Connectedness of Efficient Solutions in Multiple Criteria Combinatorial Optimization

Matthias Ehrgott* Universität Kaiserslautern

Kathrin Klamroth* Universität Kaiserslautern

Abstract

In multiple criteria optimization an important research topic is the topological structure of the set X_e of efficient solutions. Of major interest is the connectedness of X_e, since it would allow the determination of X_e without considering non-efficient solutions in the process. We review general results on the subject, including the connectedness result for efficient solutions in multiple criteria linear programming. This result can be used to derive a definition of connectedness for discrete optimization problems. We present a counterexample to a previously stated result in this area, namely that the set of efficient solutions of the shortest path problem is connected. We will also show that connectedness does not hold for another important problem in discrete multiple criteria optimization: the spanning tree problem.

1 Introduction and General Results

In this paper we consider multiple criteria optimization problems of the form

$$\min \ f(x)$$
$$\text{s.t.} \quad x \in X$$

where $f : \mathbb{R}^n \to \mathbb{R}^Q$ and $X \subseteq \mathbb{R}^n$. Although we assume familiarity with the basic concepts of multicriteria optimization we will briefly give the most important definitions.

In the general case $\mathbb{R}^Q$ is ordered by a cone K (see [15] for general results on orders defined by cones). In multiple criteria optimization the notion of optimality is usually replaced by efficiency, since in general different solution values in $\mathbb{R}^Q$ exist which can be considered as "best" solutions of the problem in the sense that they cannot be improved. $x_e \in X$ is called efficient solution if $(\{f(x_e)\} \Leftrightarrow K) \cap f(X) = \{f(x_e)\}$. Note that for $Y, K \subset \mathbb{R}^Q$ the set $Y \Leftrightarrow K$ is defined to be $\{y \Leftrightarrow k \mid y \in Y, k \in K\}$. The set of all efficient solutions is denoted by X_e. Most of the research in this area has been devoted to the case where $K = \mathbb{R}^Q_+$. Then the ordering defined by K is the componentwise order and $x_e \in X$ is efficient if there is no $x \in X$ dominating x_e, i.e. there is no $x \in X$ such that $f_q(x) \le f_q(x_e), q = 1, \ldots, Q$ and strict inequality holds in at least one case.

The topic of this paper will be connectedness of X_e. Although of great importance for the construction of algorithms there is not much work done in that field. For general "continuous" problems we are dealing with the question: Is X_e connected in the topological sense? In order to state a general result providing the answer to this question, we define a set $Y \subset \mathbb{R}^Q$ to be $K\Leftrightarrow$compact if $(\{y\} \Leftrightarrow K) \cap Y$ is compact for all $y \in Y$.

Theorem 1 ([13]) *If K is a closed, convex, pointed (i.e. $x \in K \Rightarrow \Leftrightarrow x \notin K$) cone such that $int(K) \neq \emptyset$ and $Y = f(X)$ is closed, convex and $K\Leftrightarrow$compact, then X_e is connected.*

*Partially supported by grants of the Deutsche Forschungsgemeinschaft and grant ERBCHRXCT930087 of the European HC&M Programme

Algorithmically this result can be applied to determine the set X_e by local search methods, i.e. given any efficient solution $x \in X_e$ we can search locally to find efficient solutions in the neighbourhood of x until eventually the whole set X_e is determined.
The result of Theorem 1 has been generalized in [8] to the case where $\mathbb{R}^n$ and $\mathbb{R}^Q$ are replaced by locally convex spaces. Several authors proved connectedness of X_e for special types of functions [14, 2]. Also several results on the connectedness of the set of weakly efficient solutions are known [14], where $x_{we} \in X$ is said to be weakly efficient if $(f(x_{we}) \Leftrightarrow int(K)) \cap f(X) = \emptyset$. In the following we will only consider the case $K = \mathbb{R}^Q_+$. In this case efficient solutions are often called pareto optimal solutions.
For combinatorial optimization problems, however, this result is not applicable. In this case we will use the concept of connectedness of a graph rather than the topological connectedness of Theorem 1. We define a graph, the nodes of which represent the efficient solutions of the combinatorial problem. Edges are introduced between all pairs of nodes which are "adjacent" in some sense. If the resulting graph is connected this fact will allow the development of algorithms to find X_e by search among adjacent solutions.
The most promising definition of adjacency seems to be by using the link between these two concepts of connectedness, which is provided by linear programming. A multiple criteria linear program (MCLP) is defined as

$$\min(c^1 x, \ldots, c^Q x) \text{ where } c^q \in \mathbb{R}^n, q = 1, \ldots, Q$$

$$\text{s.t. } x \in X = \left\{x \in \mathbb{R}^n \mid x \geq 0, a^k x \leq b_k, k = 1, \ldots, m\right\}$$

