Abstract

Colorful linear programming (CLP) is a generalization of linear programming that was introduced by Bárány and Onn. Given \( k \) point sets \( C_1, \ldots, C_k \subset \mathbb{R}^d \) that each contain a point \( b \in \mathbb{R}^d \) in their positive span, the problem is to compute a set \( C \subseteq C_1 \cup \cdots \cup C_k \) that contains at most one point from each set \( C_i \) and that also contains \( b \) in its positive span, or to state that no such set exists. CLP is known to be NP-hard.

We consider a generalization of CLP in which we are given additionally for each set \( C_i \) a number \( l_i \in \mathbb{N} \), \( i = 1, \ldots, k \), and we want to find a set that contains at most \( l_i \) points from \( C_i \). We call this problem \emph{generalized colorful linear programming} (GCLP). While we show that even seemingly simple cases of GCLP remain NP-hard, we present a weakly-polynomial algorithm for the special case that there are only two colors and that the vectors of each set \( C_i \) contain \( b \) in their positive span. This case is particularly interesting due to its connection with the colorful Carathéodory theorem. Furthermore, we consider additional applications of CLP to problems on colored graphs.

1 Introduction

The colorful Carathéodory theorem [2] states that given \( C_1, \ldots, C_{d+1} \subset \mathbb{R}^d \) point sets that all contain the origin in their convex hulls, there always exists a set \( C \subset C_1 \cup \cdots \cup C_{d+1} \) that contains at most one point from each set \( C_i \), \( i = 1, \ldots, d+1 \), and that also contains the origin in its convex hull. We call the sets \( C_i \), \( i = 1, \ldots, d+1 \), \emph{color classes} and we call a set with at most one point from each color class a \emph{colorful choice}. Bárány also gave the following more general version.

**Theorem 1** ([2]) \textit{Let} \( C_1, \ldots, C_d \subset \mathbb{R}^d \) \textit{be point sets and} \( b \in \mathbb{R}^d \) \textit{a point with} \( b \in \text{pos}(C_i) \), \textit{for} \( i = 1, \ldots, d \). \textit{Then}, \textit{there is a colorful choice} \( C \) \textit{with} \( b \in \text{pos}(C) \).

Here, we denote with \( \text{pos}(P) = \{ \sum_{p_i \in P} \alpha_i p_i \mid \alpha_i \geq 0 \text{ for all } p_i \in P \} \) for a set \( P \subset \mathbb{R}^d \) the set of all nonnegative linear combinations of points in \( P \). Using a simple lifting argument, it can be shown that Theorem 1 implies the classic (convex) version of the colorful Carathéodory theorem as stated in the beginning.

In the spirit of the colorful Carathéodory theorem, Bárány and Onn [3] generalized linear programming to the colorful setting: given a point \( b \in \mathbb{R}^d \) and point sets \( C_1, \ldots, C_k \subset \mathbb{R}^d \), we want to find a colorful choice \( C \) with \( b \in \text{pos}(C) \) or state that there is none. We call this problem \emph{colorful linear programming} (CLP) and we call the decision problem to decide whether there exists such a colorful choice DCLP. Bárány and Onn [3] showed that DCLP is NP-complete even if \( k = d \) and each \( C_i \) contains \( 0 \) in its convex hull. This was extended by Mulzer and Stein [8] who showed that DCLP is NP-complete even if \( k = d+1 \) and each \( C_i \) does not necessarily contain \( 0 \) in its convex hull, and by Meunier and Sarrabezolles [7] who showed that DCLP is NP-complete for all values of \( k \) if each \( C_i \) does not necessarily contain \( 0 \) in its convex hull. We define the following generalization of CLP (GCLP): given a point \( b \in \mathbb{R}^d \), point sets \( C_1, \ldots, C_k \subset \mathbb{R}^d \), and numbers \( l_1, \ldots, l_k \in \mathbb{N} \), we want to find a set \( C \) such that \( |C \cap C_i| \leq l_i \) for \( i \in [k] \) and such that \( b \in \text{pos}(C) \) or state that there is none. We obtain CLP by setting \( l_1 = \cdots = l_k = 1 \).

