

Convex Combinatorial Optimization*

Shmuel Onn and Uriel G. Rothblum

Technion—Israel Institute of Technology,
32000 Haifa, Israel
{onn,rothblum}@ic.technion.ac.il

Abstract. We introduce the convex combinatorial optimization problem, a far-reaching generalization of the standard linear combinatorial optimization problem. We show that it is strongly polynomial time solvable over any edge-guaranteed family, and discuss several applications.

1. Introduction

The general linear combinatorial optimization problem is the following:

Linear Combinatorial Optimization. Given a family $\mathcal{F} \subseteq 2^N$ of subsets of $N := \{1, \dots, n\}$ and a rational weighting $w: N \rightarrow \mathbb{Q}$, find $F \in \mathcal{F}$ of maximum weight $w(F) := \sum_{j \in F} w(j)$.

There is a massive body of knowledge on the computational complexity of this problem for various classes of families presented in various ways (in terms of n and sometimes additional parameters), and efficient algorithms in numerous cases, see [13]. For instance, if \mathcal{F} is the family of stable sets in a given graph with vertex set N then the problem is NP-hard whereas if \mathcal{F} is the family of matchings in a given graph with edge set N then the problem is polynomial time solvable.

In this article we consider the following generalization of linear combinatorial optimization.

* The first author was supported in part by a grant from ISF (the Israel Science Foundation), by Technion VPR Fund (E. and J. Bishop Research Fund), by the Technion President Fund, and at MSRI by NSF Grant DMS-9810361. The second author was supported in part by a grant from ISF (the Israel Science Foundation), by a VPR grant at the Technion, and by the Fund for the Promotion of Research at the Technion.

Convex Combinatorial Optimization. Given $\mathcal{F} \subseteq 2^N$ with $N = \{1, \dots, n\}$, a vectorial weighting $w: N \rightarrow \mathbb{Q}^d$, and a convex functional $c: \mathbb{Q}^d \rightarrow \mathbb{Q}$, find $F \in \mathcal{F}$ of maximum value $c(w(F))$.

The standard linear combinatorial optimization problem over a family \mathcal{F} is recovered as the special case with $d = 1$, $w: N \rightarrow \mathbb{Q}$ weighting by scalars, and $c: \mathbb{Q} \rightarrow \mathbb{Q}$: $x \mapsto x$ the identity.

Convex combinatorial optimization has a very broad expressive power and conveniently captures a variety of problems studied in the operations research and mathematical programming literature including quadratic assignment, inventory management, scheduling, reliability, bargaining games, clustering, and vector partitioning, see [2], [4], [7], [10], [12], [22], [27], [43], and references therein. In Section 3 we discuss some of these applications in detail and demonstrate that, as a consequence of our framework, all admit a simple unified strongly polynomial time algorithm.

A particularly successful general methodology for linear combinatorial optimization is the geometric approach inaugurated by Edmonds [17] and culminated by Grötschel et al. [25], outlined as follows. With each family member $F \in \mathcal{F}$ is associated its indicator $\mathbf{1}_F := \sum_{j \in F} \mathbf{1}_j$ with $\mathbf{1}_j$ the j th standard unit vector in \mathbb{Q}^n , and with the family is associated the polytope

$$\mathcal{P}^{\mathcal{F}} := \text{conv}\{\mathbf{1}_F : F \in \mathcal{F}\} \subset \mathbb{Q}^n.$$

Extending w to \mathbb{Q}^n by $w(x) := \sum_{j=1}^n w(j)x_j$, the problem reduces to maximizing the linear functional w over $\mathcal{P}^{\mathcal{F}}$. This leads to the study of *facets* of $\mathcal{P}^{\mathcal{F}}$; when these can be suitably controlled, the problem is polynomial time solvable via the ellipsoid method [34] for linear programming.

In this article we further develop the geometric approach and use it to provide a widely applicable general methodology for convex combinatorial optimization as well. Our framework leads to the study of *edge-directions* of $\mathcal{P}^{\mathcal{F}}$; when these can be suitably controlled, the problem is efficiently solvable via zonotope (or hyperplane arrangement) methods as follows.

Theorem 1.1. *For any fixed d , there is a strongly polynomial oracle time algorithm solving convex combinatorial optimization over any edge-guaranteed family presented by a membership oracle.*

A family is edge-guaranteed if it comes with a set of vectors that contains a direction of every edge of the polytope $\mathcal{P}^{\mathcal{F}}$; the precise definition is given in Section 2. All families underlying the various applications in Section 3 naturally possess this property. The assumption of fixed d is also natural and necessary: already for $d = 1$, the problem generalizes linear combinatorial optimization which is frequently intractable; and when d is variable, the problem captures NP-hard instances even for the simple power set family $\mathcal{F} = 2^N$, see Example 3.1 below.

The main part of the proof of this theorem is a reduction of the convex combinatorial optimization problem over a family \mathcal{F} to the solution of polynomially many standard linear combinatorial optimization counterparts over the same family \mathcal{F} . The reduction

makes use of several results about zonotopes which are available in combinatorial and computational geometry, but so far have not been integrated and harnessed in a systematic way to discrete optimization.

The repeated solution of each of the standard linear combinatorial optimization counterparts can be done following either one of the following two approaches. The first is to use any efficient ad hoc algorithm available in the literature for the specific family \mathcal{F} at hand: this typically leads to the best overall running time. This approach indeed applies to all of the applications discussed in Section 3, since each admits a very fast ad hoc algorithm (ranging from simple greedy to sophisticated min-cost flow). The second approach, which is generic and works for any \mathcal{F} , takes advantage of the fact that a *test set* (see [47]) is readily available for any edge-guaranteed family, and (see [45] and references therein), using scaling [19] and Diophantine approximation [20], allows the efficient solution of the necessary linear optimization counterparts.

The article is organized as follows. In Section 2 we prove Theorem 1.1 as well as some other results, and discuss some relevant issues, as follows. In Section 2.1 we discuss the necessary preliminaries on zonotopes and edge-directions. In Section 2.2 we prove Theorem 2.6 providing the reduction of convex combinatorial optimization to polynomially many standard linear combinatorial optimization counterparts. In Section 2.3 we discuss the generic approach for solving the counterparts and combine it with Theorem 2.6 to establish Theorem 1.1. In Section 2.4 we discuss the problem of finding short monotone paths on $(0, 1)$ -polytopes, provide Lemma 2.10 which is a certain $(0, 1)$ -analog of the Klee–Minty cube [36], and raise some questions. In Section 2.5 we consider classes of edge-guaranteed families and conclude Corollary 2.12 concerning such *edge-well-behaved classes* (defined therein). In Section 2.6 we discuss projection representation which sometimes helps control edge-directions. Section 3 is devoted to applications: in Section 3.1 we discuss quadratic assignment and matroids, whereas in Section 3.2 we make use of projection representation and discuss in detail the broadly applicable shape vector partitioning problem. Section 4 contains some final remarks and open problems.

We conclude this Introduction with some comments. First, our results make use of and provide an efficient enumeration of the vertices of the polytope $\mathcal{P}_w^{\mathcal{F}} := \text{conv}\{w(F) : F \in \mathcal{F}\}$ which is a projection of $\mathcal{P}^{\mathcal{F}}$; as the maximum of a convex functional c over $\mathcal{P}_w^{\mathcal{F}}$ is attained at a vertex and each vertex has the form $w(F)$ with $F \in \mathcal{F}$, this provides a strategy for addressing the convex combinatorial optimization problem. One of the difficulties we overcome is that the number of sets in \mathcal{F} is typically exponential in n and hence it is generally impossible to construct $\mathcal{P}_w^{\mathcal{F}}$ directly efficiently. As a consequence of our efficient vertex enumeration of $\mathcal{P}_w^{\mathcal{F}}$, our results immediately extend to the larger class of problems where c is any functional which is guaranteed to attain a maximum over $\mathcal{P}_w^{\mathcal{F}}$ at a vertex, e.g., when c is (edge-)quasi-convex on $\mathcal{P}_w^{\mathcal{F}}$, see [31]. In particular, our results apply when c is (asymmetric) Schur convex and the edge-directions of $\mathcal{P}_w^{\mathcal{F}}$ are differences of standard unit vectors in \mathbb{Q}^d , again see [31]. Also, the results can be generalized to some extent from $(0, 1)$ -problems to integer programming. Second, we note that in studying edge-directions, we make use of projections of polytopes; thus the Billera–Sturmfels theory of fiber polytopes [9], related to various aspects of polytope projection, may be helpful in the classification of edge-well-behaved classes of families. Also, some new questions that we raise about graphs of $(0, 1)$ -polytopes might be addressed through

the Billera–Sarangerajan universal embedding of such polytopes as traveling salesman polytopes [8].