Obviously Theorem 1 immediately implies that X_e is connected in this special case. Before the general result of Theorem 1 was known the connectedness result for MCLP had been proved by various authors [3, 5, 16].
The most important solutions in linear programming are basic solutions which correspond to extreme points of the polyhedral feasible set X, and fundamental solutions which correspond to extreme rays of X, if X is unbounded. Let B and F denote the sets of basic feasible and fundamental solutions, respectively. Then $x_e^1, x_e^2 \in X_e \cap B$ are said to be adjacent if they have $m \Leftrightarrow 1$ basic variables in common and $\alpha x_e^1 + (1 \Leftrightarrow \alpha) x_e^2$ is efficient for all $\alpha \in [0, 1]$. Furthermore $x_e \in B \cap X$ and $x_f \in F \cap X$ are said to be adjacent if $x_e + \beta x_f$ is efficient for all $\beta \geq 0$. Now let $\mathcal{B}$ be the index set of all efficient basic feasible solutions and let $\mathcal{F}$ be the index set of all fundamental solutions which are adjacent to an efficient basic feasible solution. The main result in [9] is the following:

Theorem 2 ([9]) *Define a graph $G = (V, E)$ by introducing a node for each index in $\mathcal{B} \cup \mathcal{F}$ and an edge between two nodes if the corresponding solutions are adjacent. Then G is connected.*

Theorem 2 is used in MCLP simplex-algorithms: The set of all efficient extreme points of X is determined by pivoting among efficient bases only, i.e. by moving from efficient extreme point to adjacent efficient extreme point.
Thus in this paper we will restrict ourselves to combinatorial optimization problems which have a linear programming formulation, s.t. feasible solutions of the combinatorial problem correspond to basic feasible solutions of the associated linear program. Two feasible solutions of the combinatorial problem are then called adjacent if the corresponding basic feasible solutions of the associated LP are adjacent.

2 Combinatorial Problems

2.1 The Shortest Path Problem

In [12] a multiple criteria dynamic programming algorithm for the shortest path problem is derived. This algorithm allows finding all efficient paths from node s to node t in a given directed graph

$G = (V, A)$. The multiple criteria shortest path problem can be formulated as a linear program by

$$\min(c^1 x, \ldots, c^Q x)$$

$$\text{s.t.} \quad \sum_j x_{ij} \Leftrightarrow \sum_j x_{ji} = \begin{cases} 1 & i = s \\ 0 & i \notin \{s, t\} \\ \Leftrightarrow 1 & i = t \end{cases}$$

According to the definitions in Section 1 two paths from s to t are adjacent if they correspond to two adjacent basic feasible solutions of the above LP. These basic feasible solutions represent spanning trees of the underlying directed graph G. From the algorithm and the connectedness result for MCLP the author concludes that the following result holds:

Theorem 3 ([12]) *Let p, p' be two efficient s⇔t⇔paths. Then there exists a sequence of adjacent efficient s⇔t⇔paths $(p, p_1, \ldots, p_k, p')$.*

In the sense of Theorem 2 this means that the graph defined by adjacency of efficient s⇔t⇔paths is connected. But this conclusion is not true in general, see Theorem 4.

2.2 The Spanning Tree Problem

Another important discrete optimization problem is the spanning tree problem: Given an undirected graph $G = (V, E)$, find $\min(c^1(T), \ldots, c^Q(T))$ such that T is a spanning tree of G. The linear programming formulation of this problem is:

$$\min(c^1 x, \ldots, c^Q x)$$

$$\begin{aligned} \text{s.t.} \sum_{e \in E} x_e &= n \Leftrightarrow 1 \\ \sum_{e \in E(S)} x_e &\leq |S| \Leftrightarrow 1 \qquad \forall S \subset V \\ &\ \text{where } E(S) = \{e = [i, j] \in E \mid i, j \in S\} \\ x_e &\geq 0 \end{aligned}$$

Again according to Section 1 two (efficient) spanning trees are adjacent if they have $n \Leftrightarrow 2$ edges in common. We will now formally introduce the efficiency graph corresponding to a spanning tree problem and a shortest path problem, which has been introduced as Pareto graph in [4].

Definition 1 *Let $G = (V, E)$ with edge costs $c^1, \ldots, c^Q : E \to I\!\!R$ be a given graph. The efficiency graph $\mathcal{EG}^T(G)$ for the spanning tree problem on G is defined as follows: Its node set is the set of efficient spanning trees of G. Two nodes are joined by an edge if the corresponding spanning trees are adjacent. Analogously the efficiency graph $\mathcal{EG}^{P(s,t)}(G)$ for the shortest path problem on G with end nodes s and t is defined: Its node set is the set of efficient paths from s to t. Two nodes are joined by an edge if the corresponding paths are adjacent, where adjacency is defined as in Section 2.1.*

Connectivity Conjecture: $\mathcal{EG}^T(G)$ and $\mathcal{EG}^{P(s,t)}(G)$ are connected.