Since CLP is NP-hard, GCLP is NP-hard as well. However, as with regular linear programming and integer programming, GCLP is very versatile and can be used to model colorful versions of many combinatorial problems. Therefore, it is of interest to identify special cases of GCLP that can be solved in polynomial time or to show that even the more restricted version of the problem remains NP-hard. We consider several such examples and delineate a more precise boundary between easy and hard colorful problems.

2 Generalized Colorful Linear Programming

In CLP, we want to find a set that contains at most one point from each color class. In \emph{generalized colorful linear programming} (GCLP) we allow additionally to be given \( k \) nonnegative integers \( l_1, \ldots, l_k \) that determine the number of points that we are allowed to take from each color class. We call a set \( C \) with \( |C \cap C_i| \leq l_i \) for \( i \in [k] \) an \((l_1, \ldots, l_k)-\text{colorful choice}\) or (with a slight abuse of notation) just a \emph{colorful choice}. This is an extended abstract of a presentation given at EuroCG 2016. It has been made public for the benefit of the community and should be considered a preprint rather than a formally reviewed paper. Thus, this work is expected to appear in a conference with formal proceedings and/or in a journal.
2.1 Complexity

Since GCLP is a generalization of CLP, it remains NP-hard. However, even seemingly simple special cases such as \( k = 1 \land l_1 = d \) [3, 6] or \( k = 2 \land l_1 = l_2 = d/2 \) [3] have been shown to be NP-hard as well. We show that for all \( \lambda \), the problem remains NP-hard. We prove this for the convex version of GCLP. That is, we want to find a colorful choice \( C \) that contains \( b \) in its convex hull instead of just in its positive span. Without loss of generality, we can assume \( b = 0 \). By a lifting argument, it can be easily shown that the convex version of GCLP is a special case of GCLP as stated in the introduction. Hence, any hardness results for the convex version hold for the main tool in the reduction. The theorem was first obtained by Knauer et al. [6], albeit with a different proof. We compare both proofs below.

**Theorem 2** Let \( P \subset \mathbb{R}^d \) be a set of size \( 2d \). It is NP-complete to decide whether there is a subset \( P' \subset P \) of size \( d \) containing the origin in its convex hull.

**Proof.** Let \( A = \{a_1, \ldots, a_d\} \) be an instance of PARTITION, for \( d \) even. For \( i \in \{1, \ldots, d-1\} \), we define the vector \( v_i \in \mathbb{R}^d \) as having its \( i \)-th coordinate equal to \( 1 \), its last coordinate equal to \( a_i \), and all other coordinates equal to 0. The vector \( v_d \) has all its coordinates equal to \(-1\) except for the last coordinate, which is equal to \( a_d \). Similarly, we define vectors \( w_i \in \mathbb{R}^d \), and just replace the last coordinate by \( -a_i \). Assume there is a partition \( A_1, A_2 \) of \( A \) with \( \sum_{a \in A_j} a = \sum_{a \in A_j} a \). Then, we have \( \sum_{a \in A_1} v_i = -\sum_{a \in A_2} w_i \) and hence \( 0 \in \text{conv}(\{v_1 | a_i \in A_1\} \cup \{w_1 | a_i \in A_2\}) \). On the other hand, let \( V' = \{v_1, \ldots, v_d\} \) and \( W' = \{w_1, \ldots, w_d\} \) be s.t. |\( V' \)\| + |\( W' \)\| = \( d \) and s.t. \( \mathbf{0} = \sum_{v \in V'} \lambda_v v + \sum_{w \in W'} \lambda_w w \), where \( \sum_{v \in V'} \lambda_v = \sum_{w \in W'} \lambda_w = 1 \) and \( \lambda_v, \lambda_w \geq 0 \) for all \( v \in V', w \in W' \). By construction, for all \( i = 1, \ldots, d \), we have either \( v_i \in V' \) or \( w_i \in W' \) and furthermore, all coefficients \( \lambda_v, \lambda_w \), \( v \in V', w \in W' \), are equal. Hence, the sets \( A_1 = \{a_i | v_i \in V'\}, A_2 = \{a_i | w_i \in W'\} \) form a partition of \( A \) with \( \sum_{a \in A_1} a = \sum_{a \in A_2} a \).