We hope this exposition will make our framework a widely accessible tool in the arsenal of discrete optimization, and will stimulate the study of edge-directions of polytopes $\mathcal{P}^{\mathcal{F}}$ for various combinatorially defined families. Since convex combinatorial optimization is often intractable, there is also much room for the study of approximation algorithms for this problem for various families \mathcal{F} , and we hope this article will stimulate research on this yet unexplored ground.

2. Edge-Directions and the Algorithmic Solution

2.1. Edge-Directions and Zonotopes

We start by introducing the necessary terminology and collecting several facts that we make use of; for some we only provide a reference and for others we provide a short proof.

The *zonotope* generated by a set of vectors $E = \{e^1, \dots, e^m\}$ in \mathbb{Q}^d is the Minkowski sum

$$\begin{aligned} \mathcal{Z} &= \text{zone}(E) := \sum_{i=1}^m [-e^i, e^i] = \left\{ \sum_{i=1}^m \lambda_i e^i : -1 \leq \lambda_i \leq 1 \right\} \\ &= \text{conv} \left\{ \sum_{i=1}^m \lambda_i e^i : \lambda_i = \pm 1 \right\} \subset \mathbb{Q}^d. \end{aligned}$$

The following bound on the number of vertices of zonotopes has been rediscovered many times over the years; see, e.g., [11] and [28] for early references and [23] and [48] for recent extensions and refinements.

Lemma 2.1. *The number of vertices of any d -dimensional zonotope generated by m vectors is at most $2 \sum_{i=0}^{d-1} \binom{m-1}{i}$. Thus, for fixed d it is $O(m^{d-1})$ and hence polynomially bounded in m .*

Each vector $a \in \mathbb{Q}^d$ is also interpreted as the linear functional on \mathbb{Q}^d given via the standard inner product $a \cdot x = \sum_{i=1}^d a_i x_i$. The *normal cone* of a polytope \mathcal{P} at its face F is the (relatively open) cone $C_{\mathcal{P}}^F$ of those linear functionals a which are maximized over \mathcal{P} precisely at points of F . The following computational analogue of Lemma 2.1 is provided by the algorithm in [15] and [16] (the latter reference provides a necessary correction of the former); some extensions are again in [23].

Lemma 2.2. *Fix any $d \geq 3$. Then all vertices of any d -dimensional zonotope \mathcal{Z} generated by m given vectors can be listed, each vertex u along with a linear functional $a(u) \in C_{\mathcal{Z}}^u$ uniquely maximized over \mathcal{Z} at u , in strongly polynomial time using $O(m^{d-1})$ arithmetic operations.*

A similar statement holds for $d \leq 2$ as well but with arithmetic complexity $O(m^d)$. Also, we point out that the space complexity of the algorithm of Lemma 2.2 coincides with its time complexity. Note that throughout we are mainly interested in *strongly polynomial (oracle) time algorithms*, that is, algorithms that perform a polynomial number of arithmetic operations (and calls to the relevant oracles if any) and are also polynomial time in the Turing computation model.

The collection of normal cones of a polytope \mathcal{P} at all faces is called the *normal fan* of \mathcal{P} (see [26]). A polytope \mathcal{Z} is a *refinement* of a polytope \mathcal{P} if the closure of each normal cone of \mathcal{P} is the union of closures of normal cones of \mathcal{Z} . A standard result shows that \mathcal{Z} is a refinement of \mathcal{P} if and only if the normal cone of every vertex of \mathcal{Z} is contained in the normal cone of some vertex of \mathcal{P} , and we use this property interchangeably with the above definition of refinement. A *direction* of an edge $[u, v]$ of a polytope \mathcal{P} is any nonzero scalar multiple of $v - u$. We provide a simple proof of the following fact (see [23]) which is quite central to our considerations.

Lemma 2.3. *Let \mathcal{P} be a polytope and let E be a finite set of vectors containing a direction of every edge of \mathcal{P} . Then the zonotope $\mathcal{Z} := \text{zone}(E)$ generated by E is a refinement of \mathcal{P} .*

Proof. Let $E = \{z^1, \dots, z^m\}$. Consider any vertex u of \mathcal{Z} . Then $u = \sum_{i=1}^m \lambda_i z_i$ for some $\lambda_i = \pm 1$ and hence its normal cone $\mathcal{C}_{\mathcal{Z}}^u$ consists of those a satisfying $a \cdot \lambda_i z_i > 0$ for all i . Let v be a vertex of \mathcal{P} at which some such \hat{a} (belonging to $\mathcal{C}_{\mathcal{Z}}^u$) is maximized over \mathcal{P} . Consider any edge $[v, w]$ of \mathcal{P} . Then $v - w = \alpha_i z_i$ for some scalar $\alpha_i \neq 0$ and some z_i , and $0 \leq \hat{a} \cdot (v - w) = \hat{a} \cdot \alpha_i z_i$, implying $\alpha_i \lambda_i > 0$. It follows that every a in the cone $\mathcal{C}_{\mathcal{Z}}^u$ of the vertex u of \mathcal{Z} satisfies $a \cdot (v - w) > 0$ for every edge of \mathcal{P} containing v and therefore a is also in the cone $\mathcal{C}_{\mathcal{P}}^v$ of the vertex v of \mathcal{P} , and hence $\mathcal{C}_{\mathcal{Z}}^u \subseteq \mathcal{C}_{\mathcal{P}}^v$. Since u was arbitrary, it follows that the normal cone of every vertex of \mathcal{Z} is contained in the normal cone of some vertex of \mathcal{P} and we are done by the aforementioned standard result. \square

Finally, we need the following statement about edge-directions of linear images of polytopes.

Lemma 2.4. *Let $\mathcal{Q} := \omega(\mathcal{P})$ be the image of a polytope \mathcal{P} under a linear map ω . Then every direction q of an edge of \mathcal{Q} is the image under ω of some direction p of an edge of \mathcal{P} .*

Proof. Let q be a direction of an edge $[x, y]$ of \mathcal{Q} . Consider the face $F := \omega^{-1}([x, y])$ of \mathcal{P} . Let V be the set of vertices of F and let $U = \{u \in V : \omega(u) = x\}$; as $x \neq y$, $U \neq V$. Further, as the graph of F is connected there must be an edge $[u, v]$ of F , and hence of \mathcal{P} , for some $u \in U$ and $v \in V \setminus U$. Then $\omega(v) \in (x, y]$ hence $\omega(v) = x + \alpha q$ for some $\alpha \neq 0$. Therefore $q = (1/\alpha)(\omega(v) - \omega(u)) = \omega((1/\alpha)(v - u)) = \omega(p)$ with $p := (1/\alpha)(v - u)$, a direction of the edge $[u, v]$ of \mathcal{P} . \square

2.2. Reduction of Convex to Linear Combinatorial Optimization

We now reduce convex to linear combinatorial optimization. We make the following assumptions. The ground set is $N := \{1, \dots, n\}$ and the family $\mathcal{F} \subseteq 2^N$ is *edge-guaranteed*, which means that it is nonempty and comes with an explicit set $E = \{e^1, \dots, e^m\} \subseteq \mathbb{Q}^n$ of vectors guaranteed to contain a direction of each edge of the polytope $\mathcal{P}^{\mathcal{F}} = \text{conv}\{\mathbf{1}_F : F \in \mathcal{F}\}$ associated with \mathcal{F} . In this subsection we assume that \mathcal{F} is presented by a *linear combinatorial optimization oracle* that, given $b : N \rightarrow \mathbb{Q}$, returns a family member $F \in \mathcal{F}$ of maximum weight $b(F) = \sum_{j \in F} b(j)$. The convex functional $c : \mathbb{Q}^d \rightarrow \mathbb{Q}$ is presented by an *evaluation oracle* that, given $x \in \mathbb{Q}^d$, returns the value $c(x)$. The weighting $w : N \rightarrow \mathbb{Q}^d$ is given by an explicit list $w(1), \dots, w(n) \in \mathbb{Q}^d$. We consider d as fixed; otherwise, as mentioned before, the problem becomes intractable at once even for the simple family $\mathcal{F} = 2^N$, see Example 3.1 below. The following algorithm, applied to the data above, provides a reduction of convex to linear combinatorial optimization.