The connectivity conjecture has been stated in [4] and [6] for the general matroid optimization problem and the matroid intersection problem, respectively. Also Theorem 3 is a reformulation of the connectivity conjecture for the shortest path problem. The important implication is that if the conjecture were true it would be possible to find all efficient solutions of the spanning tree and shortest path problem by neighbourhood search, i.e. by exchanges of one edge in the trees which correspond to efficient basic feasible solutions of the linear programming formulations of the two problems. In particular the approximation algorithms stated in [7] or [11] would find all efficient spanning trees and thus be exact.

2.3 A Counterexample to the Connectivity Conjecture

Theorem 4 *$\mathcal{EG}^T(G)$ and $\mathcal{EG}^{P(s,t)}(G)$ are not connected in general.*

The proof is provided by Example 1 and Lemma 1.

Example 1 *Consider the graph $G_1 = (V, E)$ given in Figure 1. There are 12 efficient spanning trees of G_1, listed in Table 1. Obviously each efficient spanning tree contains all edges with cost $(0,0)$. Therefore in Table 1 we only list edges with positive costs.*

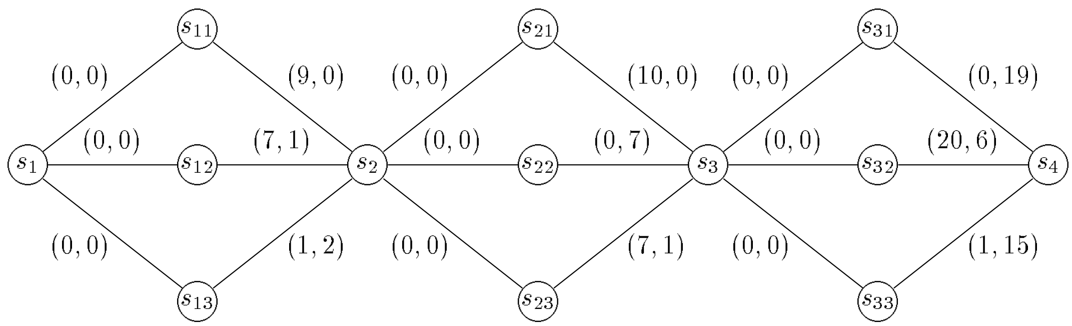

Figure 1: Graph G_1 has nonadjacent efficient spanning trees

Efficient Tree	Edges	Cost
T_1	$[s_{13}, s_2][s_{22}, s_3][s_{31}, s_4]$	(1,28)
T_2	$[s_{13}, s_2][s_{22}, s_3][s_{33}, s_4]$	(2,24)
T_3	$[s_{13}, s_2][s_{23}, s_3][s_{31}, s_4]$	(8,22)
T_4	$[s_{13}, s_2][s_{23}, s_3][s_{33}, s_4]$	(9,18)
T_5	$[s_{13}, s_2][s_{21}, s_3][s_{33}, s_4]$	(12,17)
T_6	$[s_{11}, s_2][s_{23}, s_3][s_{33}, s_4]$	(17,16)
T_7	$[s_{11}, s_2][s_{21}, s_3][s_{33}, s_4]$	(20,15)
T_8	$[s_{12}, s_2][s_{22}, s_3][s_{31}, s_4]$	(27,14)
T_9	$[s_{13}, s_2][s_{23}, s_3][s_{31}, s_4]$	(28,9)
T_{10}	$[s_{13}, s_2][s_{21}, s_3][s_{31}, s_4]$	(31,8)
T_{11}	$[s_{11}, s_2][s_{23}, s_3][s_{31}, s_4]$	(36,7)
T_{12}	$[s_{11}, s_2][s_{21}, s_3][s_{31}, s_4]$	(39,6)

Table 1: Efficient spanning trees of G_1

It is easy to see that T_8 is not adjacent to any other efficient spanning tree.

We will now look at the problem of finding efficient paths from s_1 to s_4 in the same graph of Figure 1. Clearly if $\mathcal{EG}^{P(s_1,s_4)}(G_1)$ is not connected, the same holds in the directed case: G_1 can be made directed by just orienting each arc from left to right in Figure 1. Lemma 1 then provides the counterexample to Theorem 2.

