unique polynomial that dominates any other polynomial in the above subcollection for *all* $0 < \epsilon < \epsilon_1$. Furthermore, every base which maximizes the $\bar{\epsilon}$-perturbed objective for *some* $\bar{\epsilon}$, $0 < \bar{\epsilon} < \epsilon_1$, also maximizes the ϵ-perturbed objective for all $0 < \epsilon < \epsilon_1$. Due to the continuity of g such a base is also optimal for the original problem, i.e., $\epsilon = 0$. Thus, to solve the original problem it is sufficient to find some $\bar{\epsilon}$, $0 < \bar{\epsilon} < \epsilon_1$, explicitly, and then compute some base, $S^*(\bar{\epsilon})$, which maximizes the $\bar{\epsilon}$-perturbed objective.

Let $\bar{\mathscr{R}}$ be the set obtained by augmenting $\epsilon = 0$ to $\mathscr{R}$. $\bar{\epsilon}$ can be chosen as any positive lower bound on the distance between distinct elements of $\bar{\mathscr{R}}$. To obtain an explicit expression of $\bar{\epsilon}$ we can use the results in [CAT1, GG, MIN]. We only need to know explicit upper bounds on the degrees and the sizes of the coefficients of the polynomials in the above collection.

Such bounds can easily be obtained and expressed in terms of the numbers $\{(a_i, b_i)\}$, $i \in N$, and the coefficients of the objective polynomial g. It is easily observed from [CAT1, GG, MIN] that the size of $\bar{\epsilon}$ is polynomial in the size of the above numbers. Furthermore, the total number of elementary rational operations which are necessary to compute $\bar{\epsilon}$ is independent of the encoding of the data. This number is bounded by a polynomial in n.

Finally, note that by using a different perturbation we can also relax the strict convexity assumption, and assume that g is only convex. In this case $g(A, B)$ is augmented by the term $\epsilon(A^2 + B^2)$. An argument similar to the one used above shows that it is sufficient to solve the modified objective for a specified value of ϵ that can be obtained explicitly in strongly polynomial time.

4.2. *On the solution of nonmatroidal systems.* The solution procedure that we have presented above is not applicable, in general, to combinatorial problems that do not satisfy matroidal properties. However, in those cases we might still get meaningful results if we apply the general method of generating all relevant extreme points described in the Introduction. This is the approach taken in [CAT] to produce a polynomial (but not strongly polynomial) algorithm to find the minimum cost/reliability ratio over all spanning trees of a graph.

To illustrate this general procedure consider the problem of minimizing the cost/reliability ratio over all directed simple paths of a graph connecting two specified nodes s and t.

Let $G = (V, E)$ be a directed graph with V and E as its sets of vertices and (directed) edges respectively. For each edge $i \in E$ let b_i $(0 < b_i \leq 1)$, and a_i, respectively, be the probability of functioning and the positive cost.

Given a source-sink pair, $s, t \in V$, $s \neq t$, let β denote the set of all simple directed paths from s to t. The set is not the collection of bases of a matroid. The optimization problem is

$$\text{(4.1)} \qquad \min_{P \in \beta} \frac{\sum_{i \in P} a_i}{\prod_{i \in P} b_i}.$$

Define $b_i' = -\log b_i$, $i \in E$, $A(P) = \sum_{i \in P} a_i$, and $B'(P) = \sum_{i \in P} b_i'$. Then (4.1) is equivalent to

$$\text{(4.2)} \qquad \min_{P \in \beta} g(A(P), B'(P))$$

where $g(A, B') = \log A + B'$.

The concavity of $g(A, B')$ implies that a minimizer of (4.2) is a path, say P^*, for which $A(P^*), B'(P^*)$ is an extreme point of the convex hull of the set of two dimensional points $(A(P), B'(P))$, $P \in \beta$.

