Universal Shortest Paths

Lara Turner and Horst W. Hamacher

Abstract We introduce the universal shortest path problem (Univ-SPP) which generalizes both - classical and new - shortest path problems. Starting with the definition of the even more general universal combinatorial optimization problem (Univ-COP), we show that a variety of objective functions for general combinatorial problems can be modeled if all feasible solutions have the same cardinality. Since this assumption is, in general, not satisfied when considering shortest paths, we give two alternative definitions for Univ-SPP, one based on a sequence of cardinality contrained subproblems, the other using an auxiliary construction to establish uniform length for all paths between source and sink. Both alternatives are shown to be (strongly) NP-hard and they can be formulated as quadratic integer or mixed integer linear programs. On graphs with specific assumptions on edge costs and path lengths, the second version of Univ-SPP can be solved as classical sum shortest path problem.

1 Introduction

The shortest path problem (SPP) is one of the best known models in mathematical optimization with a huge number of publications (see e.g. Zwick [14]). One of the reasons for this is its potential in tackling real-world problems, another its usage as

Lara Turner
Technical University of Kaiserslautern, Department of Mathematics, P. O. Box 3049, 67653
Kaiserslautern, Germany, e-mail: turner@mathematik.uni-kl.de

Horst W. Hamacher
Technical University of Kaiserslautern, Department of Mathematics, P. O. Box 3049, 67653
Kaiserslautern, Germany, e-mail: hamacher@mathematik.uni-kl.de

subproblem in larger optimization problems, but also in other areas like simulation or game theory.

In general, one assumes that SPP is the sum-SPP, where paths in a given directed graph are compared with respect to the sum of their costs. That is, one looks for a path P such that the sum $\sum_{e \in P} c(e)$ of the edge costs along P is minimized. Another well-known SPP is the bottleneck or max-SPP in which we minimize the maximum edge cost $\max_{e \in P} c(e)$ in P. Other versions of shortest path problems include balanced SPP (minimizes the difference between largest and smallest edge cost), k-max SPP or k-sum SPP (Garfinkel et al. [2]) where the latter two determine a path in which the kth-largest and the sum of the k largest edge costs are as small as possible, respectively. More exotic SPPs are the (k, l)-balanced SPP (with minimal difference between k^{th}-largest and l^{th}-smallest edge cost), the $(k + l)$-max SPP (minimizing the sum of kth-largest and l^{th}-largest edge cost) and the (k, l)-trimmed-mean SPP in which the k largest and l smallest edge costs are ignored and the costs of the remaining edges are added. These problems are addressed in some detail in Turner [9].

In the past, several attempts have been made to develop unifying theories with the goal of finding common properties and solution algorithms for a large class of SPPs including sum- and max-SPP. Most notably, this was achieved by the class of algebraic path problems (see e.g. Zimmermann [13], or, more recently, Mohri [5]) where the sum or max operation in the objective function is replaced by a more general bilinear operator and the real numbers are replaced by the elements of a semigroup or semiring.

In this paper, we choose a different approach where the objective function is of the sum type and multiplicative weights are used to model various instances of shortest path problems. In Section 2, we first define such a universal objective function for general combinatorial problems. Since this definition is assuming the same cardinality for all feasible solutions, it cannot be directly carried over to universal path problems, denoted Univ-SPP. But we use this general definition as starting point for two alternative formulations of Univ-SPP in Sections 3 and 4. We analyse the complexity of the resulting versions and study special cases in which the problem is solvable as classical sum-SPP. Integer programming formulations are given in Section 5. The paper is concluded in Section 6 with a summary of our results and some ideas for current and future work.

2 Universal Combinatorial Optimization Problem

We consider the shortest path problem as a special case of a combinatorial optimization problem (COP) given by a finite ground set $E = \{e_1, \ldots, e_m\}$ of cardinality m, a set of feasible solutions $F \subseteq 2^E$ and a cost function $c : E \to \mathbb{R}$ assigning costs $c(e) \in \mathbb{R}$ to the elements in E. The most popular objective functions of combinatorial problems are of the sum and bottleneck type given by $\min_{S \in F} \sum_{e \in S} c(e)$ and

$\min_{S \in F} \max_{e \in E} c(e)$, respectively. These two objectives are special cases of the more general problem introduced next.

Definition 1. Consider a COP in which all feasible solutions $S \in F$ have the same cardinality $|S| \leq m$. For any $S \in F$ we denote with $c_{(i)}(S)$, $i = 1, \dots, |S|$, its ith-largest cost coefficient, i.e. $c_{(1)}(S) \geq \dots \geq c_{(|S|)}(S)$, and with

$$c_{\geq}(S) := (c_{(1)}(S), \dots, c_{(|S|)}(S))$$

its **vector of sorted costs**. Given universal weights $\lambda_i \in \mathbb{R}$, $i = 1, \dots, |S|$, the **universal combinatorial optimization problem, Univ-COP,** is

$$\min_{S \in F} f_\lambda(S) := \sum_{i=1}^{|S|} \lambda_i c_{(i)}(S). \tag{1}$$

A feasible solution $S^* \in F$ minimizing (1) is a **universal optimal solution** with **universal cost $\mathbf{f}_\lambda(\mathbf{S}^*)$.**

To call such problems universal truly makes sense, since, by choosing appropriate universal weights, classical and new objective functions can be represented by (1). Objective functions which can be modeled are, for instance, the standard sum or bottleneck objective for which we set $\lambda_i = 1$ for all $i = 1, \dots, |S|$ or $\lambda_1 = 1$ and $\lambda_i = 0$ otherwise. The balanced objective value $\max_{e \in S} c(e) - \min_{e \in S} c(e)$ for any solution $S \in F$ is formulated by setting $\lambda_1 = 1$, $\lambda_{|S|} = -1$ and $\lambda_i = 0$ else. Other less-known objectives as k-max or k-sum objective function, $k \leq |S|$, are obtained using $\lambda_k = 1$ or $\lambda_1 = \dots = \lambda_k = 1$ and the remaining $\lambda_i = 0$. Table 1 contains a non-exhaustive list of objective functions of type (1), showing the power of the Univ-COP approach (see also Example 1).