We note that the set \( P \) constructed in the proof of Theorem 2 was first described by Bárány and Omm [3, Theorem 5.1]. However, they used it to prove the weaker statement DCLP is NP-complete even for \( k = d \). This result is a consequence from Theorem 2 by setting \( C_1 = \cdots = C_d = P \). Also, NP-hardness of the two special cases \( k = 1 \land l_1 = d \) and \( k = 2 \land l_1 = l_2 = d/2 \) follows from Theorem 2 by setting \( C_1 = P \land l_1 = d \) and \( C_1 = C_2 = P \land l_1 = l_2 = d/2 \), respectively.

Note further that the problem from Theorem 2 was first shown to be NP-complete by Knauer et al. [6]. Additionally, the proof of Theorem 2 gives an alternative proof for the \#P-completeness of computing the simplicial depth. This hardness result was first obtained by Afshani et al. [1] and the alternative reduction is analogous to the proof of [1, Theorem 9]. It is not immediate that the reduction from Knauer et al. [6] has similar implications.

In the following, let \( GCLP_k(r_1, \ldots, r_k), r_i \in (0, 1) \), denote GCLP restricted to instances with exactly \( k \) color classes and the \( l_i \)'s are given by \( l_i = |r_i|^2 r_i \) for \( i \in [k] \), where \( r_i = |C_i| \). That is, we are allowed to take a constant fraction of each color class.

**Theorem 3** For any fixed \( k \in \mathbb{N} \) and any fixed ratios \( r_1, \ldots, r_k \in (0, 1) \), \( GCLP_k(r_1, \ldots, r_k) \) is NP-hard.

**Proof.** We prove the statement by a reduction similar to the proof of Theorem 2. Given some partition instance \( A = \{a_1, \ldots, a_d\} \), let \( P \subset \mathbb{R}^d \), denote the same point set as in the proof of Theorem 2. If \( r_1|P| = d \), we set \( C_1 = P \) and create “dummy” points for \( C_2, \ldots, C_k \) that will never be part of a convex combination of \( 0 \). To ensure this, we lift \( P \) to \( \mathbb{R}^{d+1} \) by appending a 0-coordinate. Now, we set \( C_i = \{c_i\} \) for \( i = 2, \ldots, k \), where the coordinates of \( c_i \in \mathbb{R}^{d+1} \) are 0 in dimensions \( j = 1, \ldots, d \) and some positive number in dimension \( d + 1 \). Now, assume \( r_1|P| < d \) and hence \( |P| < d/r_1 \). We add \( |d/r_1 - |P|| \) dummy points together with \( P \) to \( C_1 \) and create the other color classes as before. Then, we have \( |r_1|C_1| = d \) as desired.

The last case is \( |r_1|P| > d \). Again, we set \( C_1 = P \) and construct \( C_2, \ldots, C_k \) as above. To ensure that we only take \( d \) points from \( P \), we add “mandatory” points to \( C_1 \) that have to be part of any convex combination of \( 0 \). We construct a mandatory point \( q \) by introducing a new dimension in which we set the coordinates of all other points to 1. The new point \( q \) has coordinates set to 0 in all but the new dimension, where it is set to \(-1\). A short calculation reveals that we have to add \( m = \left\lceil \frac{|r_1|P| - d}{d - r_1} \right\rceil \) mandatory points together with \( P \) to \( C_1 \) in order to ensure that \( |r_1|C_1| = d + m \).

Thus, the existence of a \( (r_1|C_1|, \ldots, r_k|C_k|) \)-colorful choice is equivalent to the existence of a partition of \( A \) into two sets \( A_1, A_2 \) with \( \sum_{a \in A_1} a = \sum_{a \in A_2} a \). Since \( r_1 \) is constant, we can create the additional dummy/mandatory points in polynomial time.

2.2 A Special Case

We now consider the following special case of GCLP: given a point \( b \in \mathbb{R}^d \), a ratio \( r \in [0, 1] \), and point sets \( C_1, C_2 \subset \mathbb{R}^d \) of size \( d \) with \( b \in \text{pos}(C_i) \) for \( i = 1, 2 \), we want to find an \( \{\lfloor rd \rfloor, \lceil rd \rceil\} \)-colorful choice \( C \) with \( b \in \text{pos}(C) \), or state that there is none.