Furthermore, since g is isotone, i.e., montonic in both variables, P^* is an extreme point which is the unique minimizer of some isotone linear function. That is, there exists $\gamma > 0$, such that P^* is the unique minimizer of $A + \gamma B'$, over the above convex hull.

In our model $0 < b_i \leq 1$, $i \in E$, and therefore

$$\text{(4.3)} \qquad (A(P), B'(P)) \geq 0 \quad \text{for all paths } P.$$

It is shown in [G] that, p, the number of distinct (extreme) paths of the above convex hull that will minimize isotone linear functions, $A + \gamma B'$, $\gamma \geq 0$, is $p = O((2n + 1)^{1 + \log n})$ (n is the number of vertices of G).

Using standard techniques (see [CAN, G]) all these extreme paths can be constructed by solving $O(p)$ shortest path problems. Due to (4.3), the edge lengths in these shortest path problems are nonnegative. For a typical problem the length of edge i, d_i, will be given by

$$\text{(4.4)} \qquad d_i = a_i|B'(P_1) - B'(P_2)| + |A(P_1) - A(P_2)|b_i',$$

where P_1 and P_2 is a distinguished pair of paths.

Suppose that the original data $\{a_i\}$ and $\{b_i\}$ are rational. Then each d_i is a linear form in the logarithms of the data. Using the results in [CAT] we can determine signs of such expressions in polynomial time.

Therefore, the traditional algorithms to solve the shortest path problem can be implemented in polynomial time to the data $\{d_i\}$ in (4.4). Coupled with the above bound on the number of extreme paths, we have thus produced a subexponential algorithm for the minimum cost/reliability ratio path problem (4.1). To our knowledge, this is the first subexponential algorithm for this problem. See [HE] for related results on parametric shortest path problems.

It is worth mentioning that there are several nonmatroidal systems for which the above general method of generating all relevant extreme points yields polynomial algorithms. One such model is the following.

Let $T = (V, E)$ be an undirected tree, and let x be some distinguished vertex in V. Suppose that each edge i in E is associated with a pair of numbers (a_i, b_i). Let β be the set of all nonempty connected subtrees of T rooted at x, i.e., subtrees containing x. Consider the maximization (3.1) over β. This model can be solved in strongly polynomial time using the general approach, if g is, for example, polynomial. The polynomiality follows mainly from the fact that there is a linear number of relevant extreme points in this case.

Another solvable variant of this model is obtained by letting β be the set of all nonempty subtrees of T. In this case the number of extreme points can be shown to be quadratic in $|V|$.

4.3. *Extensions to multiparameter problems.* The algorithm, presented in §3, to maximize two-parameter objectives can conceptually be extended to multiparameter problems. For the k-dimensional case the ith element of the matroid will be associated with k parameters, $a_i^1, a_i^2, \ldots, a_i^k$. We then define for each base S, $A^j(S) = \sum_{i \in S} a_i^j$, $j = 1, \ldots, k$.

The optimization model is

$$\text{(4.5)} \qquad \operatorname*{Max}_{S \in \beta} g(A^1(S), \ldots, A^k(S))$$

where g is a convex polynomial, and β is the set of bases of the matroid M. Given $A^1, \ldots, A^k$, we say that j dominates i if

$$g\left(A^1 - a_i^1 + a_j^1, \ldots, A^k - a_i^k + a_j^k\right) > g(A^1, \ldots, A^k).$$

With this domination ordering we define the appropriate tournament and obtain the obvious extension of Theorems 3.1 and 3.2.

We can also obtain the topological cells induced on R^k by the respective polynomials $\{g_{ij}\}$, using the algorithms in [TAR, CO, BKR, KY]. These algorithms are polynomial for a fixed value of k.

Therefore, for a fixed value of k, we can still construct a polynomial algorithm for (4.5), provided that a polynomial routine for testing independence in the matroid is available.