The idea of associating weight coefficient λ_i with the ith-largest cost element of a feasible solution $S \in F$ was first introduced by Yager [12] for ordered weighted averaging aggregation operators in multicriteria decisionmaking and later extended to location problems using the notion of ordered median functions (see, for instance, Nickel and Puerto [6]). The concept of universal combinatorial optimization problems was suggested by Kalsch [4] and Hamacher and Turner [3]. This is related to another generalization of combinatorial problems called discrete optimization problems with ordering proposed by Fernández et al. [1]. In [1, 3] examples of Univ-COPs can be found which are in their general form solvable in polynomial time. An example of such a Univ-COP is the universal spanning tree problem (Univ-STP) for which the greedy algorithm can be shown to find a universal optimal solution if $\lambda_i \geq 0$ for all $i = 1, \dots, |S|$. The same is true for the universal matroid base problem, Univ-MBP.

Example 1. We consider the Univ-SPT in the undirected graph $G = (V, E)$ of Figure 1 where the edge labels denote the edge costs $c(e)$ for all $e = [i, j] \in E$. It can be shown that the spanning tree $T = \{[1, 2], [2, 3], [2, 4]\}$ computed by the standard

Table 1 Objective functions modeled by Univ-COP

$\lambda \in \mathbb{R}^{|S|}$	$f_\lambda(S)$	Objective function name
$(1,\ldots,1)$	$\sum\limits_{e \in S} c(e)$	Sum
$(1,0,\ldots,0)$	$\max\limits_{e \in S} c(e)$	Bottleneck/max
$(0,\ldots,0,1)$	$\min\limits_{e \in S} c(e)$	Min
$(\underbrace{1,\ldots,1}_{k},0,\ldots,0)$	$\sum\limits_{i=1}^{k} c_{(i)}(S)$	k-sum, $k \leq |S|$
$(\underbrace{0,\ldots,0}_{k-1},1,0,\ldots,0)$	$c_{(k)}(S)$	k-max, $k \leq |S|$
$(\underbrace{0,\ldots,0}_{k-1},1,0,\ldots,\underbrace{0,1,0,\ldots,0}_{l-1})$	$c_{(k)}(S) + c_{(l)}(S)$	$(k+l)$-max, $k < l \leq |S|$
$(\underbrace{0,\ldots,0}_{k},\underbrace{1,\ldots,1}_{l},0,\ldots,0)$	$\sum\limits_{i=k+1}^{|S|-l} c_{(i)}(S)$	(k,l)-trimmed-mean, $k,l \leq |S|$, $k+l \leq |S|$
$(\underbrace{1,\ldots,1}_{k},0,\ldots,0,\underbrace{1,\ldots,1}_{l})$	$\sum\limits_{i=1}^{k} c_{(i)}(S) + \sum\limits_{i=|S|-l+1}^{|S|} c_{(i)}(S)$	(k,l)-anti-trimmed-mean, $k,l \leq |S|$, $k+l \leq |S|$
$(1,0,\ldots,0,-1)$	$\max\limits_{e \in S} c(e) - \min\limits_{e \in S} c(e)$	Balanced
$(1,0,\ldots,\underbrace{0,-1}_{k-1},0,\ldots,0)$	$\max\limits_{e \in S} c(e) - c_{(k)}(S)$	k-balanced, $1 < k \leq |S|$
$(\underbrace{0,\ldots,0}_{k-1},1,0,\ldots,0,\underbrace{-1,0,\ldots,0}_{l-1})$	$c_{(k)}(S) - c_{(|S|-l+1)}(S)$	(k,l)-balanced, $k,l \leq |S|$, $k+l \leq |S|$
$\vdots$	$\vdots$	$\vdots$

greedy algorithm is universal optimal if all universal weight coefficients λ_i are non-negative. This is e.g. the case for sum, bottleneck, 2-sum, 2-max, or $(1,1)$-trimmed mean objective. The corresponding objective function values are 6 (sum objective), 3 (bottleneck objective), 5 (2-sum objective), and 2 for the 2-max as well as the $(1,1)$-trimmed-mean objective function. Note that tree T is the unique optimal solution for the sum objective while for bottleneck or $(1,1)$-trimmed-mean objective function there exist alternative optimal solutions (e.g. $T' = \{[1,3],[2,3],[2,4]\}$ or

$T'' = \{[1,2],[2,3],[3,4]\}$). The fact that the non-negativity of the universal weights is essential in the validity argument of the Greedy algorithm is shown by the following examples: The greedy solution does not provide an optimal solution of the balanced spanning tree problem which is given by $\{[1,3],[2,4],[3,4]\}$. The $(1,2)$-balanced shortest spanning trees are T' and $T''' = \{[1,2],[1,3],[2,4]\}$ with an objective function value of 0. In both of these examples some of the universal weights are equal to -1 (see Table 1).
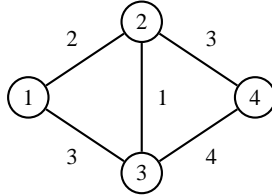

Fig. 1 Undirected graph G

A crucial point in Definition 1 and the validity proofs in [3] is, however, that the cardinality $|S|$ of the feasible solutions is fixed. Obviously, this assumption is, in general, not satisfied for the (s,t)-paths P in a given digraph. This is e.g. illustrated by the balanced SPP. In order to make its objective $\max_{e \in P} c(e) - \min_{e \in P} c(e)$ a special case of (1), one would necessarily have to set $\lambda_1 = 1$ in order to reproduce the first part $\max_{e \in P} c(e)$ of the objective function. To reproduce the second part $-\min_{e \in P} c(e)$ two different universal weight vectors with components $\lambda_l = -1, \lambda_{l'} = 0$ and $\lambda_l = 0, \lambda_{l'} = -1$, respectively, would be required to compare two paths with different number of edges l and l'.

In order to define a universal shortest path problem a modified definition of universality is needed which is based on Definition 1. We propose in the following sections two alternatives of Univ-SPP which can both handle paths of different length.