Algorithm 2.5. Given data as above, perform the following steps:

1. Consider the linear map $\omega : \mathbb{Q}^n \rightarrow \mathbb{Q}^d$ defined by $\omega(x) := \sum_{j=1}^n w(j)x_j$.
 - (a) Compute the image $\omega(E) = \{\omega(e^1), \dots, \omega(e^m)\}$ of E under ω .
2. Consider the zonotope $\mathcal{Z} := \text{zone}(\omega(E)) = \sum_{i=1}^m [-\omega(e^i), \omega(e^i)]$ in d -space \mathbb{Q}^d .
 - (a) Compute the list $\{u^1, \dots, u^k\}$ of all vertices of \mathcal{Z} .
 - (b) For each u^i compute a linear functional $a^i \in \mathcal{C}_{\mathcal{Z}}^{u^i}$ in the normal cone of \mathcal{Z} at u^i .
3. (a) For each a^i compute $b^i : N \rightarrow \mathbb{Q}$ defined by $b^i(j) := a^i \cdot w(j) = \sum_{t=1}^d a_t^i w(j)_t$.
 - (b) For each b^i query the oracle of \mathcal{F} and obtain $F^i \in \mathcal{F}$ of maximum weight $b^i(F^i)$.
 - (c) For each F^i query the oracle of c and obtain the value $c(w(F^i)) = c(\sum_{j \in F^i} w(j))$.
4. Output $F^i \in \mathcal{F}$ of maximum value $c(w(F^i))$ among F^1, \dots, F^k .

The following theorem is stated for $d \geq 3$; as for Lemma 2.2, a similar statement holds for $d \leq 2$ as well but with arithmetic complexity $O(nm^d)$ (but same query complexity $O(m^{d-1})$).

Theorem 2.6. For $d \geq 3$, Algorithm 2.5 solves the convex combinatorial optimization problem with data as above in strongly polynomial oracle time using $O(nm^{d-1})$ arithmetic operations and $O(m^{d-1})$ queries of the linear combinatorial optimization oracle of \mathcal{F} and evaluation oracle of c .

Proof. First we justify the algorithm. Recall the polytope $\mathcal{P}_w^{\mathcal{F}} = \text{conv}\{w(F) : F \in \mathcal{F}\}$. As

$$\mathcal{P}_w^{\mathcal{F}} = \text{conv}\{w(F) : F \in \mathcal{F}\} = \text{conv}\{\omega(\mathbf{1}_F) : F \in \mathcal{F}\} = \omega(\mathcal{P}^{\mathcal{F}}),$$

$\mathcal{P}_w^{\mathcal{F}}$ is the image of $\mathcal{P}^{\mathcal{F}}$ under the linear map ω defined in step 1 of the algorithm. Thus, by Lemma 2.4, the image $\omega(E)$ of E under ω contains a direction of every edge of $\mathcal{P}_w^{\mathcal{F}}$. Therefore, by Lemma 2.3, the zonotope

$$\mathcal{Z} := \text{zone}(\omega(E)) = \sum_{i=1}^m [-\omega(e^i), \omega(e^i)]$$

defined in step 2 is a refinement of $\mathcal{P}_w^{\mathcal{F}}$. Now consider any vertex v of $\mathcal{P}_w^{\mathcal{F}}$. Since \mathcal{Z} refines $\mathcal{P}_w^{\mathcal{F}}$, the normal cone of $\mathcal{P}_w^{\mathcal{F}}$ at v contains the normal cone of \mathcal{Z} at some vertex u^i of \mathcal{Z} found in step 2(a). This implies that the corresponding linear functional $a^i \in \mathcal{C}_{\mathcal{Z}}^{u^i}$ found in step 2(b) is maximized uniquely over $\mathcal{P}_w^{\mathcal{F}}$ at v . Now, consider the corresponding weighting b^i defined in step 3(a). As v is the unique maximizer of a^i over $\mathcal{P}_w^{\mathcal{F}} = \text{conv}\{w(F) : F \in \mathcal{F}\}$, we have

$$b^i(F) = \sum_{j \in F} a^i \cdot w(j) = a^i \cdot \sum_{j \in F} w(j) = a^i \cdot w(F) \leq a^i \cdot v$$

for each $F \in \mathcal{F}$, with equality if and only if $w(F) = v$. Thus, the member $F^i \in \mathcal{F}$ obtained in step 3(b) from the linear combinatorial optimization oracle of \mathcal{F} when maximizing b^i has $v = w(F^i)$. It follows that every vertex of $\mathcal{P}_w^{\mathcal{F}}$ equals $w(F^i)$ for some F^i obtained in step 3(b). Since c is convex, the maximum value $c(w(F))$ of $F \in \mathcal{F}$ occurs at some vertex $v = w(F^i)$ of $\mathcal{P}_w^{\mathcal{F}}$. Thus the member $F^i \in \mathcal{F}$ output by the algorithm in step 4, which has maximum value $c(w(F^i))$ among the values computed in step 3(c), is an optimal solution to the convex combinatorial optimization problem.

Next we verify the claimed complexity, where, as explained, d is considered fixed. The computation of the linear image $\omega(E)$ in step 1(a) takes $O(dnm) = O(nm)$ operations. By Lemma 2.1, the number of vertices of the zonotope \mathcal{Z} defined in step 2 satisfies $k = O(m^{d-1})$, and the computation of these vertices u^i and of the corresponding linear functionals a^i in steps 2(a) and 2(b) requires $O(m^{d-1})$ operations by Lemma 2.2. The number of queries in step 3(b) of the oracle of \mathcal{F} and in step 3(c) of the oracle of c are $k = O(m^{d-1})$ as claimed. The computation of each b^i in step 3(a) and of each $w(F^i)$ in step 3(c) takes $O(dn) = O(n)$ operations totalling together over all i to $O(kn) = O(nm^{d-1})$ arithmetic operations. Finally, the arithmetic complexity of finding the maximum among the k values $c(w(F^i))$ in step 4 is $O(k) = O(m^{d-1})$. Thus, the dominant arithmetic complexity is $O(nm^{d-1})$ as claimed. \square

As the proof shows, convex combinatorial optimization is solved by enumerating all vertices of the polytope $\mathcal{P}_w^{\mathcal{F}}$ and picking the best. While each vertex of $\mathcal{P}_w^{\mathcal{F}}$ is the image under w of some vector $\mathbf{1}_F$, with $F \in \mathcal{F}$, the difficulty is that the number of sets in \mathcal{F} is typically exponential in n and hence it is generally impossible to construct $\mathcal{P}_w^{\mathcal{F}}$ directly in polynomial arithmetic complexity (in particular, each $\mathbf{1}_F$ is a vertex of $\mathcal{P}^{\mathcal{F}}$). The efficient construction of $\mathcal{P}_w^{\mathcal{F}}$ is made possible by the given set of edge-directions of $\mathcal{P}^{\mathcal{F}}$ and by proceeding, indirectly, through the zonotope \mathcal{Z} that refines $\mathcal{P}_w^{\mathcal{F}}$. While the number of vertices of \mathcal{Z} can be much larger than that of $\mathcal{P}_w^{\mathcal{F}}$, the vertices of \mathcal{Z} can be better controlled and this leads to the polynomial complexity bound. So if the polytope $\mathcal{P}^{\mathcal{F}}$ of a family \mathcal{F} admits a relatively small set containing a direction of each edge which can be efficiently constructed or even characterized, then the problem is efficiently reducible.