Lemma 1 *In Example 1, $\mathcal{EG}^{P(s_1,s_4)}(G)$ and $\mathcal{EG}^T(G)$ are isomorphic.*

Proof:

Every tree $T_i \in V(\mathcal{EG}^T(G))$ must contain exactly one of the edges $[s_{j1}, s_{j+1}]$, $[s_{j2}, s_{j+1}]$, $[s_{j3}, s_{j+1}]$ for each $j = 1, 2, 3$, respectively. Analogously every path $P_i \in V(\mathcal{EG}^{P(s_1,s_4)}(G))$ contains exactly one of these three edges for each $j = 1, 2, 3$. Now let $f : V(\mathcal{EG}^T(G)) \to V(\mathcal{EG}^{P(s_1,s_4)}(G))$ be defined by $f(T_i) = P_l$ if and only if $\forall j, k = 1, 2, 3$ $[s_{jk}, s_{j+1}] \in T_i \Rightarrow [s_{jk}, s_{j+1}] \in P_l$. It is easy to see that f is bijective. Thus it remains to check whether T_i is adjacent to T_j in $\mathcal{EG}^T(G)$ if and only if $f(T_i)$ is adjacent to $f(T_j)$ in $\mathcal{EG}^{P(s_1,s_4)}(G)$. If T_i is adjacent to T_j in $\mathcal{EG}^T(G)$ by the definition of adjacency of paths it follows immediately that $f(T_i)$ is adjacent to $f(T_j)$. On the other hand, if T_i is not adjacent to T_j then $f(T_i)$ and $f(T_j)$ differ in at least two of the three subpaths $(s_1, s_2), (s_2, s_3)$ and (s_3, s_4). Thus there can't exist two spanning trees $\tilde{T}_1$ and $\tilde{T}_2$ with $f(T_1) \subset \tilde{T}_1$, $f(T_2) \subset \tilde{T}_2$ and $\tilde{T}_1, \tilde{T}_2$ being adjacent, i.e. having 11 edges in common.

□

2.4 Generalization

A disconnected efficiency graph may also occur in more general situations and not only in the example presented above. We will show that it is possible to extend any graph in such a way that the efficiency graph for the problem on the extended graph is not connected. This holds for both the spanning tree and the shortest path problem.

Let $G = (V, E)$ be an arbitrary given (connected) graph. We construct a graph $\tilde{G}$ containing G as a subgraph such that $\mathcal{EG}^T(\tilde{G})$ is not connected.

Let $c : E \to I\!R^Q_+$ be the cost-function on the edges of G. For simplicity we restrict ourselves to the case $Q = 2$. Note that the results also hold for $Q > 2$.

First we consider a subset of the efficient spanning trees of G, the so called extremal efficient spanning trees:

Definition 2 *A spanning tree T^* of G is an extremal efficient spanning tree if there exist $0 < \lambda < 1$ such that $T^* \in$ argmin $\{\sum_{e \in E(T)} \lambda c^1(e) + (1 - \lambda) c^2(e) \,|\, T$ is spanning tree of $G\}$.*

Notice that despite Theorem 4 it is known that the set of extremal efficient spanning trees *is connected*, see [4].

Some further definitions are required to facilitate notation in the following .

Definition 3 1. *A spanning tree T of G is a lexicographic minimal spanning tree w.r.t. (c^1, c^2) if there exists no other tree T' such that $(c^1(T'), c^2(T')) <_L (c^1(T), c^2(T))$ where $<_L$ denotes the "lexicographical smaller" relation on $I\!R^2$. The set of all such trees is denoted by $\mathcal{T}_1$. Analogously T is lexicographically minimal w.r.t. (c^2, c^1) if there is no tree T' such that $(c^2(T'), c^1(T')) <_L (c^2(T), c^1(T))$. These trees are denoted by $\mathcal{T}_2$.*

2. *Let T_1, T_2 be spanning trees of G. T_1 dominates T_2 if $c_i(T_1) \le c_i(T_2), i = 1, 2$ with strict inequality in at least one case.*

3. *For a graph G and $v \in V(G)$ let $d_G(v)$ denote the degree of v, i.e. the number of edges incident to v. Let $V^* \subset V$. Then $\delta_{V^*}(G) := \min_{v \in V \setminus V^*} d_G(v)$ denotes the minimal degree of nodes not in V^*.*

We will assume that G has at least one efficient spanning tree which is not contained in $\mathcal{T}_1 \cup \mathcal{T}_2$, and henceforth exclude this case, which is trivial from a multiple objective point of view.

In the following we will construct a graph G' containing G as a subgraph such that, in the corresponding efficiency graph, $\delta_{\mathcal{T}_1 \cup \mathcal{T}_2}(\mathcal{EG}^T(G')) < \delta_{\mathcal{T}_1 \cup \mathcal{T}_2}(\mathcal{EG}^T(G))$. Thus applying the procedure of constructing G' iteratively we will be able to find a graph $\tilde{G}$ containing G such that $\delta_{\mathcal{T}_1 \cup \mathcal{T}_2}(\mathcal{EG}^T(\tilde{G})) = 0$, implying that the efficiency graph of $\tilde{G}$ is disconnected.