3 Universal Shortest Path Problem: Sequential Definition

The classic single-source single-sink shortest path problem is a special type of a combinatorial optimization problem where the ground set is given by the edges of a directed graph $G = (V,E)$, the feasible solutions are the paths from source s to sink t and costs $c(e) \in \mathbb{R}$ are assigned to the edges $e \in E$. The set of all (s,t)-paths in G is denoted by $\mathscr{P}_{st}$ and, by definition, such a path P is a sequence of nodes and edges $P = (s = i_0, e_1, i_1, \ldots, i_{l(P)-1}, e_{l(P)}, i_{l(P)} = t)$ with $e_k = (i_{k-1}, i_k) \in E$ for all $k = 1, \ldots, l(P)$ such that neither nodes nor edges are repeated. The number of edges in path P is called its length $l(P)$ which is at most $n - 1 = |V| - 1$. We assume without loss of generality that there are no incoming edges in s and no outgoing edges

from t in graph G.

The first approach consists in splitting Univ-SPP into $n-1$ universal subproblems with fixed cardinality. As for universal combinatorial optimization problems, we first define the vectors of sorted edge costs.

Definition 2. For any path $P \in \mathscr{P}_{st}$ of length $l(P) = l$ with $l \in \{1, \ldots, n-1\}$, the **sorted cost vector** of path P sorts the costs along P in non-increasing order, i.e.

$$c_{\geq}^{l}(P) := (c_{(1)}^{l}(P), \ldots, c_{(l)}^{l}(P))$$

where $c_{(i)}^{l}(P)$, $i = 1, \ldots, l$, is the ith-largest edge cost in path P.

Definition 3. Let $G = (V, E)$ be a directed graph with costs $c(e) \in \mathbb{R}$ for all $e \in E$. For any fixed path length $l \in \{1, \ldots, n-1\}$ and any given universal weight vector $\lambda^{l} \in \mathbb{R}^{l}$ the **universal shortest path problem with cardinality l**, **Univ-SPP(l)**, is

$$\min_{P \in \mathscr{P}_{st} : l(P) = l} f_{\lambda^l}(P) := \sum_{i=1}^{l} \lambda_i^l c_{(i)}^l(P).$$

If there exists no (s,t)-path of length l, we set the optimal objective value of Univ-SPP(l) to ∞ .

Then, Univ-SPP in its sequential definition can be formulated as follows:

Definition 4. If we denote with P_l^* any optimal solution of Univ-SPP(l), the **sequential universal shortest path problem**, **Univ-SPP(1, ..., n − 1)**, is defined with respect to a set of $n-1$ universal weight vectors $\{\lambda^l \in \mathbb{R}^l : l \in 1, \ldots, n-1\}$ as

$$\min_{l \in \{1, \ldots, n-1\}} f_{\lambda^l}(P_l^*). \tag{2}$$

A path $P^* \in \mathscr{P}_{st}$ minimizing (2) is called a **universal shortest path** with **universal cost** $\mathbf{f}_{\lambda^1}(\mathbf{P}^*)$ where $l = l(P^*)$.

Choosing universal weight vectors λ^l for any path length $l \in \{1, \ldots, n-1\}$, classical sum- and bottleneck SPP are modeled in an obvious way by setting $\lambda_i^l = 1$ for all $i = 1, \ldots, l$ or $\lambda_1^l = 1$ and $\lambda_i^l = 0$ otherwise. As indicated in Section 2, $\lambda_1^l = 1$, $\lambda_l^l = -1$ and $\lambda_i^l = 0$ else yields the balanced SPP if we set either $\lambda_1^1 = 0$ or $\lambda_1^1 = \infty$ for paths consisting of only one edge. Similarly for objective functions like k-max or k-sum that require a minimum path length $k \leq n-1$ to be well-defined we can define $\lambda_i^l = 0$ or $\lambda_i^l = \infty$ for all $l < k$ and $i = 1, \ldots, l$ depending on whether paths with less than k edges are assumed to be feasible or not.

The complexity status of the sequential Univ-SPP is easily established, since the SPP with classical sum-objective and cardinality constraint is a special case of Univ-SPP(1, ..., n − 1) and is well-known to be NP-hard.

Theorem 1. *Univ-SPP$(1,\dots,n-1)$ is (strongly) NP-hard.*

Note that the sequential approach reduces to solving a single universal shortest path problem without cardinality constraint if all paths from s to t have the same length. This holds, for instance, in layered graphs which are discussed in Section 4 (see Definition 7). There we will use this fact to propose a solution algorithm which - in contrast to the result of Theorem 1 - has polynomial time complexity.

4 Universal Shortest Paths with Cardinality $|E|$

For the case of non-negative costs $c(e) \geq 0$ which we assume throughout this section we suggest a second definition for Univ-SPP. The main idea is to enforce for all paths from s to t a length of $m = |E|$ where m is the number of edges in graph G. This is achieved by extending each (s,t)-path in the original graph by artificial edges of cost 0. Using this approach, universal shortest path problems can be defined by a single universal weight vector $\lambda \in \mathbb{R}^m$ instead of a sequence of vectors $\lambda^l, l \in \{1,\dots,n-1\}$, as in the first definition given in Section 3. We start by extending the definition of sorted cost vectors.

Definition 5. The **extended sorted cost vector** of a path $P \in \mathscr{P}_{st}$ is given as

$$c_\geq(P) := (c_{(1)}(P),\dots,c_{(l(P))}(P),\underbrace{0,\dots,0}_{m-l(P)})$$

where $c_{(i)}(P)$, $i = 1,\dots,l(P)$, is the ith-largest edge cost in P, i.e. $c_{(1)}(P) \geq \dots \geq c_{(l(P))}(P) \geq 0$, and $c_{(i)}(P) := 0$ for all $i = l(P)+1,\dots,m$.

Definition 6. Given a directed graph $G = (V,E)$ with non-negative costs $c(e) \geq 0$ for all $e \in E$, and a universal weight vector $\lambda \in \mathbb{R}^m$, the **universal shortest path problem with cardinality $|E| = m$**, denoted **Univ-SPP($|E|$)**, is

$$\min_{P \in \mathscr{P}_{st}} f_\lambda(P) := \sum_{i=1}^m \lambda_i c_{(i)}(P) \tag{3}$$

A path $P^* \in \mathscr{P}_{st}$ for which (3) is minimal is called an **extended universal shortest path** with **extended universal cost $\mathbf{f}_\lambda(\mathbf{P}^*)$**.