2.3. Generic Solution of the Linear Combinatorial Optimization Counterparts

We now discuss how to realize an oracle that will repeatedly solve each of the linear combinatorial optimization counterparts queried upon in Algorithm 2.5. As before, we assume that our family \mathcal{F} is edge-guaranteed and hence nonempty and comes with an explicit set $E = \{e^1, \dots, e^m\}$ containing a direction of each edge of $\mathcal{P}^{\mathcal{F}}$. We assume moreover that the family comes with one member $F^0 \in \mathcal{F}$ to start with. We consider the following three oracle presentations of \mathcal{F} :

- *Membership oracle*: when queried about $F \subseteq N$, this oracle asserts whether or not $F \in \mathcal{F}$.
- *Augmentation oracle*: when queried about $F \in \mathcal{F}$ and $b: N \rightarrow \mathbb{Q}$, this oracle returns a family member $\hat{F} \in \mathcal{F}$ with $b(\hat{F}) > b(F)$ or asserts that F has maximum weight in \mathcal{F} .
- *Linear combinatorial optimization oracle*: when queried about $b: N \rightarrow \mathbb{Q}$, this oracle returns a family member $F \in \mathcal{F}$ of maximum weight $b(F)$.

Lemma 2.7. *For any edge-guaranteed family, a membership oracle enables us to simulate an augmentation oracle in strongly polynomial oracle time.*

Proof. Without loss of generality, assume that each e^i is a $\{-1, 0, 1\}$ -vector. The simulation is simple. Consider a query about F and $b: N \rightarrow \mathbb{Q}$. Call an edge-direction e^i *improving* if $b(e^i) = \sum_{j=1}^n b(j)e_j^i > 0$; call it *admissible at F* if $\mathbf{1}_F + e^i$ is a $\{0, 1\}$ -vector whose support $F^i := \text{supp}(\mathbf{1}_F + e^i)$ is in \mathcal{F} . If there is an edge-direction e^i which is both improving and admissible then return F^i ; otherwise assert that F has maximum weight in \mathcal{F} . The simulation works correctly since, as is well known, a vertex u is not a maximizer of a linear functional b over a polytope if and only if the polytope has an edge $[u, v]$ for some vertex v with $b(v) > b(u)$. \square

The next lemma is from [24] and [46]; see [45] for the state of the art on this line of research. The lemma involves a computationally heavy Diophantine approximation step [20] and a scaling step [19]. We include an outline of the proof, which is relevant for the discussion in the next subsection.

Lemma 2.8. *For any family, an augmentation oracle enables us to simulate a linear combinatorial optimization oracle in strongly polynomial oracle time.*

Proof. We outline the simulation. Consider the query about $b: N \rightarrow \mathbb{Q}$. The Diophantine approximation step (see Theorem 3.3 of [20]) replaces b by $a: N \rightarrow \mathbb{Z}$ with the following two properties: first, it is equivalent to b in that, for any pair $F, G \subseteq N$, it satisfies $a(F) \leq a(G)$ if and only if $b(F) \leq b(G)$; and, second, the maximum number of bits $k := 1 + \max_{j \in N} \lceil \log |a(j)| \rceil$ in the binary representation of the weight under a of any element is polynomial in n .

The scaling step (inspired by [19]) is the following. Applying a simple transformation (see [46]) we may assume a is nonnegative. Following the proof of Theorem 9.2 of [24], for $i = 0, \dots, k$, starting with $F^i \in \mathcal{F}$, find a maximizing $F^{i+1} \in \mathcal{F}$ with respect to the

weighting $a^i := \lfloor 2^{i-k} a \rfloor$. As shown in [24] or [46], each F^{i+1} is obtained from F^i by calling the augmentation oracle at most n times. Thus, the maximizer F^{k+1} of $a = a^k$ is found using at most kn calls, and since k is polynomial in n , the desired maximizer of b is obtained in strongly polynomial oracle time. \square

We can now prove Theorem 1.1. As discussed before, the family $\mathcal{F} \subseteq 2^N$ comes with one explicit $F^0 \in \mathcal{F}$ and an explicit set $E = \{e^1, \dots, e^m\} \subset \mathbb{Q}^n$ containing a direction of each edge of $\mathcal{P}^{\mathcal{F}}$, and is presented by a membership oracle. The complexity is measured in terms of n and m .

Theorem 1.1. *For any fixed d , there is a strongly polynomial oracle time algorithm solving convex combinatorial optimization over any edge-guaranteed family presented by a membership oracle.*

Proof. Theorem 2.6 guarantees that Algorithm 2.5 solves the convex combinatorial optimization problem over \mathcal{F} efficiently using a linear combinatorial optimization oracle which, by Lemmas 2.7 and 2.8, can be efficiently simulated from the membership oracle presenting \mathcal{F} . \square

We conclude this subsection with several important remarks which lead to the discussion in the next subsection. First, the complexity behind Theorem 1.1 is quite horrendous: for each linear combinatorial optimization counterpart invoked by Algorithm 2.5, an application of the Diophantine approximation step which takes $O(n^8)$ arithmetic operations [20] is required. However, improved complexity bounds follow from Theorem 2.6 when Algorithm 2.5 is used with a more efficient linear combinatorial optimization oracle whenever a particular family admits one.

Second, what about real data and real arithmetic computation (where pairs of real numbers can be added, multiplied, or compared in unit time)? Algorithm 2.5 remains valid and polynomial and the analog of Theorem 2.6 (with “strongly polynomial oracle time” replaced by “polynomially many real arithmetic operations and queries”) holds. So does the conversion of the membership oracle to the augmentation oracle manifested by Lemma 2.7 above. However, the proof of both parts of Lemma 2.8 (scaling and Diophantine approximation) breaks down for real data, and the conversion of the augmentation oracle to the optimization oracle is no longer available.

Can these obstacles be waved and does the real analog of Theorem 1.1 remain valid? Our families are edge-guaranteed, which is stronger than having a test set (see [47] and references therein) and even more so than having a mere augmentation oracle: can we take advantage of that and simulate standard linear combinatorial optimization directly and more efficiently, avoiding scaling and Diophantine approximation? We discuss some of these issues next.

2.4. On the Hirsch Conjecture and the Klee–Minty Problem for $(0, 1)$ -Polytopes

A form of the *Hirsch conjecture*, open to date, asks whether the diameter of (the graph of) every n -polytope with f facets is bounded above by a polynomial in n and f ; for

all one knows, the linear upper bound $f - n$ may suffice, see [33] and [35]. Also open is the analogous form of the *monotone Hirsch conjecture* asking whether the shortest increasing path under any linear functional from any vertex to some maximizing vertex is polynomially bounded in n and f . Both variants are true for $(0, 1)$ -polytopes [38] as well as for some more general classes of integer polytopes [14], [37]. The following slightly stronger form, relevant to the discussion below, holds:

Lemma 2.9. *Any $(0, 1)$ n -polytope P admits, under any linear functional, a nondecreasing path from any vertex u to any maximizing vertex w , of length at most n using no edge-direction twice.*

Proof. The claim being trivial for $n = 0$, we proceed by induction. If u and w lie on a common proper face of P then induction takes over. Otherwise, there is a nondecreasing arc (u, v) ; pick any i with $u_i \neq v_i$; then $v_i = w_i$ and hence, by induction, there is a nondecreasing (v, w) -path of length at most $n - 1$ using no edge-direction twice on the face $F := \{x \in P : x_i = w_i\}$. As no edge of F can have direction $v - u$, this path preceded by the arc (u, v) gives the desired (u, w) -path. \square

The *effective Hirsch conjecture* asks, broadly, whether a monotone path could be efficiently traced. To make things precise, the presentation of the polytope has to be specified. For instance, tracing such a path in strongly polynomial time for polytopes presented by linear inequalities would imply a strongly polynomial time algorithm for linear programming via the simplex method which does not seem likely; but tracing it in subexponential time is possible [33]. A natural question is: how long can an *arbitrary* increasing path be? A classical construction by Klee and Minty [36] transforms the n -cube so as to admit increasing paths of exponential length 2^n . However, $(0, 1)$ -polytopes are very special, as Lemma 2.9 shows. How long, then, can an arbitrary increasing path in a $(0, 1)$ -polytope be? We now show that, unfortunately, $(0, 1)$ -polytopes admit such paths of length exponential in the dimension as well: in this sense, the following lemma can be regarded as a $(0, 1)$ analog of the Klee–Minty cube; we thank Tal Raviv for a related discussion.