Let $T_1, \ldots, T_m$ be the set of efficient spanning trees of G. Furthermore let $T_k \in \{T_1, \ldots, T_m\} \setminus (\mathcal{T}_1 \cup \mathcal{T}_2)$ such that $d_{\mathcal{EG}^T(G)}(T_k) = \delta_{\mathcal{T}_1 \cup \mathcal{T}_2}(\mathcal{EG}^T(G))$. We denote the costs of T_i by (a_i, b_i) and will use $(x, y) = (a_k, b_k)$ for easier distinction. We assume that the numbering of efficient trees

is such that the costs are ordered lexicographically, i.e. $a_1 \leq \ldots \leq x \leq \ldots \leq a_l \leq \ldots a_m$ and $b_1 \geq \ldots \geq y \geq \ldots \geq b_l \geq \ldots \geq b_m$. Furthermore let $T_l \in \{T_{k+1}, \ldots, T_m\}$ such that T_k and T_l are connected by an edge in $\mathcal{EG}^T(G)$.
We distinguish two cases and extend G in two different ways:

Extension 1:
First let us assume that there exist $n \in I\!N$ and $0 < \epsilon < \min\{x \Leftrightarrow a_1, a_l \Leftrightarrow x\}$ such that

$$\begin{aligned} x &> \frac{1}{n} a_l + \frac{n \Leftrightarrow 1}{n}(a_1 + \epsilon) && (1) \\ y &> \frac{1}{n} b_l + \frac{n \Leftrightarrow 1}{n}(b_1 + \epsilon) && (2) \end{aligned}$$

Then $G' = (V(G) \cup \{v'\} \cup \{v_0, \ldots, v_n\}, E(G) \cup \{[v, v_0], \ldots, [v, v_n]\} \cup \{[v_0, v'], \ldots, [v_n, v']\})$ (where v is an arbitrary node of $V(G)$).
Let $C = \Leftrightarrow (n \Leftrightarrow 1)(y \Leftrightarrow b_1 \Leftrightarrow \epsilon)$ and assign the following costs to the additional edges:

$$\begin{aligned} c(v, v_i) &= (0, 0); \qquad i \in \{0, \ldots, n\} \\ c(v_0, v') &= (a_1 \Leftrightarrow x \Leftrightarrow \epsilon, C + b_1 \Leftrightarrow b_l + \epsilon) \\ c(v_i, v') &= ((i \Leftrightarrow 1)(x \Leftrightarrow a_1 \Leftrightarrow \epsilon), C + (i \Leftrightarrow 1)(y \Leftrightarrow b_1 \Leftrightarrow \epsilon)); \qquad i \in \{1, \ldots, n\} \end{aligned}$$

Lemma 2 *If conditions (1) and (2) hold, then* $\delta_{\mathcal{T}_1 \cup \mathcal{T}_2}(\mathcal{EG}^T(G')) < \delta_{\mathcal{T}_1 \cup \mathcal{T}_2}(\mathcal{EG}^T(G))$.

Proof:
It is obvious that any efficient spanning tree of G' must contain exactly one edge $[v_i, v']$ and all of the edges $[v, v_i]$ together with an efficient spanning tree of G. Therefore we consider the set $\{T_{ij} | i \in \{0, \ldots, n\}, j \in \{1, \ldots, m\}\}$ of spanning trees, where $E(T_{ij}) = E(T_j) \cup [v_i, v'] \cup \{[v, v_l] | l = 0, \ldots, n\}$.
Below we list their costs:

