Min-Max Connected Multiway Cut

Petr Kolman
Charles University, Faculty of Mathematics and Physics, Department of Applied Mathematics, Czechia

Hans Raj Tiwary
Charles University, Faculty of Mathematics and Physics, Department of Applied Mathematics, Czechia

Abstract

We introduce a variant of the multiway cut that we call the min-max connected multiway cut. Given a graph $G = (V, E)$ and a set $\Gamma \subseteq V$ of t terminals, partition V into t parts such that each part is connected and contains exactly one terminal; the objective is to minimize the maximum weight of the edges leaving any part of the partition. This problem is a natural modification of the standard multiway cut problem and it differs from it in two ways: first, the cost of a partition is defined to be the maximum size of the boundary of any part, as opposed to the sum of all boundaries, and second, the subgraph induced by each part is required to be connected. Although the modified objective function has been considered before in the literature under the name min-max multiway cut, the requirement on each component to be connected has not been studied as far as we know.

We show various hardness results for this problem, including a proof of weak NP-hardness of the weighted version of the problem on graphs with tree-width two, and provide a pseudopolynomial time algorithm as well as an FPTAS for the weighted problem on trees. As a consequence of our investigation we also show that the (unconstrained) min-max multiway cut problem is NP-hard even for three terminals, strengthening the known results.

2012 ACM Subject Classification Mathematics of computing → Discrete mathematics; Theory of computation → Design and analysis of algorithms

Keywords and phrases Graphs, Partitions, Clustering, Complexity, Algorithms, Approximations

Digital Object Identifier 10.4230/LIPIcs...

Funding *Petr Kolman*: Partially supported by grant 24-10306S of GA ČR.
Hans Raj Tiwary: Partially supported by the AGATE project funded from the Horizon Europe Programme under Grant Agreement No. 101183743.

1 Introduction

Cuts in graphs are a fundamental concept in computer science, playing a crucial role in both theory and numerous applications [23]. They are used for defining important graph parameters such as bisection, expansion, and tree-width, and have stimulated the development of various algorithmic techniques, including linear programming, divide and conquer, and semidefinite relaxations. Graph cuts are employed in diverse areas such as computer graphics, parallel and distributed systems, transportation, military and agriculture applications, analysis of social networks, and VLSI design, to name a few.

Generally speaking, in usual cut problems, the goal is to *separate* some (pairs of) vertices, and it is irrelevant what happens to the other pairs - whether they end up in the same or in different connected components. However, in applications such as image segmentation, route planning, power grid management, forest planning and harvest scheduling, and market zoning, it is important that parts of G obtained by the cut are *connected* [7, 14, 5, 25, 12].

The Min-Max Connected Multiway Cut problem combines these two orthogonal requirements: separating certain graph parts while keeping others connected. This kind of combination of separation and connectivity constraints is a natural extension of classical problems and brings interesting challenges. Though each of the two constraints is relatively well-understood *individually*, and in some cases, the fulfillment of the connectivity requirement comes for free as a bonus of the optimality of the cut objective (see Related Results), the *combination* of the requirements in the general setting makes the problems challenging. Surprisingly, little theoretical research has addressed both requirements simultaneously.

1.1 Terminology

A family of disjoint subsets $S_1, \ldots, S_k \subseteq V$ is a *partition* of the set V if $\bigcup_{i=1}^{k} S_i = V$. For a graph $G = (V, E)$ and a subset $S \subseteq V$ of vertices, $G[S]$ is the subgraph of G induced by the subset S. For a partition $S_1, \ldots, S_k$ of the vertex set of a graph $G = (V, E)$, $E(S_1, \ldots, S_k) = \{\{u, v\} \mid u \in S_i, v \in S_j, i \neq j\}$ denotes the set of edges between the parts $S_1, \ldots, S_k$. For a positive integer k, $[k] = \{1, \ldots, k\}$.

Given a graph $G = (V, E)$ and a subset $\Gamma = \{t_1, \ldots, t_k\}$ of k vertices, called *terminals*, the *Connected Multiway Cut* is a partition $S_1, \ldots, S_k$ of V such that

1. for each $i \in [k]$, $t_i \in S_i$, and
2. for each $i \in [k]$, $G[S_i]$ is connected.

Without the second requirement, we simply have a Multiway Cut [4, 24]. The *cost of a Multiway Cut* $C = \{S_1, \ldots, S_k\}$ of a graph $G = (V, E)$ with edge weights $w : E \to \mathbb{N}^+$ is $\mathrm{cost}(C) = \max_i \delta(S_i)$ where $\delta(S) = \sum_{e \in E(S, V \setminus S)} w(e)$ denotes the sum of weights of edges between S and $V \setminus S$. We define the *cost of a Connected Multiway Cut* analogously. The *Min-Max Connected Multiway Cut* problem is to find a Connected Multiway Cut of minimum cost; the minimum cost is denoted $\mathrm{MCMC}(G, T)$. In the unweighted version of the problem, all edges have weight one.