Lemma 2.10. *For every n there is a $(0, 1)$ -polytope of dimension less than $\frac{1}{4}n^4$ with $n!$ vertices that admits a Hamiltonian (and hence $n!$ -long) nondecreasing path under every linear functional.*

Proof. The *Young polytope* $Y_{n-2,2}$ is the convex hull of all $\binom{n}{2} \times \binom{n}{2}$ matrices of permutations of edges of the complete graph K_n induced by the $n!$ permutations of its vertices. For instance, the matrix corresponding to the permutation of vertices $\sigma = (1, 2, 3, 4)$ (in cycle notation) is

$$\Sigma = \begin{matrix} & \begin{matrix} 12 & 13 & 14 & 23 & 24 & 34 \end{matrix} \\ \begin{matrix} 12 \\ 13 \\ 14 \\ 23 \\ 24 \\ 34 \end{matrix} & \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \end{matrix}.$$

The claims about the dimension and number of vertices of $Y_{n-2,2}$ are obvious. In [39] it was shown that $Y_{n-2,2}$ is 2-neighborly, that is, its graph is the complete graph $K_{n!}$: the very existence of 2-neighborly $(0, 1)$ -polytopes is an amazing fact in itself! It follows that if a is any linear functional, then any ordering $v_1, \dots, v_{n!}$ of the vertices of $Y_{n-2,2}$ satisfying $a(v_1) \leq \dots \leq a(v_{n!})$ gives a Hamiltonian (and hence exponentially long) nondecreasing path under a . \square

Here, however, we are especially interested in the polytopes $\mathcal{P}^{\mathcal{F}}$ of *edge-guaranteed* families. Lemmas 2.7 and 2.8 imply that for such a family, a monotone path can be traced in time strongly polynomial in m and n , alas, for *rational* functionals only, and using the heavy Diophantine approximation procedure. Can we do better? What is the maximal length $I(n, m)$ of any increasing path in any n -dimensional $(0, 1)$ -polytope with m pairwise nonproportional edge-directions?

2.5. Edge-Well-Behaved Classes

In most applications, in particular all of those discussed in Section 3, one is concerned with a class of families possessing some unifying structure. It is therefore useful to make some formal definitions regarding such classes and then use it to obtain a suitable corollary of Theorem 1.1. For $n \geq 0$ let $N = \{1, \dots, n\}$ as before and let \mathcal{U}_n be the set of all families with ground set N ,

$$\mathcal{U}_n := 2^{\{1, \dots, n\}} = \{\mathcal{F} : \mathcal{F} \subseteq 2^N\}.$$

A *class of families* is a (typically infinite) set of families $\mathcal{C} = \biguplus_{n \geq 0} \mathcal{C}_n$ with $\mathcal{C}_n \subseteq \mathcal{U}_n$ for all n .

Definition 2.11. A class \mathcal{C} is *edge-well-behaved* if there is a polynomial time algorithm that, given n , produces a set $E_n = \{e^1, \dots, e^{m(n)}\} \subseteq \{-1, 0, 1\}^n$ with respect to which every $\mathcal{F} \in \mathcal{C}_n$ is edge-guaranteed. In particular, $m(n)$ is polynomial in n and each $\mathcal{F} \in \mathcal{C}$ is nonempty.

While the existence of such a “uniform” polynomial time algorithm that produces sets containing edge-directions for the polytopes of all families in the class may seem a strong assumption, we will see in Section 3 that in many applications such an algorithm is readily available. Also, the assumption that the edge-directions are $(-1, 0, 1)$ -valued is not restrictive since, for $(0, 1)$ -polytopes, each edge is a difference of two vertices and hence admits a $(-1, 0, 1)$ -direction. The next corollary follows at once from Theorem 1.1; here the complexity is in terms of n only.

Corollary 2.12. *Fix any d . Then for any edge-well-behaved class there is a strongly polynomial oracle time algorithm that solves the convex combinatorial optimization problem over any family in the class which is presented by a membership oracle.*

While this statement may seem a reformulation of Theorem 1.1, it is natural and useful in uniformly establishing the polynomial solvability in all of the applications discussed in Section 3.

2.6. Projection Representation and Circuits

We conclude Section 2 by discussing a useful setup that helps in controlling edge-directions, and which is used and demonstrated in the application given in Section 3.2. A *circuit* of an $r \times s$ matrix A is a nonzero solution $z \in \mathbb{Q}^s$ of the system $Az = 0$ whose support is inclusion-minimal. It is known (see Exercise 10.14 of [44]) that any nonzero solution of $Az = 0$ has a *conformal circuit decomposition*, i.e., can be expressed as $z = \sum_i \alpha_i z^i$ with the α_i positive scalars and the z^i pairwise nonproportional circuits such that $z_j^i z_j^j > 0$ for all i and all $j \in \text{supp}(z^i)$. Consider the *standard polytope* $\mathcal{P} = \{x \in \mathbb{Q}^s : Ax = b, l \leq x \leq u\}$ defined by A , right-hand side $b \in \mathbb{Q}^r$, lower bound $l \in \mathbb{Q}^s$, and upper bound $u \in \mathbb{Q}^s$. We provide a short proof of the following useful property of edge-directions of the standard polytope (see [41] for a refinement of this property which characterizes edge-directions).

Lemma 2.13. *Each edge-direction of a standard \mathcal{P} is a circuit of its defining matrix A .*

Proof. Consider any $x, y \in \mathcal{P}$. Then $A(y - x) = 0$ so $y - x$ admits a conformal circuit decomposition $y - x = \sum_i \alpha_i z^i$. It is then not hard to verify that for every circuit z^i participating in that decomposition, both $x + \alpha_i z^i$ and $y - \alpha_i z^i$ satisfy the lower and upper bounds and hence are in \mathcal{P} . They belong, moreover, to any face F containing both x and y . Indeed, pick any c in the normal cone \mathcal{C}_F^A : then $c \cdot x \geq c \cdot (x + \alpha_i z^i)$ which implies $c \cdot z^i \leq 0$, and $c \cdot y \geq c \cdot (y - \alpha_i z^i)$ which implies $c \cdot z^i \geq 0$. It follows that $c \cdot z^i = 0$ and hence $c \cdot (x + \alpha_i z^i) = c \cdot x$ and $c \cdot (y - \alpha_i z^i) = c \cdot y$ implying $x + \alpha_i z^i, y - \alpha_i z^i \in F$. Now, if the decomposition $y - x = \sum_i \alpha_i z^i$ involves more than one circuit, say z^1, z^2 , then any face containing x, y contains the three noncollinear points $x, x + \alpha_1 z^1, x + \alpha_2 z^2$ and hence is not an edge. So if $[x, y]$ is an edge of \mathcal{P} then $y - x = \alpha z$ for some circuit z . Any direction of that edge is a nonzero multiple of z and hence a circuit of A . \square

Lemma 2.13 implies that any inclusion-maximal set Z of pairwise nonproportional circuits of A contains a direction of each edge of \mathcal{P} . So if the size of A is $r \times s$ then \mathcal{P} admits such a set Z with no more than $\binom{s}{r}$ elements. Every polytope \mathcal{Q} is the linear image of a standard polytope: if $\mathcal{Q} = \{x : Bx \leq b\}$ is a description by inequalities then, adding a suitable “slack” vector y , we get $\mathcal{Q} = \varphi(\mathcal{P})$ with $\mathcal{P} = \{(x, y) : Bx + Iy = b, y \geq 0\}$ and with φ the “ y forgetting” projection $\varphi(x, y) = x$. In particular, the polytope $\mathcal{P}^{\mathcal{F}}$ of any family \mathcal{F} is the linear image $\mathcal{P}^{\mathcal{F}} = \varphi(\mathcal{P})$ of a standard polytope $\mathcal{P} = \{x \in \mathbb{Q}^s : Ax = b, l \leq x \leq u\}$. Typically the number of inequalities describing $\mathcal{P}^{\mathcal{F}}$ is exponentially large and hence so is the dimension of \mathcal{P} , but when \mathcal{P} has small dimension, we can benefit from such a “projection representation” in two ways as follows.