$$\begin{aligned} c(T_{01}) &= (2a_1 \Leftrightarrow x \Leftrightarrow \epsilon, C + 2b_1 \Leftrightarrow b_l + \epsilon) \\ c(T_{0k}) &= (a_1 \Leftrightarrow \epsilon, C + y + b_1 \Leftrightarrow b_l + \epsilon) \\ c(T_{0l}) &= (a_1 + a_l \Leftrightarrow x \Leftrightarrow \epsilon, C + b_1 + \epsilon) \\ c(T_{0m}) &= (a_1 + a_m \Leftrightarrow x \Leftrightarrow \epsilon, C + b_1 + b_m \Leftrightarrow b_l + \epsilon) \\ \\ c(T_{11}) &= (a_1, C + b_1) \\ c(T_{1k}) &= (x, C + y) \\ c(T_{1l}) &= (a_l, C + b_l) \\ c(T_{1m}) &= (a_m, C + b_m) \\ \\ c(T_{21}) &= (x \Leftrightarrow \epsilon, C + y \Leftrightarrow \epsilon) \\ c(T_{2k}) &= (x + (x \Leftrightarrow a_1 \Leftrightarrow \epsilon), C + y + (y \Leftrightarrow b_1 \Leftrightarrow \epsilon)) \\ c(T_{2l}) &= (a_l + (x \Leftrightarrow a_1 \Leftrightarrow \epsilon), C + b_l + (y \Leftrightarrow b_1 \Leftrightarrow \epsilon)) \\ c(T_{2m}) &= (a_m + (x \Leftrightarrow a_1 \Leftrightarrow \epsilon), C + b_m + (y \Leftrightarrow b_1 \Leftrightarrow \epsilon)) \\ \ldots \\ c(T_{n1}) &= (a_1 + (n \Leftrightarrow 1)(x \Leftrightarrow a_1 \Leftrightarrow \epsilon), C + b_1 + (n \Leftrightarrow 1)(y \Leftrightarrow b_1 \Leftrightarrow \epsilon)) \\ c(T_{nk}) &= (x + (n \Leftrightarrow 1)(x \Leftrightarrow a_1 \Leftrightarrow \epsilon), C + y + (n \Leftrightarrow 1)(y \Leftrightarrow b_1 \Leftrightarrow \epsilon)) \\ c(T_{nl}) &= (a_l + (n \Leftrightarrow 1)(x \Leftrightarrow a_1 \Leftrightarrow \epsilon), C + b_l + (n \Leftrightarrow 1)(y \Leftrightarrow b_1 \Leftrightarrow \epsilon)) \\ c(T_{nm}) &= (a_m + (n \Leftrightarrow 1)(x \Leftrightarrow a_1 \Leftrightarrow \epsilon), C + b_m + (n \Leftrightarrow 1)(y \Leftrightarrow b_1 \Leftrightarrow \epsilon)) \end{aligned}$$

We omitted all trees $T_{ij}, i \in \{3, \dots, n \Leftrightarrow 1\}, j \notin \{1, k, l, m\}$. Then we observe that $[T_{i_1,j}, T_{i_2,j}] \in E(\mathcal{EG}^T(G'))$ for $i_1, i_2 \in \{0, \dots, n\}, j \in \{1, \dots, m\}$ if $i_1 \neq i_2$ and that $[T_{i,j_1}, T_{i,j_2}] \in E(\mathcal{EG}^T(G'))$ if $[T_{j_1}, T_{j_2}] \in E(\mathcal{EG}^T(G)$. It is easy to see that

- T_{ik} is dominated by $T_{i+1,1}, i \in \{1, \dots, n \Leftrightarrow 1\}$
- T_{nk} is dominated by T_{1l} due to (1) and (2)
- T_{0k} is efficient, since $\epsilon < x \Leftrightarrow a_1$
- T_{0l} is dominated by T_{11} since $\epsilon < x \Leftrightarrow a_l$

It follows that T_{0k} is not connected to any $T_{ik}, i \in \{1, \dots, n\}$ and thus there are only edges $[T_{0k}, T_{0j}]$ in $\mathcal{EG}^T(G')$ if $[T_k, T_j] \in E(\mathcal{EG}^T(G))$ and T_{0j} is efficient. Therefore $d_{\mathcal{EG}^T(G')}(T_{0k})$ is at least one less than $d_{\mathcal{EG}^T(G)}(T_k)$.

□

Extension 2:
In the second case we consider the situation that (1) or (2) do not hold. Then let $G^* = (V(G) \cup \{v^*, v_1, v_2\}, E(G) \cup \{[v, v_1], [v, v_2], [v_1, v^*], [v_2, v^*]\})$ where v is an arbitrary node of $V(G)$. We assign the following costs to the additional edges:

$$\begin{aligned} c(v, v_i) &= (0, 0); \quad i = 1, 2 \\ c(v_1, v^*) &= (0, \beta) \\ c(v_2, v^*) &= (a_l \Leftrightarrow a_1 \Leftrightarrow \delta, 0) \end{aligned}$$

where $\beta \geq \max\{\frac{x + a_l - 2a_1 - \delta}{x - a_1}, b_1 \Leftrightarrow b_l\}$ and $\delta > 0$ is sufficiently small. Then we can argue as before that all efficient spanning trees of G^* must be contained in $\{T_{ij} | i \in \{1, 2\}, j \in \{1, \dots m\}\}$ where $E(T_{ij}) = E(T_j) \cup \{[v, v_1], [v, v_2], [v_i, v^*]\}$. The costs of these trees are listed below:

$$\begin{array}{rl} c(T_{11}) = (a_1, b_1 + \beta) & c(T_{21}) = (a_l \Leftrightarrow \delta, b_1) \\ \dots & \dots \\ c(T_{1k}) = (x, y + \beta) & c(T_{2k}) = (x + a_l \Leftrightarrow a_1 \Leftrightarrow \delta, y) \\ \dots & \dots \\ c(T_{1l}) = (a_l, b_l + \beta) & c(T_{2l}) = (2a_l \Leftrightarrow a_1 \Leftrightarrow \delta, b_l) \\ \dots & \dots \\ c(T_{1m}) = (a_m, b_m + \beta) & c(T_{2m}) = (a_m + (a_l \Leftrightarrow a_1 \Leftrightarrow \delta), b_m) \end{array}$$

We observe that:

- By the choice of β, T_{1l} is dominated by T_{21}
- T_{1k} and T_{11} are efficient.