Since the edge costs $c(e)$, $e \in E$, are non-negative all artificial costs which are equal to 0 are ranked after the costs of the edges in path P. Hence, sum-SPP and bottleneck SPP are modeled correctly by Univ-SPP($|E|$). The k-sum SPP of minimizing the sum of the k largest edge costs in a path is obtained as special case of (3) by setting $\lambda_1 = \dots = \lambda_k = 1$, $\lambda_{k+1} = \dots = \lambda_m = 0$. This definition assigns paths with length $l(P) < k$ the total sum of their edge costs as objective value, coinciding with the definition by Punnen and Aneja [8] for k-sum objectives in general combinatorial optimization. For the k-max SPP, $k \leq m$, which determines a path whose k^{th}-largest

edge cost is as small as possible, we choose $\lambda_k = 1$ and $\lambda_i = 0$ for all $i \neq k$ and obtain an optimal objective value of 0 if a path with less than k edges exists.

A comparison of the two formulations for Univ-SPP, Univ-SPP$(1, \ldots, n-1)$ and Univ-SPP$(|E|)$, will be given in the concluding Section 6 of this paper. In the following, we will derive some results on the general Univ-SPP$(|E|)$.

Lemma 1. *If $\lambda_i \geq 0$ for all $i = 1, \ldots, m$, digraph G can without loss of generality be assumed to have no loops nor parallel edges.*

Proof. The non-negativity of the universal weights λ_i implies that the deletion of loops and the largest (with respect to cost $c(e)$) of two parallel edges does not worsen the objective value of Univ-SPP$(|E|)$. $\quad\square$

The next example shows, however, that the non-negativity assumption in Lemma 1 is necessary in the argumentation for deleting parallel edges.

Example 2. Let $\lambda = (1, -1, 0, 0)$ in the following directed graph $G = (V, E)$.
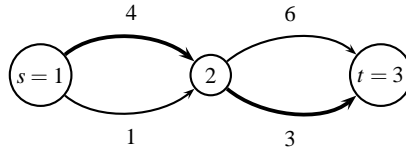

Fig. 2 Directed graph G whose parallel edges cannot be deleted

The extended universal shortest path with respect to $\lambda = (1, -1, 0, 0, 0)$ is indicated in bold. Deleting edge $(s, 2)$ with cost 4 would destroy this path.

In contrast to the classical sum-SPP which can be solved in polynomial time for non-negative edge costs, Univ-SPP$(|E|)$ turns out to be NP-hard.

Theorem 2. *Univ-SPP$(|E|)$ is (strongly) NP-hard.*

Proof. Setting $\lambda_i = -1$ for all $i = 1, \ldots, m$, the longest path problem can be reduced to Univ-SPP$(|E|)$ since for any (s, t)-paths P and P' $\sum_{e \in P} c(e) \geq \sum_{e \in P'} c(e)$, i.e. $\sum_{i=1}^{m} c_{(i)}(P) \geq \sum_{i=1}^{m} c_{(i)}(P')$, if and only if $f_\lambda(P) \leq f_\lambda(P')$. $\quad\square$

Since the k-sum shortest path problem is a special case of Univ-SPP$(|E|)$, Theorem 2 generalizes a result of Garfinkel et al. [2] who prove the NP-hardness of k-sum SPP, albeit for arbitrary costs $c(e) \in \mathbb{R}$. Whether the NP-hardness of Univ-SPP$(|E|)$ is still valid in acyclic digraphs or under the additional assumption that $\lambda_i \geq 0$ for all $i = 1, \ldots, m$ is at this point of time still an open question.

Another important difference compared with sum-SPP is that the optimality principle of dynamic programming which guarantees the correctness of algorithms as Dijkstra or Bellman-Ford is no longer valid. We see this in the following example where a subpath of an extended universal shortest path is not optimal.

Example 3. We consider the directed graph $G = (V,E)$ of Figure 3 and $\lambda = (1,0,2,0,0) \in \mathbb{R}^5$. Obviously, there are only two (s,t)-paths $P = (s,2,4,t)$ and $P' = (s,3,4,t)$ in graph G. The extended universal costs of these paths are $f_\lambda(P) = 4$ and $f_\lambda(P') = 3$, respectively, such that path P' (bold) is optimal for Univ-SPP($|E|$) with $\lambda = (1,0,2,0,0)$. However, its subpath from source s to node 4 has larger objective function value than the corresponding subpath of P (dashed). Consequently, an extended universal shortest path tree does, in general, not exist.
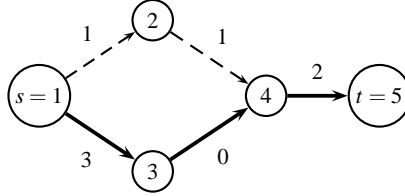

Fig. 3 Extended universal shortest path with non-optimal subpath

Theorem 2 and Example 3 illustrate that the naive transfer of algorithmic ideas from classical sum-SPP to Univ-SPP($|E|$) fails. This motivates the search for more sophisticated methods and the investigation of special cases of Univ-SPP($|E|$). We start with the following obvious observation.

Lemma 2. *Let $\lambda_i \geq 0$ for all $i = 1,\ldots,m$ and let $P, P' \in \mathscr{P}_{st}$ be two (s,t)-paths with extended sorted cost vectors satisfying $c_\geq(P) \leq c_\geq(P')$, i.e. $c_{(i)}(P) \leq c_{(i)}(P')$ for all $i = 1,\ldots,m$. Then $f_\lambda(P) \leq f_\lambda(P')$.*

Example 4 shows, however, that there may be exponentially many incomparable extended sorted cost vectors.

Example 4. Let $G = (V,E)$ be a directed graph with node set $V = \{0,\ldots,n\}$, source $s = 0$, sink $t = n$, and where $n \in 3\mathbb{N}$. The edges of G and its costs are given in Table 4. Here, we assume that $\varepsilon > 0$ is sufficiently small.