First, if the defining matrix A admits an efficiently determinable set $Z = \{z^1, \dots, z^m\} \subset \mathbb{Q}^s$ containing a scalar multiple of each circuit of A then, by Lemmas 2.4 and 2.13, its image $E := \varphi(Z)$ contains a direction of each edge of $\mathcal{P}^{\mathcal{F}}$, making \mathcal{F} an edge-guaranteed family. Second, linear combinatorial optimization over \mathcal{F} can be “lifted” to linear programming over the polytope \mathcal{P} , giving an alternative way to Section 2.3 for solving the counterparts called upon by Algorithm 2.5.

3. Some Applications

3.1. Some Direct Applications

Here we give two examples where the set of edge-directions can be directly determined and used.

Example 3.1 (Positive Semidefinite Quadratic Assignment). The quadratic assignment problem is the following: given a real $n \times n$ matrix M , find $x \in \{0, 1\}^n$ maximizing the quadratic form $x^T M x$ induced by M ; see [43] for an overview of this problem and its applications. We consider the instance where M is positive semidefinite, in which case it can be assumed to be presented as $M = W^T W$ with W a given $d \times n$ matrix. If the rank d of W and M is variable then this problem is NP-hard [27]. For fixed d it is polynomial time solvable [2].

When W is rational, the problem can be modeled as convex combinatorial optimization with the following data: the family is the entire power set $\mathcal{F} = 2^N$ of N with the natural correspondence $\mathcal{F} \leftrightarrow \{0, 1\}^n$; the weight of $j \in N$ is the j th column $w(j) := W^j$ of the matrix W ; and $c: \mathbb{Q}^d \rightarrow \mathbb{Q}: x \mapsto \|x\|^2$ is the squared standard l_2 norm. Indeed, for each $F \in \mathcal{F}$ we then have $\mathbf{1}_F^T W^T W \mathbf{1}_F = c(w(F))$.

Now, the polytope of the family \mathcal{F} here is just the n -cube $\mathcal{P}^{\mathcal{F}} = [0, 1]^n$; therefore the trivially computable set of n standard unit vectors $E := \{\mathbf{1}_1, \dots, \mathbf{1}_n\}$ contains a direction of each edge. Thus, the class of all such families is edge-well-behaved with $m(n) = n$ and Corollary 2.12 applies and guarantees the efficient solution. Here, one obtains a faster solution by using Algorithm 2.5 together with a linear combinatorial optimization oracle realized by simple sign checking as follows: given $b: N \rightarrow \mathbb{Q}$, a member $F \in \mathcal{F}$ maximizing $b(F)$ is simply $F := \{j : b(j) > 0\}$.

Example 3.2 (Convex Matroid Optimization). This problem (see [29] and [40]) is the special case of convex combinatorial optimization where \mathcal{F} is either the collection \mathcal{B} of bases or the collection \mathcal{I} of independent sets of a matroid over N . It generalizes classical matroid optimization, first studied in [18], and has a rich modeling power on its own: useful matroids include the forest matroid of a graph and, more generally, the matroid of linear dependencies of a matrix over a field. For us it suffices that the matroid is presented by a membership oracle for \mathcal{F} .

It can be derived from the matroid-bases-axioms that the trivially computable set $D := \{\mathbf{1}_i - \mathbf{1}_j : 1 \leq i < j \leq n\}$ of $\binom{n}{2}$ differences of unit vectors contains a direction of each edge of the polytope $\mathcal{P}^{\mathcal{B}}$ of the family \mathcal{B} of bases. Likewise, it can be derived from the matroid-independence-axioms that the $\binom{n+1}{2}$ -element union $D \cup E$ of D and the set $E := \{\mathbf{1}_1, \dots, \mathbf{1}_n\}$ of unit vectors contains a direction of each edge of the polytope $\mathcal{P}^{\mathcal{I}}$ of the family \mathcal{I} of independent sets. Thus, the class of all such families \mathcal{B} (respectively, families \mathcal{I}) is edge-well-behaved with $m(n) = \binom{n}{2}$ (respectively, $m(n) = \binom{n+1}{2}$), and Corollary 2.12 applies and guarantees the efficient solution.

Here, too, one obtains a faster solution by using Algorithm 2.5 together with a linear combinatorial optimization oracle over $\mathcal{F} = \mathcal{B}$ or $\mathcal{F} = \mathcal{I}$ which is efficiently realizable from a membership oracle for \mathcal{F} using the classical greedy algorithm (see [13] and [18]) that, given $b: N \rightarrow \mathbb{Q}$ makes use of sorting the values $b(j)$ to find the lexicographically b -largest member $F \in \mathcal{F}$ which can be shown to be the one maximizing $b(F)$.

3.2. Shaped Vector Partitioning

The *shaped partition problem* concerns the partitioning of a multiset $V = \{v^1, \dots, v^n\}$ of n vectors in d -space into p parts so as to maximize an objective function which is convex on the sum of vectors in each part, subject to constraints on the number of elements in each part. To describe the problem precisely we need some notation. A p -partition of the index set $\{1, \dots, n\}$ of V is an ordered tuple $\pi = (\pi_1, \dots, \pi_p)$ of pairwise disjoint sets whose union is $\{1, \dots, n\}$. The *shape* of a partition is the tuple of cardinalities of its parts, $|\pi| := (|\pi_1|, \dots, |\pi_p|)$. In addition to the set of vectors V , the data includes vectors $l, u \in \{0, 1, \dots, n\}^p$ with $l \leq u$ providing lower and upper bounds on the shape of admissible partitions. With each partition π is associated a $d \times p$ matrix

$$V^\pi := \left[\left(\sum_{i \in \pi_1} v^i \right), \dots, \left(\sum_{i \in \pi_p} v^i \right) \right] = \sum_{j=1}^p \left(\sum_{i \in \pi_j} v^i \right) \mathbf{1}_j^T,$$

whose j th column is the sum (representing the “total value”) of vectors assigned to the j th part. The data also includes a convex functional $c: \mathbb{Q}^{d \times p} \rightarrow \mathbb{Q}$ on $d \times p$ matrices which “weighs together” the sums of vectors in the various parts. The problem is to find a p -partition π whose shape satisfies the lower and upper bounds $l \leq |\pi| \leq u$ and which maximizes the value $c(V^\pi)$.

Shaped partition problems have applications in diverse fields such as clustering, inventory, reliability, and more—see [7], [10], [12], [30], [32], [43], and references therein. Here is a typical example.

Example 3.3 (Minimal Variance Clustering). This is the following problem, which has numerous applications in the analysis of statistical data: given n sample points v^1, \dots, v^n in d -space, group the points into p clusters π_1, \dots, π_p so as to minimize the sum of cluster variances:

$$\sum_{j=1}^p \frac{1}{|\pi_j|} \sum_{i \in \pi_j} \left\| v^i - \left(\frac{1}{|\pi_j|} \sum_{i \in \pi_j} v^i \right) \right\|^2.$$

We consider the instance where there are $n = p \cdot m$ points and the clustering sought is *balanced*, that is, the clusters should have equal size m . Suitable manipulation of the sum of variances shows that the problem is equivalent to a shaped partition problem with the lower and upper bounds $l := u := (m, \dots, m)$ (forcing the single shape $|\pi| = (m, \dots, m)$ on partitions), and with the convex functional (to be maximized) simply as the square of the l_2 norm on $d \times p$ matrices, given by $c: \mathbb{Q}^{d \times p} \rightarrow \mathbb{Q}: X \mapsto \|X\|^2 = \sum_{i=1}^d \sum_{j=1}^p |X_{i,j}|^2$.

If either the dimension d or the number of parts p is variable, then the shaped partition problem instantly captures NP-hard problems and hence is presumably intractable [30]. Therefore, it is interesting to study the worst case arithmetic complexity in terms of the number n of points with both d and p fixed. In the special case where there are no shape restrictions (partitions of all shapes are admissible), an upper bound of $O(n^{d(p-1)-1})$ on the complexity is given in [42] and a quite compatible lower bound of $\Omega(n^{\lfloor (d-1)/2 \rfloor p})$

is in [5]. In the more general case where arbitrary sets of shapes are allowed, the best upper bound to date is $O(n^{dp^2})$ from [30]; while a matching lower bound is unknown, the lower bound $O(n^{d\binom{p}{2}})$ from [3] on the related number of *separable partitions* indicates that the quadratic term p^2 in the exponent may be unavoidable.