Then if T_{2k} is dominated by some other efficient spanning tree of G^* we have the same result as in the first case: $d_{\mathcal{EG}^T(G^*)}(T_{1k})$ is at least one less than $d_{\mathcal{EG}^T(G)}(T_k)$. Otherwise we consider the edge $[T_{1k}, T_{2k}]$ and check conditions (1) and (2):

$$\begin{aligned} x &> \frac{1}{n}(x + (a_l \Leftrightarrow a_1 \Leftrightarrow \delta)) + \frac{n \Leftrightarrow 1}{n}(a_1 + \epsilon) \\ \Leftrightarrow n &> \frac{x + a_l \Leftrightarrow 2a_1 \Leftrightarrow \delta \Leftrightarrow \epsilon}{x \Leftrightarrow a_1 \Leftrightarrow \epsilon} \\ y + \beta &> \frac{1}{n}y + \frac{n \Leftrightarrow 1}{n}(b_1 + \beta + \epsilon) \\ \Leftrightarrow \beta &> (n \Leftrightarrow 1)(b_1 \Leftrightarrow y + \epsilon) \end{aligned}$$

If we choose $n = \lfloor \frac{x+a_l-2a_1-\delta-\epsilon}{x-a_1-\epsilon} \rfloor + 1$ and $\epsilon > 0$ small enough, conditions (1) and (2) hold. Hence after appropriate renumbering of the efficient trees we have exactly the situation of the first case with T_{1k} in the place of T_k and T_{2k} in the place of T_l.

Analogously to Example 1 Extension 1 and Extension 2 can be easily transfered to the shortest path problem by replacing "spanning tree" by "$s \Leftrightarrow v \Leftrightarrow$path" respectively "$s \Leftrightarrow v' \Leftrightarrow$path" and T by P in Definition 2 and in Extensions 1 and 2. Then Lemma 2 can be reformulated as follows:

Lemma 3 *If conditions (1) and (2) hold, then* $\delta_{\mathcal{P}_1 \cup \mathcal{P}_2}(\mathcal{EG}^{P(s,v')}(G')) < \delta_{\mathcal{P}_1 \cup \mathcal{P}_2}(\mathcal{EG}^{P(s,v)}(G))$.

Theorem 5 *For a given graph* $G = (V, E)$ *and costs* $c^1, \ldots, c^Q : E \rightarrow I\!R_+$ *there exists a graph* $\tilde{G}_T$ *and costs* $\tilde{c}^1, \ldots, \tilde{c}^Q : E(\tilde{G}_T) \rightarrow I\!R_+$ *containing* G *as a subgraph such that* $\mathcal{EG}^T(\tilde{G}_T)$ *is not connected.*
Analogously, for a given graph $G = (V, E)$, *vertices* $s, v \in V(G)$ *and costs* $c^1, \ldots, c^Q : E \rightarrow I\!R_+$ *there exists a graph* $\tilde{G}_P$ *and costs* $\tilde{c}^1, \ldots, \tilde{c}^Q : E(\tilde{G}_P) \rightarrow I\!R_+$ *containing* G *as a subgraph such that* $\mathcal{EG}^{P(s,\tilde{v})}(\tilde{G}_P)$ *is not connected.*

Proof:
We apply Extension 1 and, if necessary, Extension 2 iteratively. By Lemma 2 (Lemma 3) it is clear that $\delta_{\mathcal{T}_1 \cup \mathcal{T}_2}(\mathcal{EG}^T(G'))$ $(\delta_{\mathcal{P}_1 \cup \mathcal{P}_2}(\mathcal{EG}^{P(s,v')}(G')))$ decreases at least in every second step. After finitely many steps we have constructed a graph $\tilde{G}$ such that $\mathcal{EG}^T(\tilde{G})$ $(\mathcal{EG}^{P(s,\tilde{v})}(\tilde{G}))$ is disconnected.
It should be noted that after application of Extension 1 or 2 $T_{0k}(P_{0k})$ is still not lexicographically minimal, i.e. not contained in $\mathcal{T}_1 \cup \mathcal{T}_2(\mathcal{P}_1 \cup \mathcal{P}_2)$. The ordering of the spanning trees is without loss of generality, since it is always possible to interchange the first and the second cost function. Thus the assumptions of Extensions 1 and 2 are still fulfilled after each iteration.