4.4. *Other related models.* The solution procedure of §3 can easily be adapted to maximize $g(A(S), B(S))$ over $\mathscr{F}$, the set of all independent sets, instead of β, the set of all bases. If m is the rank of $M = (N, \mathscr{F})$, for each integer k, $k = 1, \ldots, m$, define the k-truncated matroid $M_k = (N, \mathscr{F}_k)$ by $\mathscr{F}_k = \{S \mid S \in \mathscr{F}, |S| \le k\}$.

Let β_k be the set of all bases of M_k. Using the procedure in §3, we can find for each k, $k = 1, \ldots, m$, S_k^*, a minimizer of $\operatorname{Max}_{S \in \beta_k} g(A(S), B(S))$.

It is now obvious that at least one of the sets $S_1^*, \ldots, S_m^*$ is a minimizer of $\operatorname{Max}_{S \in \mathscr{F}} g(A(S), B(S))$.

At this point it is worth mentioning that, unlike the linear case, when g is a general convex function, the bases $S_1^*, \ldots, S_m^*$ may not be nested. For example, let $N = \{1,2,3\}$ and let $\mathscr{F}$ be the power set of N. If $(a_1, b_1) = (1,0)$ and $(a_2, b_2) = (a_3, b_3) = (0, 3/4)$ we obtain $S_1^* = \{1\}$ and $S_2^* = \{2,3\}$ for the objective in (1.1).

We should also observe that problem (3.1) may possess a local solution base, (i.e., one which can not be improved upon by a single swap), that is not globally optimal. For example, let $N = \{1,2,3,4\}$ and let $\mathscr{F}$ be the power set of N. Consider M_2, the 2-truncated matroid of $M = (N, \mathscr{F})$. Set $(a_1, b_1) = (a_2, b_2) = (3/4, 0)$ and $(a_3, b_3) = (a_4, b_4) = (0,1)$. Consider problem (1.1) on M_2. The set $\{1,2\}$ is a local optimal base which is not globally optimal. Indeed, for this example, $\{1,2\}$ is one of the bases generated and examined by our solution procedure.

Finally we consider the problem obtained from (3.1) by replacing the Max operator by the Min operator. We claim that in this case the problem becomes NP-hard even when g is quadratic and M is the uniform matroid.

Given a set of positive integers $\{a_1, \ldots, a_n\}$ consider the following partition problem which is known to be NP-complete.

Determine whether there exists a subset S of $N = \{1, \ldots, n\}$, such that $A(S) = A(N) - A(S)$.

Consider the matroid $M = (N, \mathscr{F})$ by letting $\mathscr{F}$ be the power set of N, i.e., $\mathscr{F} = \{S \mid S \subseteq N, S \neq \emptyset\}$. Let $g(A)$ be the quadratic $g(A) = A^2 + (A(N) - A)^2$. Then it is clear that $\{a_1, \ldots, a_n\}$ has a partition if and only if the solution value to the following minimization problem is $2(A(N)/2)^2$.

$$\text{(4.6)} \qquad \min_{S \in \mathscr{F}} g(A(S)).$$

We have already noted above that (4.6) can be solved by solving for each $k = 1, \ldots, n$, the problem

$$\text{(4.7)} \qquad \min_{S \in \beta_k} g(A(S))$$

where β_k is the set of all bases of the k-truncated matroid. (Note that the k-truncated matroid is exactly the k-uniform matroid, i.e., $S \in \mathscr{F}_k$ if and only if S is a nonempty subset of N whose cardinality does not exceed k.)

Thus, we conclude that the minimization version of (3.1) is NP-hard even when g is quadratic and M is the uniform matroid.

References

[B] Brylawski, T. H. (1973). Some Properties of Basic Families of Subsets. *Discrete Math.* **6** 333–341.

[BKR] Ben-Or, M., Kozen, D. and Rief, J. (1984), (1986). The Complexity of Elementary Algebra and Geometry. *Proc. 16th Annual ACM Sympos. Theory of Computing*, Washington, 1984, 457–464. Also *J. Comput. System Sci.* **32** 251–264.