Theorem 1 guarantees the existence of such a colorful choice: we set the first \( \lfloor rd \rfloor \) color classes to copies
of \(C_1\), and the next \([rd]\) color classes to copies of \(C_2\). Hence, this simple case of only two colors is particularly interesting as we know that there always exists a solution, but computing it is already nontrivial. Note that for \(l_1 = [rd] - 1\) or \(l_2 = [rd] - 1\) the problem becomes NP-hard as a consequence of Theorem 2.

We give a weakly-polynomial algorithm for the two-color case that is based on constructing a family of linear programs. Let \(L\) denote the linear system \(Ax = b, x \geq 0\), where \(A \in \mathbb{R}^{d \times 2d}\) contains \(C_1\) as its first \(d\) columns and \(C_2\) as its second \(d\) columns. In the following, we assume that \(L\) is in general position. Given a cost vector \(c \in \mathbb{R}^d\), we denote with \(L_c\) the linear program that maximizes the objective function \(c^T x\) subject to the equalities and inequalities from \(L\).

Let \(c_1\) and \(c_2\) be two generic cost vectors such that \(C_1\) and \(C_2\) are optimal bases. One can show that \(c_1\) and \(c_2\) can be obtained in polynomial time. For \(\lambda \in [0, 1]\), we denote with \(c_\lambda\) the cost vector \(\lambda c_1 + (1 - \lambda)c_2\) and with \(L_\lambda\) the linear program \(L_{c_\lambda}\). That is, the linear programs \(L_\lambda\), \(\lambda \in [0, 1]\), differ only in their cost functions which are convex combinations of \(c_1\) and \(c_2\). Our construction has the following properties.

**Lemma 4** There is a finite number of ordered intervals \(I_1, \ldots, I_s\) with pairwise disjoint interiors such that \(\bigcup_{i=1}^s I_i = [0, 1]\) and such that

(i) The length of each interval \(I_i, i \in [s]\), is at least \(1/K\), where \(K \in \mathbb{N}\) and \(\log K\) is bounded by a polynomial in the description size of \(L\).

(ii) For each \(i \in \{1, \ldots, s\}\), there is a unique feasible basis that is optimal for all \(L_\lambda\), where \(\lambda\) is contained in the interior of \(I_i\).

(iii) For \(\lambda\) belonging to two distinct intervals \(I_i, I_{i+1}\), there are exactly two optimal bases that differ exactly by one column.

**Proof.** (i): This follows from standard tools such as Cramer’s rule and the Leibniz formula for determinants. (ii) & (iii): Let \(\lambda \in [0, 1]\) and let \(B\) be an optimal basis for \(L_\lambda\). We denote with \(N\) the set of columns from \(A\) not in \(B\). Then, the reduced cost vector [5] is given by \(r_{B, \lambda} = (c_\lambda)_N - A_N^T (A_B^{-1})^T\), where \((c_\lambda)_N\) denotes the subvector of \(c_\lambda\) restricted to the coordinates corresponding to columns in \(N\), \(A_N\) denotes the submatrix of \(A\) with columns in \(N\), and \(A_B\) denotes the submatrix of \(A\) with columns in \(B\). If the sign of the \(i\)th coordinate of \(r_{B, \lambda}\) is positive, then swapping the corresponding column from \(N\) into \(B\) increases the cost and otherwise (if the sign is non-positive), the cost remains equal or decreases. Since we want to maximize the objective function, a basis is optimal iff all coordinates of \(r_{B, \lambda}\) are non-positive, and it is unique if all coordinates of \(r_{B, \lambda}\) are negative.