$\square$

3 Conclusions and Future Research

First let us note that shortest path and spanning tree problems are not the only discrete multiple criteria problems for which the set of efficient solutions is not connected in general. The method described in Section 2.4 can also be applied to construct examples of nonconnected efficiency graphs for multiple criteria matroid optimization problems, where the matroid is either a partition or a transversal matroid.
Despite the negative results of Theorem 4 and Theorem 5 we remark that according to our experience a disconnected graph $\mathcal{EG}^T(G)$ appears only very rarely. We carried out computational tests together with M. Lind from Aarhus University, Denmark [10]. He implemented a program for finding efficient spanning trees based on the connectedness hypothesis. The approach is as follows: First all extremal efficient spanning trees are found. Then a neighbourhood search is used to find non-extremal efficient spanning trees.
A total of 50 randomly generated graphs with 10 to 50 nodes was tested and no example of a disconnected efficiency graph was found. In these tests we compared the efficient solutions found under the hypothesis of connectedness with all efficient solutions calculated by an enumeration approach.
Therefore we conclude that, although the efficiency graph is not connected in general, a procedure based on the connectedness hypothesis, as proposed in [7] and [11, 1], yields a very good approximation of the set of efficient spanning trees. In many cases all efficient spanning trees will be found and in many others only few will be missing. On the other hand the approach implemented in [10] is much faster than an enumeration approach to find all efficient solutions. Running times were within minutes of CPU-time even for larger graphs of 50 nodes, whereas for some graphs with 50 nodes and even for dense graphs with 20 nodes we were not able to find the set of all efficient spanning trees using the enumeration method within 10 hours of computing time.

With respect to future research topics, we shall focus on two main directions. The first concerns the concept of connectivity. By now we do not know any combinatorial problem for which the connectivity conjecture is true, if the current definition is used. Hence it should be investigated if there exist other concepts of connectivity, by introducing other definitions of adjacency of efficient solutions, e.g. of the spanning tree or shortest path problem, such that the corresponding efficiency graph is connected. Such a concept would then have all the advantages pointed out in Section 1. The second direction of research is related to the question: Do there exist combinatorial optimization problems such that their efficient solutions are "connected"? This holds e.g. for special cases of the spanning tree problem, namely if G contains only one cycle. Then any two spanning trees differ by only one edge and hence $\mathcal{EG}^T(G)$ is connected. But obviously we are interested in problems where the connectivity conjecture is true for every instance.

References

[1] K.A. Andersen, K. Joernsten, and M. Lind. On bicriterion minimal spanning trees: An approximation. Technical Report 95/13, Department of Operations Research, University of Aarhus, 1995.

[2] E.A. Choo and A.D. Atkins. Connectedness in multiple linear fractional programming. *Management Science*, 29:250–255, 1983.

[3] J.G. Ecker and I.A. Kouada. Finding efficient points for linear multiple objective programs. *Mathematical Programming*, 8:375–377, 1975.

[4] M. Ehrgott. On matroids with multiple objectives. *To appear in Opimization*, 1996.

[5] J.P. Evans and R.E. Steuer. A revised simplex method for linear multiple objective programs. *Mathematical Programming*, 5:375–377, 1973.

[6] H.W. Hamacher. Matroid intersections with several objectives. Presented at ECCO VII, Milan, February 21.-23., 1994.

[7] H.W. Hamacher and G. Ruhe. On spanning tree problems with multiple objectives. *Annals of Operations Research*, 52:209–230, 1994.

[8] S. Helbig. On the connectedness of the set of weakly efficient points of a vector optimization problem in locally convex spaces. *Journal of Optimization Theory and Applications*, 65:257–271, 1990.

[9] H. Isermann. The enumeration of the set of all efficient solutions for a linear multiple objective program. *Operations Research Quarterly*, 28(3):711–725, 1977.

[10] M. Lind. Neighbourhood search for the bicriterion spanning tree problem. Program implemented at University of Aarhus and Universität Kaiserslautern, February 1995.

[11] M. Lind. On bicriterion spanning tree problems. Presented at the Twelfth International Conference on MCDM, Hagen, June 18.-23., 1995.

[12] E.Q.V. Martins. On a multicriteria shortest path problem. *European Journal of Operational Research*, 16:236–245, 1984.

[13] P.H. Naccache. Connectedness of the set of nondominated outcomes in multicriteria optimization. *Journal of Optimization Theory and Applications*, 25:459–467, 1978.

[14] A.R. Warburton. Quasiconcave vector maximization: Connectedness of the sets of pareto-optimal and weak pareto-optimal alternatives. *Journal of Optimization Theory and Applications*, 40:537–557, 1983.

[15] P.L. Yu. Cone convexity, cone extreme points and nondominated solutions in decision problems with multiobjectives. *Journal of Optimization Theory and Applications*, 14:319–377, 1974.

[16] P.L. Yu and M. Zeleny. The set of all nondominated solutions in linear cases and a multicriteria simplex method. *Journal of Mathematical Analysis and Applications*, 49:430–468, 1975.