We now show how to solve the shaped partition problem efficiently using our framework. We begin by modeling it as a convex combinatorial optimization problem. The ground set is taken to be $N := \{(i, j) : 1 \leq i \leq n, 1 \leq j \leq p\}$. Each p -partition $\pi = (\pi_1, \dots, \pi_p)$ is encoded as the set $F_\pi := \{(i, j) : i \in \pi_j\} \subseteq N$. The family consists of all such sets corresponding to p -partitions of admissible shapes, $\mathcal{F} := \{F_\pi : l \leq |\pi| \leq u\}$. The weight of element $(i, j) \in N$ is the $d \times p$ matrix $w(i, j) := v^i \mathbf{1}_j^T$ whose j th column is v^i and whose other columns are zero. Finally, the convex functional is simply the given one c defined on $d \times p$ matrix space. It is not hard to verify that this indeed casts the shaped partition problem as a convex combinatorial optimization problem with a ground set of size $|N| = np$ and weight vectors (matrices) in dimension dp .

To show that the class of all such families is edge-well-behaved we discuss the family polytope $\mathcal{P}^\mathcal{F} \subset \mathbb{Q}^{n \times p}$. The indicator of a family member $F_\pi \in \mathcal{F}$ is the $(0, 1)$ -valued $n \times p$ matrix $\mathbf{1}_\pi$ whose (i, j) th entry equals 1 precisely when $i \in \pi_j$. The polytope $\mathcal{P}^\mathcal{F}$ admits a simple projection-representation as follows. Consider $(n+1) \times p$ matrices whose rows are indexed by $\{0, 1, \dots, n\}$. Define lower and upper bound matrices L, U in terms of the given vectors l, u as follows: for $j = 1, \dots, p$ set $L_{i,j} := 0$, $U_{i,j} := 1$ if $1 \leq i \leq n$ and $L_{0,j} := n - u_j$, $U_{0,j} := n - l_j$. Let \mathcal{P} be the *transportation polytope* defined by this data, which is the following standard polytope:

$$\mathcal{P} := \left\{ X \in \mathbb{Q}^{(n+1) \times p} : \sum_{j=1}^p X_{i,j} = 1 (1 \leq i \leq n), \right. \\ \left. \sum_{i=0}^n X_{i,j} = n (1 \leq j \leq p), L \leq X \leq U \right\}.$$

Then $\mathcal{P}^\mathcal{F} = \varphi(\mathcal{P})$ with $\varphi: \mathbb{Q}^{(n+1) \times p} \rightarrow \mathbb{Q}^{n \times p}$ the projection erasing the zeroth row of a matrix.

Let $K_{n+1,p}$ be the complete bipartite graph with edge set $\{(i, j) : 0 \leq i \leq n, 1 \leq j \leq p\}$ corresponding to this transportation system. Each circuit of $K_{n+1,p}$ gives an $(n+1) \times p$ matrix supported on that circuit, with values ± 1 alternating along the edges of the circuit and 0 elsewhere. It is well known that each circuit of the $(n+1) \times (n+1)p$ matrix of coefficients of the equation system defining \mathcal{P} is proportional to some such circuit-supported matrix. Let $Z := \{z^1, \dots, z^m\}$ be the set of all such $(n+1) \times p$ matrices corresponding to the $m := \sum_{i=2}^p \frac{1}{2} \binom{p}{i} \binom{n+1}{i} i! (i-1)!$ distinct circuits of $K_{n+1,p}$. Then the projection $E = \{e^1, \dots, e^m\} := \varphi(Z)$ is a set of $(-1, 0, 1)$ -valued $n \times p$ matrices which, as explained in Section 2.6, contains a direction of each edge of $\mathcal{P}^\mathcal{F}$.

Since p is assumed to be fixed, the set of circuits Z and its projection E are computable in time polynomial in n , and therefore the class of all such families is indeed edge-well-behaved with $m(n) = \sum_{i=2}^p \frac{1}{2} \binom{p}{i} \binom{n+1}{i} i! (i-1)! = O(n^p)$. Thus, Corollary 2.12 applies and guarantees the efficient solution. As in the examples in Section 3.1, here too one can obtain a typically faster solution by using Algorithm 2.5 together with a linear

combinatorial optimization oracle realized by lifting to \mathcal{P} and solving the corresponding transportation problem using available fast algorithms for bipartite network flows, see [1]. These algorithms, however, are not strongly polynomial and do depend on the bit size of the bounds l, u and the costs b^i called repeatedly by Algorithm 2.5.

For *unrestricted* partitioning, that is, shaped partitioning with $l = (0, \dots, 0)$ and $u = (n, \dots, n)$, the characterization of circuits obtained in [41] shows that circuits of $K_{n+1,p}$ which yield circuits of the matrix of coefficients of the equation system defining \mathcal{P} correspond to switching a single item from one part to another. As the number of such circuits is $n \binom{p}{2}$, the class of such families is edge-well-behaved with (improved) $m(n) = n \binom{p}{2}$. Thus, while Theorem 2.6 with $m(n) = O(n^p)$ and dimension dp implies a complexity bound of $n^{O(dp^2)}$ on the general shaped partition problem, in line with [30], with $m(n) = n \binom{p}{2}$ and same dimension dp it implies the improved bound of $n^{O(dp)}$ on the complexity of the unrestricted partition problem, in line with [42].

4. Concluding Remarks

In this article we have defined the convex combinatorial optimization problem and shown that it can be solved in strongly polynomial time for edge-guaranteed families and for edge-well-behaved classes of families. We have demonstrated several natural and broad applications that indeed give rise to edge-well-behaved classes and therefore are efficiently solvable through our framework.

The polynomial time solvability of linear combinatorial optimization for *facet-well-behaved* classes via the Ellipsoid method [25], [34] has stimulated over the years a broad body of work on the identification and characterization of such classes. A major research program called upon by this paper is an analogous identification and characterization of *edge-well-behaved* classes of combinatorial families, for which our framework automatically yields strongly polynomial time solvability of convex combinatorial optimization.

Some more specific questions are discussed within the body of our paper, in particular, those in Sections 2.3 and 2.4 concerning a more efficient generic solution of standard linear combinatorial optimization over edge-guaranteed families and the effective Hirsch conjecture for polytopes of such families. What is the maximal length $I(n, m)$ of any increasing path in any n -dimensional $(0, 1)$ -polytope with m pairwise nonproportional edge-directions? Can we trace such a path efficiently while avoiding scaling and the heavy Diophantine approximation procedure? Can we trace such a path in polynomially many real arithmetic operations for real linear functionals?

For solving the standard linear counterparts of a convex combinatorial optimization problem over a family \mathcal{F} with weighting w , one approach may be to try and augment along edges of the polytope $\mathcal{P}_w^{\mathcal{F}}$ downstairs (see discussion following the proof of Theorem 2.6). While a set of edge-directions of that polytope is available as the projection $\omega(E)$ of the set of edge-directions of $\mathcal{P}^{\mathcal{F}}$, this information is not enough: one needs to know, at any vertex v of $\mathcal{P}_w^{\mathcal{F}}$, which edge-directions $\omega(e^i)$ are admissible at v , and, moreover, “how much to walk”—namely, what is the nonnegative scalar α such that $v + \alpha e^i$ is the new vertex to move to. When this information is available, even in an abstract setting of a suitably defined neighborhood oracle, it may be possible to apply the vertex enumeration methodology of [6]; and in some applications, such as unrestricted vector partitioning, this can be carried out particularly efficiently as in [21].

Finally, as mentioned in the Introduction, there is much room for the study of approximation algorithms for the often intractable convex combinatorial optimization problem for various classes of families, and we hope this article will stimulate research on this yet unexplored ground.

Acknowledgement

The first author is indebted to Lou Billera for introducing him to the theory of convex polytopes.