We obtain the intervals \(I_1, \ldots, I_s\) iteratively as follows: initially we set \(\lambda = 0\). By general position and genericity of \(c_1\), the unique optimal basis for \(L_\lambda\) is \(C_1\), i.e., all coordinates of \(r_{B, \lambda}\) are negative. Now, we continuously increase \(\lambda\) until one of the coordinates of \(r_{B, \lambda}\) becomes 0. Let \(x_1\) denote this value and let \(i\) be the coordinate of \(r_{B, \lambda}\), that is 0 (by general position and genericity, \(i\) is unique). Since \(C_1\) is not an optimal basis for \(L_1\), \(\lambda_1\) exists. Because each coordinate of \(r_{B, \lambda}\) is a linear function in \(\lambda\), \(r_{B, \lambda}^i\), is positive for all \(\lambda > \lambda_1\). Then, there exists an \(\epsilon > 0\) such that \(i\) is the only nonnegative coordinate of \(r_{B, \lambda}\) for \(\lambda > \lambda_1 + \epsilon\). Hence, for all \(\lambda > \lambda_1\), the basis \(B'\) that is obtained by swapping the column from \(N\) that corresponds to coordinate \(i\) of \(r_{B, \lambda}\) into \(B\) is the unique optimal basis. Note further, that both \(B'\) and \(B\) are optimal for \(L_{\lambda_1}\). Set \(I_1 = [0, \lambda_1]\) and construct iteratively the next intervals until \(B' = C_2\). Let \(\lambda_s \in (0, 1]\) be the minimum value for which \(C_2\) is an optimal basis for \(L_{\lambda_s}\). Then, \(C_2\) is optimal for every \(\lambda > \lambda_s\). We set \(I_s = [\lambda_s, 1]\) and conclude the construction of the intervals.

We now describe the complete algorithm. In round \(i\), we maintain an interval \([a_i, b_i] \subset [0, 1]\), such that the optimal basis for \(L_{a_i}\) contains at least \([rd]\) columns from \(C_1\) (and due to the general position assumption, at most \([1-rd]\) columns from \(C_2\)) and such that the optimal basis for \(L_{b_i}\) contains at least \([rd]\) columns from \(C_1\). We maintain the following invariant: there exists a \(\lambda \in [a_i, b_i]\) such that the optimal basis for \(L_\lambda\) is the desired \((\lceil rd\rceil, \lfloor rd\rfloor)\)-colorful choice.

Initially, we set \([a_1, b_1] = [0, 1]\). By definition, \(C_1\) is the optimal basis for \(L_0\) and \(C_2\) is the optimal basis for \(L_{1}\). By Lemma 4(iii) optimal bases for two adjacent intervals differ only in one column, and hence the invariant holds for \([a_1, b_1]\). We solve then the linear program \(L_\lambda\) for \(\lambda = (a_k + b_k)/2\) and let \(B^*\) denote the optimal basis. If \(B^*\) contains at least \([rd]\) columns from \(C_1\), we set \(a_{k+1}\) to \(\lambda\) and \(b_{k+1} = b_k\). Otherwise, we set \(a_{k+1} = a_k\) and \(b_{k+1} = \lambda\). Let \(B_k\) be the optimal basis for \(L_{a_{k+1}}\), and let \(B_{k+1}\) be the optimal basis for \(L_{b_{k+1}}\). Since \(B_1\) contains at least \([rd]\) columns of \(C_1\) and since \(B_2\) contains at most \([rd]\) columns of \(C_1\), the invariant holds for \([a_{k+1}, b_{k+1}]\) again by Lemma 4(iii).

After \(i^* = O(\log K)\) iterations, the interval \([a_{i^*}, b_{i^*}]\) is contained in the union of the two adjacent intervals \(I_j, I_{j+1}\) with \(j \in [s - 1]\). Let \(B_j\) and \(B_{j+1}\) be the optimal bases for \(I_j\) and \(I_{j+1}\), respectively. Hence, by Lemma 4(ii), \(B_j\) or \(B_{j+1}\) is the desired basis.

Each round requires polynomial time, and the number of rounds is bounded by a polynomial in the bit-size of the input. The following theorem is immediate.