Table 2 Edges and costs of digraph G

Edges	Costs
$e = (i-3, i-2), i \in 3\mathbb{N}, i \leq n$	$c(e) = 2^{\frac{i}{3}} - \varepsilon$
$e = (i-3, i-1), i \in 3\mathbb{N}, i \leq n$	$c(e) = 2^{\frac{i}{3}} - \frac{\varepsilon}{2}$
$e = (i-2, i), i \in 3\mathbb{N}, i \leq n$	$c(e) = 2^{\frac{i}{3}} + \varepsilon$
$e = (i-1, i), i \in 3\mathbb{N}, i \leq n$	$c(e) = 2^{\frac{i}{3}} + \frac{\varepsilon}{2}$

We identify blocks $B_i, i = 1,\ldots,\frac{n}{3}$, which are the subgraphs of G consisting of the nodes $\{3(i-1), 3i-2, 3i-1, 3i\}$ and the corresponding four connecting edges. The
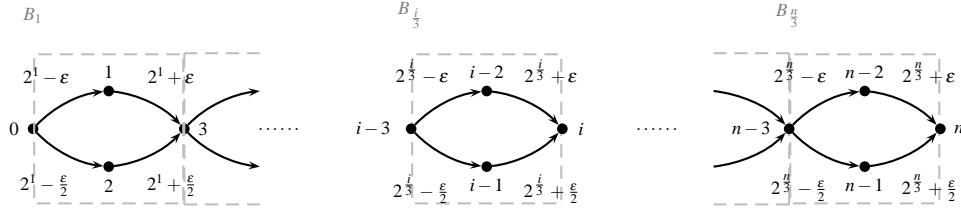
Fig. 4 Digraph G with exponentially many (s,t)-paths

graph G together with its blocks indicated as dashed rectangles is pictured in Figure 4. In this digraph there are $2^{\frac{n}{3}}$ different (s,t)-paths with extended sorted cost vectors

$$c_{\geq}(P) = (c_{(1)}(P),\dots,c_{(2\cdot\frac{n}{3})}(P),\underbrace{0,\dots,0}_{2\cdot\frac{n}{3}})$$

in which the costs of the $2 \cdot \frac{n}{3}$ edges contained in path P are sorted backwards, that is, starting with the largest cost edge into node $t = n$ and following the path until the start node $s = 0$ is reached. Two paths $P, P' \in \mathscr{P}_{st}$ with $P \neq P'$ differ in at least one block B_i, $i \in \{1,\dots,\frac{n}{3}\}$. We assume without loss of generality that path P uses the upper branch in block B_i and path P' uses the lower branch. Using the cost structure along the paths $P \in \mathscr{P}_{st}$, it suffices to consider those components of the extended sorted cost vectors $c_{\geq}(P)$ and $c_{\geq}(P')$ which belong to block B_i (see Figure 5).
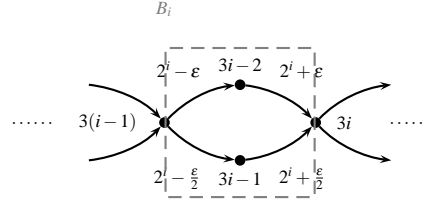

Fig. 5 Block B_i

Since $c_{\geq}(P) = (\dots,2^i + \varepsilon, 2^i - \varepsilon,\dots)$ and $c_{\geq}(P') = (\dots,2^i + \frac{\varepsilon}{2}, 2^i - \frac{\varepsilon}{2},\dots)$ digraph G has exponentially many (s,t)-paths with pairwise incomparable extended sorted cost vectors.

In the special cases which we consider next, one can, however, show that a shortest (s,t)-path P^* with respect to (modified) sum objective always satisfies $c_{\geq}(P^*) \leq c_{\geq}(P)$ for all $P \in \mathscr{P}_{st}$. An extended universal shortest path can thus be found in polynomial time.

Theorem 3. *For any positive real number $\kappa \in \mathbb{R}_+$ and non-negative universal weights $\lambda_i \geq 0$ for all $i = 1,\dots,m$, an extended universal shortest path with respect to*

a) uniform costs, i.e. $c(e) = \kappa$ for all $e \in E$, or
b) binary costs, i.e. $c(e) \in \{0, \kappa\}$ for all $e \in E$

can be found by solving the sum-SPP.

Proof. It is sufficient to prove the claim for binary costs. In this case, the extended sorted cost vectors will collect all edges of P with cost κ in the first components and all the 0's in the remaining components, i.e.

$$c_{\geq}(P) = (\kappa, \ldots, \kappa, 0, \ldots, 0)$$

for all $P \in \mathscr{P}_{st}$. Hence (3) is minimized by finding a sum shortest path from s to t. $\square$

Theorem 3 cannot be generalized to more than one non-zero cost value. The next results shows that another generalization of Theorem 3 is possible, but only if a - rather strict - additional assumption holds for the set of (s,t)-paths in G.

Theorem 4. *Let κ, $\kappa' \in \mathbb{R}_+$ and $\lambda_i \geq 0$ for all $i = 1, \ldots, m$. If $c(e) \in \{\kappa, \kappa'\}$ for all $e \in E$ and all paths $P \in \mathscr{P}_{st}$ have the same length, any optimal solution of sum-SPP is an extended universal shortest path.*

Proof. Assuming without loss of generality that $\kappa > \kappa'$, the extended sorted cost vector of any path $P \in \mathscr{P}_{st}$ is

$$c_{\geq}(P) = (\underbrace{\underbrace{\kappa, \ldots, \kappa}_{|\{e \in P : c(e) = \kappa\}|}, \underbrace{\kappa', \ldots, \kappa'}_{|\{e \in P : c(e) = \kappa'\}|}}_{l}, \underbrace{0, \ldots, 0}_{m-l})$$

where l is the uniform length of the (s,t)-paths in G. Since $c_{\geq}(P)$ and $c_{\geq}(P')$ are comparable with respect to the component-wise ordering for any two (s,t)-paths P and P', there exists a path $P^* \in \mathscr{P}_{st}$ with $c_{\geq}(P^*) \leq c_{\geq}(P)$ for all $P \in \mathscr{P}_{st}$. If $c_{\geq}(P^*) < c_{\geq}(P)$, we find a smallest $i^* \in \{1, \ldots, l\}$ with $c_{(i^*)}(P^*) = \kappa' < \kappa = c_{(i^*)}(P)$ and $c_{(i)}(P^*) \leq c_{(i)}(P)$ for any $i \neq i^*$. In particular, this implies that $\sum_{e \in P^*} c(e) < \sum_{e \in P} c(e)$, i.e. P^* is a shortest path from s to t with respect to sum objective. The claim follows since $c_{\geq}(P^{**}) = c_{\geq}(P^*)$ for any other sum shortest (s,t)-path P^{**}. $\square$

To assume equal lenghts for all (s,t)-paths in Theorem 4 is, however, essential.