[C] Chandrasekaran, R. (1977). Minimal Ratio Spanning Trees. *Networks.* **7** 335–342.

[CAN] ______, Aneja, Y. P. and Nair, K. P. K. (1984). Minimal Cost Reliability Ratio Spanning Tree. *Ann. Discrete Math.* **9** 117–123.

[CAT] ______ and Tamir, A. Polynomial Testing of the Query "Is $a^b > c^d$?" with Applications to Finding a Minimal Cost Reliability Ratio Spanning Tree. *Discrete Appl. Math.* **9** 117–123.

[CAT1] ______ and ______. (1984). Optimization Problems with Algebraic Solutions: Quadratic Fractional Programs and Ratio Games. *Math. Programming* **30** 326–339.

[CGL] Chazelle, B., Guibas, L. J. and Lee, D. T. (1983), (1985). The Power of Geometric Duality, in *Proc. 24th Annual IEEE Sympos. Foundations of Computer Science* 217–225. Also, *BIT* **25** 76–90.

[CO] Collins, G. E. (1975). Quantifier Elimination for Real Closed Fields by Cylindric Algebraic Decomposition. *Proc. 2nd GI Conf. Automata Theory and Formal Languages*, Springer Verlag LNCS **35**, Berlin 134–183.

[EG] Edelsbrunner, H. and Guibas, L. J. (1986). Topologically Sweeping an Arrangement. *18th Annual ACM Sympos. Theory of Computing*, 389–403.

[EOS] ______, O'Rourke, J. and Seidel, R. (1983), (1986). Constructing Arrangements of Lines and Hyperplanes with Applications. *Proc. 24th Annual IEEE Sympos. Foundations of Computer Science*, 83–91. Also *SIAM J. Computing* **15** 341–363.

[G] Gusfield, D. M. (1980). Sensitivity Analysis for Combinatorial Optimization. Electronic Research Laboratory, College of Engineering, University of California, Berkeley, May, Memo. No. UCB/ERL M80/22.

[GG] Güting, R. (1961). Approximation of Algebraic Numbers by Algebraic Numbers. *Michigan Math. J.* **8** 149–159.

[GLS] Grötschel, M., Lovász, L. and Schrijver, A. (1988). *The Ellipsoid Method and Combinatorial Optimization*. Springer, Berlin.

[HE] Henig, M. I. (1985). The Shortest Path Problem with Two Objective Functions. *European J. Oper. Res.* **25** 281–291.

[KY] Kozen, D. and Yap, C. K. (1985). Algebraic Cell Decomposition in NC. *Proc. 26th Annual Sympos. Foundations of Computer Science*, 515–521.

[ME] Megiddo, N. (1985). A Note on Sensitivity Analysis in Algebraic Algorithms. *IBM Research Report*, RJ 4958, December.

[MI] Milnor, J. (1964). On the Betti Numbers of Real Algebraic Varieties. *Proc. Amer. Math. Soc.* **15** 275–280.

[MIN] Mignotte, M. (1982). Identification of Algebraic Numbers. *J. Algorithms* 3 197–204.

[MO] Moon, J. W. (1986). *Topics on Tournaments*, Holt Rinehart and Winston, New York.

[TAR] Tarski, A. (1951). *A Decision Method for Elementary Algebra and Geometry* second edition, revised, University of California Press, Berkeley and Los Angeles.

[W] Warren, H. E. (1968). Lower Bounds for Approximation by Nonlinear Manifolds. *Trans. American Math. Soc.* **133** 167–178.

HASSIN: DEPARTMENT OF STATISTICS, TEL AVIV UNIVERSITY, TEL AVIV, ISRAEL

TAMIR: DEPARTMENT OF STATISTICS AND OPERATIONS RESEARCH, NEW YORK UNIVERSITY, NEW YORK, NEW YORK 10003
DEPARTMENT OF STATISTICS, TEL AVIV UNIVERSITY, TEL AVIV, ISRAEL