References

1. R.K. Ahuja, J.B. Orlin, C. Stein, and R.E. Tarjan, Improved algorithms for bipartite network flow, *SIAM Journal on Computing*, 23:906–933, 1994.
2. K. Allemand, K. Fukuda, T.M. Liebling, and E. Steiner, A polynomial case of unconstrained zero-one quadratic optimization, *Mathematical Programming Series A*, 91:49–52, 2001.
3. N. Alon and S. Onn, Separable partitions, *Discrete Applied Mathematics*, 91:39–51, 1999.
4. A. Avidor, Y. Azar, and J. Sgall, Ancient and new algorithms for load balancing in the l_p norm, *Algorithmica*, 29:422–441, 2001.
5. S. Aviran and S. Onn, Momentopes and the vertex complexity of partition polytopes, *Discrete & Computational Geometry*, 27:409–417, 2002.
6. D. Avis and K. Fukuda, A pivoting algorithm for convex hulls and vertex enumeration of arrangements and polyhedra, *Discrete & Computational Geometry*, 8:295–313, 1992.
7. E.R. Barnes, A.J. Hoffman, and U.G. Rothblum, Optimal partitions having disjoint convex and conic hulls, *Mathematical Programming*, 54:69–86, 1992.
8. L.J. Billera and A. Sarangarajan, All $(0, 1)$ -polytopes are travelling salesman polytopes, *Combinatorica*, 16:175–188, 1996.
9. L.J. Billera and B. Sturmfels, Fiber polytopes, *Annals of Mathematics*, 135:527–549, 1992.
10. E. Boros and P.L. Hammer, On clustering problems with connected optima in Euclidean spaces, *Discrete Mathematics*, 75:81–88, 1989.
11. R.C. Buck, Partition of space, *American Mathematical Monthly*, 50:541–544, 1943.
12. A.K. Chakravarty, J.B. Orlin, and U.G. Rothblum, Consecutive optimizers for a partitioning problem with applications to optimal inventory groupings for joint replenishment, *Operations Research*, 33:820–834, 1985.
13. W.J. Cook, W.H. Cunningham, W.R. Pulleyblank, and A. Schrijver, *Combinatorial Optimization*, Wiley, New York, 1997.
14. M. Deza and S. Onn, Lattice-free polytopes and their diameter, *Discrete & Computational Geometry*, 13:59–75, 1995.
15. H. Edelsbrunner, J. O’Rourke, and R. Seidel, Constructing arrangements of lines and hyperplanes with applications, *SIAM Journal on Computing*, 15:341–363, 1986.
16. H. Edelsbrunner, R. Seidel, and M. Sharir, On the zone theorem for hyperplane arrangements, in: *New Results and Trends in Computer Science*, pp. 108–123, Lecture Notes in Computer Science, vol. 555, Springer, Berlin, 1991.
17. 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.
18. J. Edmonds, Matroids and the greedy algorithm, *Mathematical Programming*, 1:127–136, 1971.
19. J. Edmonds and R.M. Karp, Theoretical improvements in algorithmic efficiency of network flow problems, *Journal of the Association for Computing Machinery*, 19:248–264, 1972.
20. A. Frank and E. Tardos, An application of simultaneous Diophantine approximation in combinatorial optimization, *Combinatorica*, 7:49–65, 1987.
21. K. Fukuda, S. Onn, and V. Rosta, An adaptive algorithm for vector partitioning, *Journal of Global Optimization*, 25:305–319, 2003.

22. D. Granot and U.G. Rothblum, The Pareto set of the partition bargaining game, *Games and Economic Behavior*, 3:163–182, 1991.
23. P. Gritzmann and B. Sturmfels, Minkowski addition of polytopes: complexity and applications to Gröbner bases, *SIAM Journal on Discrete Mathematics*, 6:246–269, 1993.
24. M. Grötschel and L. Lovász, Combinatorial optimization, in: *Handbook of Combinatorics*, R. Graham, M. Grötschel, and L. Lovász, eds., pp. 1541–1597, North-Holland, Amsterdam, 1995.
25. M. Grötschel, L. Lovász, and A. Schrijver, *Geometric Algorithms and Combinatorial Optimization*, 2nd edn., Springer, Berlin, 1993.
26. B. Grünbaum, *Convex Polytopes*, Wiley, New York, 1967.
27. L.P. Hammer, P. Hansen, P.M. Pardalos, and D.J. Rader, Maximizing the product of two linear functions in $(0, 1)$ -variables, RUTCOR Research Report, Rutgers University, 1997.
28. E.F. Harding, The number of partitions of a set of n points in k dimensions induced by hyperplanes, *Proceedings of the Edinburgh Mathematical Society*, 15:285–289, 1967.
29. R. Hassin and A. Tamir, Maximizing classes of two-parameter objectives over matroids, *Mathematics of Operations Research*, 14:362–375, 1989.
30. F.K. Hwang, S. Onn, and U.G. Rothblum, A polynomial time algorithm for shaped partition problems, *SIAM Journal on Optimization*, 10:70–81, 1999.
31. F.K. Hwang and U.G. Rothblum, Directional-quasi-convexity, asymmetric Schur-convexity and optimality of consecutive partitions, *Mathematics of Operations Research*, 21:540–554, 1996.
32. F.K. Hwang and U.G. Rothblum, *Partitions: Optimality and Clustering*, World Scientific, Singapore, forthcoming.
33. G. Kalai, Linear programming, the simplex algorithm and simple polytopes, *Mathematical Programming, Series B*, 79:217–233, 1997.
34. L.G. Khachiyan, A polynomial algorithm in linear programming, *Soviet Mathematics Doklady*, 20:191–194, 1979.
35. V. Klee and P. Kleinschmidt, The d -step conjecture and its relatives, *Mathematics of Operations Research*, 12:718–755, 1987.
36. V. Klee and J.G. Minty, How good is the simplex algorithm?, in: *Inequalities III*, O. Shisha, ed., pp. 159–175, Academic Press, New York, 1972.
37. P. Kleinschmidt and S. Onn, On the diameter of convex polytopes, *Discrete Mathematics*, 102:75–77, 1992.
38. D.J. Naddef, The Hirsch conjecture is true for $(0, 1)$ -polytopes, *Mathematical Programming*, 45:109–111, 1989.
39. S. Onn, Geometry, complexity, and combinatorics of permutation polytopes, *Journal of Combinatorial Theory, Series A*, 64:31–49, 1993.
40. S. Onn, Convex matroid optimization, *SIAM Journal on Discrete Mathematics*, 17:249–253, 2003.
41. S. Onn, U.G. Rothblum, and Y. Tangir, Edge-directions of polyhedra with applications to network flows, forthcoming.
42. S. Onn and L.J. Schulman, The vector partition problem for convex objective functions, *Mathematics of Operations Research*, 26:583–590, 2001.
43. P.M. Pardalos, F. Rendl, and H. Wolkowicz, The quadratic assignment problem: a survey and recent developments, in: *Quadratic Assignment and Related Problems*, pp. 1–42, DIMACS Series on Discrete Mathematics and Theoretical Computer Science, vol. 16, American Mathematical Society, Providence, RI, 1994.
44. R.T. Rockafellar, *Network Flows and Monotropic Optimization*, Wiley-Interscience, New York, 1984.
45. A.S. Schulz and R. Weismantel, The complexity of generic primal algorithms for solving general integer programs, *Mathematics of Operations Research*, 27:681–692, 2002.
46. A.S. Schulz, R. Weismantel, and G. Ziegler, $(0, 1)$ -integer programming: optimization and augmentation are equivalent, in: *Proceedings of the Third Annual European Symposium on Algorithms*, pp. 473–483, Lecture Notes in Computer Science, vol. 979, Springer, Berlin, 1995.
47. B. Sturmfels and R.R. Thomas, Variation of cost functions in integer programming, *Mathematical Programming*, 77:357–388, 1997.
48. T. Zaslavsky, *Facing Up to Arrangements: Face Count Formulas for Partitions of Space by Hyperplanes*, *Memoirs of the American Mathematical Society*, vol. 1, no. 154, American Mathematical Society, Providence, RI, 1975.

Received August 4, 2003, and in revised form January 26, 2004. Online publication August 19, 2004.