Example 5. In graph $G = (V, E)$ of Figure 6 with $P = (s, 2, t)$ (dashed) and $P' = (s, t)$ (bold), the extended sorted cost vectors are $c_{\geq}(P) = (1, 1, 0)$ and $c_{\geq}(P') = (3, 0, 0)$. Path P is a sum shortest (s,t)-path, but it is not optimal for $\lambda = (0, 1, 0)$. Note that $c_{\geq}(P)$ and $c_{\geq}(P')$ are incomparable, i.e. $c_{\geq}(P) \nleq c_{\geq}(P')$ and $c_{\geq}(P') \nleq c_{\geq}(P)$.

Since Theorem 3 cannot be carried over to more than two different edge costs, we propose for this case a different approach. We first show how to tackle Univ-SPP($|E|$) in layered graphs where all (s,t)-paths have the same length l and then
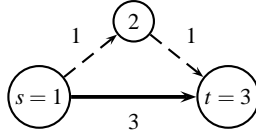
Fig. 6 Two paths with incomparable extended sorted cost vectors

apply this result to a special layered graph closely related to a given graph of arbitrary structure. Note however that the result on layered graphs is interesting in its own right, since it can be used as model within network optimization (for instance in the complexity analysis of maximal flow algorithms) and for real-world applications like evacuation planning.

Definition 7. A directed graph $G = (V, E)$ with source s and sink t is a **layered graph** if its nodes can be partitioned into sets $V_0 = \{s\}, \ldots, V_l = \{t\}$ called **layers** and all its edges run between consecutive layers, i.e. if node i is in layer V_k for some $k \in \{0, \ldots, l-1\}$ and $e = (i, j) \in E$, then j is a node in layer V_{k+1}.

Definition 8. A graph G has **monotone (s, t)-costs** if $c(e_1) \geq \ldots \geq c(e_{l(P)})$ or $c(e_1) \leq \ldots \leq c(e_{l(P)})$ for all paths $P \in \mathscr{P}_{st}$.

Theorem 5. *If G is a layered graph with monotone (s, t)-costs, and $\lambda \in \mathbb{R}^m$, Univ-SPP($|E|$) can be solved in polynomial time as sum-SPP with respect to modified costs.*

Proof. We first consider the case of non-increasing (s, t)-costs. Since G is a layered graph, any path $P \in \mathscr{P}_{st}$ has length l and the monotonicity of the cost implies

$$c_{\geq}(P) = (\underbrace{c(e_1), \ldots, c(e_l)}_{l}, \underbrace{0, \ldots, 0}_{m-l})$$

and

$$f_{\lambda}(P) = \sum_{i=1}^{l} \lambda_i c(e_i).$$

Thus, an optimal solution to Univ-SPP($|E|$) is a shortest (s, t)-path with respect to the sum objective with modified costs $c_{\lambda}(e) := \lambda_k c(e)$ for all $e = (i, j) \in E$ with $i \in V_{k-1}, j \in V_k$ and $k \in \{1, \ldots, l\}$. Since the layered graph is acyclic, the latter problem can be solved in polynomial time.
In the case of non-decreasing (s, t)-costs we can use the same arguments as before by setting $c_{\lambda}(e) := \lambda_{l-k+1} c(e)$.   □

In the next step, we show how the previous result obtained for layered graphs can be generalized.

Definition 9. The **expanded graph** $G' = (V', E')$ is a layered graph obtained from $G = (V, E)$ by including $m + 1$ copies of each node $i \in V$, i.e.

$$V' := \{i(k) : i \in V, k = 0, \ldots, m\},$$

m copies of each edge $(i, j) \in E$

$$E_1' := \{(i(k), j(k+1)) : (i, j) \in E, k = 0, \ldots, m-1\},$$

and m edges between consecutive node copies $t(k)$ and $t(k+1)$ of the sink node t

$$E_2' := \{(t(k), t(k+1)), k = 0, \ldots, m-1\}.$$

The edge set of G' is $E' := E_1' \cup E_2'$ with costs

$$c'(i(k), j(k+1)) := c_{ij}$$
$$c'(t(k), t(k+1)) := 0$$

for all $i, j \in V$ and $k = 0, \ldots, m-1$.

Example 6 shows a digraph G and its expanded graph G'. It is a layered, and thus acyclic, digraph with $n(m+1)$ nodes and $m(m+1)$ edges, where nodes $s(0)$ and $t(m)$ take on the role of source and sink node, respectively. Note that in specific instances of graphs G, the size of the expanded graph G' can be reduced considerably by a preprocessing step in which all edges are eliminated from G (and thus not copied) which can never occur in a path from s to t.

We observe the following relation between paths in G and G'.

Lemma 3. *Let G be a digraph and let G' be its expanded graph.*

a) *For any (s,t)-path P in G, there exists a path P' from $s(0)$ to $t(m)$ in graph G' which consists of exactly m edges.*
b) *For any path P' from $s(0)$ to $t(m)$ in G', there exists a walk W, i.e. a non-simple path in which node and edge repetition is allowed, from s to t in graph G with at most m edges.*
c) *A 1:1-correspondence between (s,t)-paths in G and $(s(0),t(m))$-paths in G' is only given if digraph G is acyclic.*

Proof. Claim a) follows by including $m - l(P)$ many edges from the set E_2'. Since copies of nodes and edges may be used more than once in a given $(s(0), t(m))$-path P' the resulting list of nodes and edges in G may contain cycles such that claim b) follows only for walks and not for paths in G. If G does not contain cycles, node and edge repetition cannot occur such that claim c) holds. $\square$

By construction, the extended sorted cost vectors of (s,t)-paths in G $((s(0),t(m))$-paths in G') and their equivalents in G' (in G) coincide. This motivates to solve Univ-SPP($|E|$) in the expanded graph G'. The following lemma relates the extended universal costs of walks and corresponding paths in digraph G.