**Theorem 5** Let \(b \in \mathbb{R}^d\) be a vector and let \(C_1, C_2 \subset \mathbb{R}^d\) be two sets of size \(d\) with \(b \in \text{pos}(C_i)\) for \(i = 1, 2\). Furthermore, let \(r \in [0, 1]\) be a parameter. Then, there is an algorithm that computes an \((\lceil rd\rceil, \lfloor rd\rfloor)\)-colorful choice \(C\) from \(b \in \text{pos}(C)\) in weakly-polynomial time.
3 Applications of Colorful Linear Programming

We consider two problems on colored graphs that can be cast as a CLP and analyze their complexity. The first problem is called ColorfulPath: given a directed graph $G = (V,E)$ whose edges are partitioned into $k$ color classes $C_1, \ldots, C_k$ and two vertices $s,t \in V$, the problem is to decide whether there exists a directed path from $s$ to $t$ with at most one edge from each color class. ColorfulPath is a special case of CLP, since the existence of an $s$-$t$ path can be modeled as a flow. Chakraborty et al. [4] showed this problem to be NP-complete by a reduction from 3-SAT. We present a similar but simplified proof, that reduces the number of necessary colors from $O(|V|^2)$ to $O(m)$, where $m$ is the number of clauses and $n$ is the number of variables in the 3-SAT formula.

**Theorem 6** ColorfulPath is NP-complete, even if the graph $G = (V,E)$ is acyclic and $|E| = O(|V|)$.

**Proof.** Consider a 3-SAT formula $\Phi$, with $n$ variables $x_1, \ldots, x_n$ and $m$ clauses $C_1, \ldots, C_m$, each containing exactly three literals. Our directed graph has $3m$ colors $c_{jk}$, $j = 1, \ldots, m$ and $k = 1, 2, 3$, one for each literal in each clause. We allow multiple edges between two vertices. However, our construction can be easily modified to at most one edge per vertex-pair by introducing new vertices. For each clause $C_j$ we have one clause gadget $G_j$ and for each variable $x_i$, we have one variable gadget $G'_i$. The clause gadget $G_j$ for a clause $C_j$ consists of two vertices $\{s_j, t_j\}$ and three directed edges from $s_j$ to $t_j$ with colors $c_{j1}$, $c_{j2}$, and $c_{j3}$. The variable gadget $G'_i$ for a variable $x_i$ consists of two edge-disjoint paths that are vertex disjoint except at the start and the end vertex. The first path contains one edge for each positive occurrence of $x_i$ in $\Phi$, colored with the color that corresponds to this literal. The second path contains one analogous edge for each negative occurrence of $x_i$ in $\Phi$. The graph $G$ is obtained by concatenating all clause gadgets and all vertex gadgets and by identifying the last vertex in each gadget with the first vertex in the following gadget. This construction can be performed in polynomial time, and there is a colorful path through all gadgets if and only if $\Phi$ is satisfiable. \hfill \Box

We conclude with AnotherColorfulCycle (ACC): given a graph $G = (V,E)$, where $|E| = 2|V|$ and all edges are colored with $n = |V|$ colors such that exactly two edges have the same color, and a colorful Hamilton cycle in $G$, we want to find another colorful cycle (not necessarily Hamiltonian). This is a special case of the PPAD-complete problem AnotherColorfulSimplex [7] (ACS) and related to the PPA-problem AnotherHamiltonPath [9] (AHP), in which we are given a graph $G$ and a Hamilton path in $G$, and we want to find another Hamilton path in $G$ or in its complement. While there are no polynomial-time algorithms known for ACS and AHP, we show that ACC can be solved efficiently.

**Theorem 7** AnotherColorfulCycle can be solved in polynomial time.

**Proof.** Consider the bipartite graph $G' = (V', E')$ with vertices $V' = V \cup \{C_1, \ldots, C_m\}$. There is an edge $(v, C_i) \in E'$ if there is an outgoing edge from a vertex $v \in V$ with color $C_i$ in $G$. Note that there is a bijection between $E'$ and $E$. Furthermore, the edges $M \subset E'$ in $G'$ that correspond to the edges of the Hamiltonian cycle in $G$ are a perfect matching in $G'$. Since $|E| \geq |V|$, there is a cycle $C$ in $G'$. As each vertex $C_i \in V'$, $i \in [n]$, is incident to two edges, and since one of them is contained in $M$, $C$ is of even length and its edges alternate between $M$ and $E \setminus M$. Then, $M' = M \cup C$ is a perfect matching different from $M$. It induces a colorful set of edges where each vertex $v \in V$ has exactly one outgoing edge in $M'$. Hence, $M'$ corresponds to a colorful cycle in $G$. \hfill \Box

**References**