Lemma 4. *Let W be an (s,t)-walk in graph G and let P be the (s,t)-path obtained from W by eliminating all cycles. If $\lambda_i \geq 0$ for all $i = 1, \ldots, m$, then $f_\lambda(P) \leq f_\lambda(W)$.*

Proof. Removing all cycles in walk W, we obtain the (simple) path P with $l(P) < l(W)$ and $c_{(i)}(P) \leq c_{(i)}(W)$ for all $i = 1,\ldots,m$. By the non-negativity of the λ_i and Lemma 2, it follows that $f_\lambda(P) \leq f_\lambda(W)$.   □

Theorem 6. *Univ-SPP($|E|$) in G and G' are equivalent if digraph G is acyclic or $\lambda_i \geq 0$ for all $i = 1,\ldots,m$. If the costs $c(e)$ are, in addition, monotone along (s,t)-paths and (s,t)-walks, Univ-SPP($|E|$) can be solved as sum-SPP with respect to modified costs.*

Proof. By Lemma 3, any (s,t)-path in digraph G has an equivalent path P' from $s(0)$ to $t(m)$ in G' with $c_\geq(P') = c_\geq(P)$. Conversely, any path P' from $s(0)$ to $t(m)$ in the expanded graph G' corresponds to a walk W from s to t in G with at most m edges and $c_\geq(W) = c_\geq(P')$. By Lemma 4 this walk W can be chosen as a path for general graphs and non-negative universal weights or is already a path for acyclic graphs. For monotone costs which are non-increasing the final claim follows immediately from Theorem 5 applied to the layered graph G'. If the costs are non-decreasing, the same argument is applied to a modified expanded graph where the edge set E_2' connects copies of the source nodes instead of the sink nodes by edges $(s(k),s(k+1))$, $k = 0,\ldots,m-1$.   □

But the equivalence result of Theorem 6 is, in general, no longer true if we allow cycles and negative universal weight coefficients λ_i.

Example 6. Let $\lambda = (1,-1,0,0,0)$ and consider Univ-SPP($|E|$) in the directed graph $G = (V,E)$ of Figure 7.
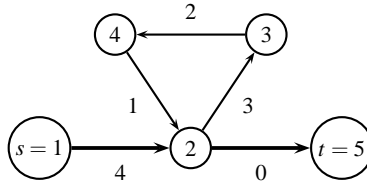

Fig. 7 Directed graph G

Since $P = (s,2,t)$ (bold) is the only feasible (s,t)-path in G it is optimal with extended universal cost $f_\lambda(P') = 4$. In the expanded graph G' which is depicted in Figure 8 the path P' (bold) given by the nodes $P' = (s(0),2(1),3(2),4(3),2(4),t(5))$ is optimal with $f_\lambda(P') = 1$. This path corresponds, however, to walk $W = (s,2,3,4,2,t)$ in digraph G.

5 Integer Programming Formulations

It is well-known that the classical sum shortest path problem can be solved as the following linear program if there are no negative dicycles in graph G:
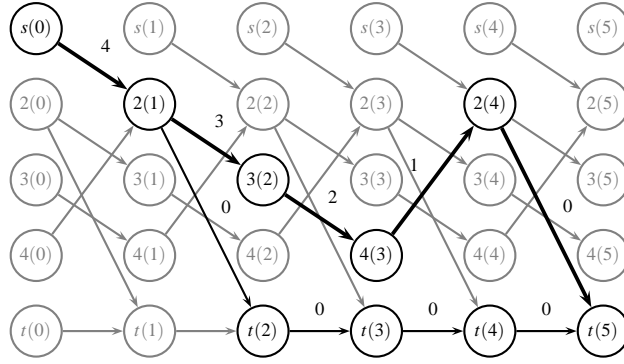
Fig. 8 Expanded graph G'

$$\min \quad \sum_{(i,j)\in E} c_{ij}x_{ij}$$

$$\text{s.t.} \quad \sum_{j\in\delta^+(i)} x_{ij} - \sum_{j\in\delta^-(i)} x_{ji} = \begin{cases} 1 & \text{if } i=s \\ -1 & \text{if } i=t \\ 0 & \text{if } i\neq s,t \end{cases} \tag{4}$$

$$x_{ij} \geq 0 \qquad\qquad\qquad\qquad \forall (i,j) \in E.$$

Since the coefficient matrix induced by the flow conservation constraints (4) is totally unimodular, an integer optimal solution $x^* \in \{0,1\}^m$ is found by choosing an optimal basic solution. This corresponds to a path $P^* \in \mathscr{P}_{st}$ with $(i,j) \in P^*$ if and only if $x_{ij}^* = 1$.

Unlike sum-SPP, Univ-SPP$(|E|)$ cannot be solved as linear program since the additional sorting problem makes the objective function non-linear. We next present a quadratic integer programming formulation:

$$\min \quad \sum_{k=1}^{m} \lambda_k \sum_{(i,j)\in E} s_{k,ij} c_{ij} x_{ij}$$

$$\text{s.t.} \quad \sum_{k=1}^{m} s_{k,ij} = 1 \qquad\qquad\qquad\qquad \forall (i,j) \in E$$

$$\sum_{(i,j)\in E} s_{k,ij} = 1 \qquad\qquad\qquad\qquad \forall k = 1,\ldots,m$$

$$\sum_{(i,j)\in E} s_{k,ij} c_{ij} x_{ij} \geq \sum_{(i,j)\in E} s_{k+1,ij} c_{ij} x_{ij} \qquad \forall k = 1,\ldots,m-1 \tag{5}$$

$$\sum_{j\in\delta^+(i)} x_{ij} - \sum_{j\in\delta^-(i)} x_{ji} = \begin{cases} 1 & \text{if } i=s \\ -1 & \text{if } i=t \\ 0 & \text{if } i\neq s,t \end{cases} \tag{6}$$

$$\sum_{i \in S} \sum_{j \in S} x_{ij} \leq |S| - 1 \qquad\qquad \forall S \subseteq V, |S| \geq 2 \qquad\qquad (7)$$

$$s_{k,ij} \in \{0,1\} \qquad\qquad \forall k = 1, \ldots, m, (i,j) \in E$$

$$x_{ij} \in \{0,1\} \qquad\qquad \forall (i,j) \in E. \qquad\qquad (8)$$

The flow conservation constraints (6) together with the subtour elimination constraints (7) and the binary constraints (8) ensure that a feasible solution x is indeed a path from s to t. The sorting variables $s_{k,ij}$, where $s_{k,ij} = 1$ if edge (i,j) is at position k of the corresponding extended sorted cost vector and $s_{k,ij} = 0$ else, guarantee that the edge costs along any path are sorted correctly (5). Standard linearization techniques can be used to obtain various equivalent mixed integer linear programs similar to those for discrete ordered median location problems, see Nickel and Puerto [6].

Another mixed integer linear programming formulation for Univ-SPP($|E|$) is based on Ogryczak and Tamir [7]. Using a partial sum reformulation of the universal objective function and $\lambda_{m+1} = 0$ it is only valid for non-increasing and non-negative universal weight coefficients λ_i. Details can be found in Turner [9].

$$\min \quad \sum_{k=1}^{m} (\lambda_k - \lambda_{k+1}) \left(kt_k + \sum_{(i,j) \in E} d_{ij,k} \right)$$

$$\text{s.t.} \quad d_{ij,k} \geq c_{ij} x_{ij} - t_k \qquad\qquad \forall (i,j) \in E, k = 1, \ldots, m$$

$$\sum_{j \in \delta^+(i)} x_{ij} - \sum_{j \in \delta^-(i)} x_{ji} = \begin{cases} 1 & \text{if } i = s \\ -1 & \text{if } i = t \\ 0 & \text{if } i \neq s,t \end{cases}$$

$$\sum_{i \in S} \sum_{j \in S} x_{ij} \leq |S| - 1 \qquad\qquad \forall S \subseteq V, |S| \geq 2 \qquad (9)$$

$$d_{ij,k} \geq 0 \qquad\qquad \forall (i,j) \in E, k = 1, \ldots, m$$

$$t_k \in \mathbb{R} \qquad\qquad \forall k = 1, \ldots, m$$

$$x_{ij} \in \{0,1\} \qquad\qquad \forall (i,j) \in E.$$

In both programs the subtour elimination constraints (7) and (9) can be deleted if the given digraph is acyclic or $\lambda_i \geq 0$ for all $i = 1, \ldots, m$ (compare Theorem 6 in Section 4). Both formulations can be adapted to describe the subproblems Univ-SPP(l) of the sequential definition Univ-SPP($1, \ldots, n-1$).

6 Conclusions

We have defined a universal objective function by sorting the edge costs along any path in non-increasing order and assigning, in sequence, universal weight coeffi-

cients to these ordered cost values. In this way we can model many types of shortest path problems.

While the classical sum shortest path problem is a special case of a combinatorial optimization problem, we have seen that its universal version, Univ-SPP, is not a special case of the corresponding universal combinatorial optimization problem, Univ-COP, in which we have assumed a fixed cardinality for all feasible solutions. For the universal shortest path problem we have, therefore, proposed two alternative formulations which are compatible with the definition of Univ-COP.

The first one, Univ-SPP$(1,\ldots,n-1)$, is based on solving a sequence of cardinality constrained subproblems Univ-SPP(l) with respect to universal weight vectors $\lambda^l \in \mathbb{R}^l, l = 1,\ldots,n-1$. Compared with the second approach, Univ-SPP$(|E|)$, in which sorted cost vectors (and paths) have been extended to cardinality m, the sequential definition seems to be more flexible with respect to modeling possibilities, but also more complex due to the additional cardinality constraints in the subproblems Univ-SPP(l). Cardinality-sensitive objective functions using i^{th}-smallest edge costs cannot be formulated by Univ-SPP$(|E|)$ since position i cannot be fixed in the extended sorted cost vector. An example for such a problem with cardinality-sensitive objective is the balanced SPP which can only be modeled by Univ-SPP$(1,\ldots,n-1)$ setting $\lambda_1^1 = 0$ or $\lambda_1^1 = \infty$ in Univ-SPP(1), $\lambda_1^l = 1, \lambda_l^l = -1$ and $\lambda_i^l = 0$ else for Univ-SPP(l) with $l \neq 1$.

Two IP formulations for Univ-SPP were suggested in Section 5. The polyhedral structure of the resulting feasibility polyhedra and the consequences of linearization strategies are under research, see Turner [9]. In this thesis, universal shortest path problems for special choices of universal weights λ such as $(k+l)$-max SPP, (k,l)-balanced SPP or (k,l)-trimmed-mean SPP will be discussed in detail. These problems can be solved in polynomial time applying results on k-max and k-sum optimization for general combinatorial problems. Further generalizations of the balanced objective function can be found in Turner et al. [10].

References

1. E. Fernández, J. Puerto, and A. M. Rodríguez-Chía. On discrete optimization with ordering. 2008. Unpublished manuscript.
2. R. Garfinkel, E. Fernández, and T. J. Lowe. The k-centrum shortest path problem. *Top*, 14(2):279–292, 2006.
3. H. W. Hamacher and L. Turner. Universal combinatorial optimization problems. 2009. Unpublished manuscript.
4. M. T. Kalsch. *Scheduling-Location (ScheLoc): Models, Theory and Algorithms*. PhD thesis, Technical University of Kaiserslautern, 2009.
5. M. Mohri. Semiring frameworks and algorithms for shortest-distance problems. *Journal of Automata, Languages and Combinatorics*, 7(3):321–350, 2002.
6. S. Nickel and J. Puerto. *Location theory: A unified approach*. Springer, 2005.
7. W. Ogryczak and A. Tamir. Minimizing the sum of the k largest functions in linear time. *Information Processing Letters*, 85(3):117–122, 2003.
8. A. P. Punnen and Y. P. Aneja. On k-sum optimization. *Operations Research Letters*, 18(5):233–236, 1996.

9. L. Turner. *Universal combinatorial optimization problems*. PhD thesis, Technical University of Kaiserslautern. To appear.

10. L. Turner, A. P. Punnen, Y. P. Aneja, and H. W. Hamacher. On generalized balanced optimization problems. 2010. To appear in Mathematical Methods of Operations Research.

11. S. Velten. *Discrete location problems with flexible objective*. PhD thesis, Saarland University, 2008.

12. R. R. Yager. On ordered weighted averaging aggregation operators in multicriteria decision-making. *IEEE Transactions on Systems, Man and Cybernetics*, 18(1):183–190, 1988.

13. U. Zimmermann. *Linear and combinatorial optimization in ordered algebraic structures*, volume 10 of *Annals of Discrete Mathematics*. North-Holland, 1981.

14. U. Zwick. Exact and approximate distances in graphs - A survey. In *Algorithms - ESA 2001*, volume 2161 of *Lecture Notes in Computer Science*, pages 33–48. Springer, 2001.