1.2 Our Results

The NP-hardness of the min-max connected multiway cut on general graphs with at least four terminals follows using the proof of the NP-hardness of the min-max multiway cut by Svitkina and Tardos [24] without any modification. Therefore, we focus our attention on either simpler graphs or fewer number of terminals. In particular, we show the following new results.

- We show that the min-max connected multiway cut problem is weakly NP-hard for three terminals on graphs of tree-width three. We also show the same weak NP-hardness for the (unconstrained) min-max multiway cut problem.
- We show that the min-max connected multiway cut problem remains weakly NP-hard on graphs of tree-width two if one allows arbitrary number of terminals.
- We give a pseudopolynomial algorithm for the weighted problem on trees complementing our hardness results for graphs of tree-width two.
- We give an FPTAS for the weighted problem on trees.
- We show that if the number of terminals is fixed, then the weighted problem is polynomial time solvable on trees. We do this by showing that the problem is FPT on trees when parameterized by the number of terminals.
- We give some evidence why the weighted problem might be hard even on trees by showing superpolynomial extension complexity for a natural polytope associated with the problem, thereby ruling out many natural LP formulations of polynomial size, and further, we show that an exact version of the weighted problem is NP-hard on trees.

1.3 Related Results

One of the most studied cut problems is the minimum $s-t$ cut problem. Given a graph $G=(V,E)$ and two distinct vertices s and t, find a partition $S, V\setminus S$ of the vertex set V such that $s\in S$, $t\in V\setminus S$, and $E(S,V\setminus S)$ is minimized. Given a minimum cut, we observe that both parts S and $V\setminus S$ are connected; otherwise, there would be a strictly smaller cut. Garg [11] provides a polyhedral description of all $s-t$ cuts in which the s-side is connected, all $s-t$ cuts in which the t-side is connected, and all $s-t$ cuts in which both sides are connected; we note that the polyhedra are unbounded.

For the global minimum cut problem where the task is to find a non-trivial partition $S, V\setminus S$ minimizing $E(S,V\setminus S)$, the optimum solution has again both sides connected.

The situation is different for the maximum cut problem where the optimality of a solution does not imply connectivity of either side. This motivated the definitions of two related problems. In the bond problem, the task is to find a partition $S, V\setminus S$ maximizing $E(S,V\setminus S)$ with both sides connected [8, 17]; in the maximum connected cut problem, the objective is the same but the connectivity constraint is imposed on one side only [13, 22]. It should be noted that the connectivity constraints make the maximum cut problem harder: in contrast to the maximum cut without the connectivity constraints, both versions with the connectivity constraints are NP-hard even on planar bipartite graphs, and the bond problem does not admit a constant factor approximation algorithm, unless P = NP [8]. Both problems are FPT with respect to the tree-width [8].

When cutting a graph into more than two parts, the most common objective minimizes the total weight of deleted edges. In contrast, the min-max multiway cut problem, mentioned earlier, minimizes the maximum weight of deleted edges adjacent to any single part. The best known approximation algorithm for this problem, due to Bansal et al. [4], achieves approximation ratio $O(\sqrt{\log n \log |\Gamma|})$, improving over $O(\log^2 n)$-approximation by Svitkina and Tardos [24].

1.4 Spanning Tree Congestion Problem

One of our motivations for dealing with the min-max connected multiway cut is its close connection to the *Spanning Tree Congestion* problem which is defined as follows [19]: given a graph $G=(V,E)$, construct a spanning tree T of G minimizing its maximum edge congestion where the congestion of an edge $e\in T$ is the number of edges $uv\in E$ in G such that the unique path between u and v in T passes through e; the optimal value for a given graph G is denoted $\mathrm{STC}(G)$.

The problem was introduced under different names in the late 1990s but till today, the computational complexity of the problem is not much understood: the problem is known to be NP-hard even for graphs of maximum degree at most three [2], there is $O(\Delta\cdot\log^{3/2} n)$-approximation algorithm for graphs of maximum degree Δ [16] but the best known approximation ratio for general graph is $n/2$ only [20]; interestingly, this is the approximation ratio achieved by *any* spanning tree.

The following two lemmas capture the connection between the two problems.

▶ **Lemma 1.** *Let $G=(V,E)$ be a connected graph, τ a spanning tree minimizing the spanning tree congestion, $r\in V$ any non-leaf vertex of τ, and $\Gamma=\{t_1,\dots,t_k\}\subset V$ the set of neighbors of r in τ. Then*

$$\mathrm{STC}(G)\geqslant \mathrm{MCMC}(G\setminus\{r\},\Gamma)+1\ .$$

Proof. Let $V_1, \ldots, V_k$ be the connected components of $\tau \setminus \{r\}$; note that they constitute a connected multiway cut for the instance $(G \setminus \{r\}, \Gamma)$. For each $i \in [k]$, the spanning tree congestion of the edge $\{r, t_i\}$ in τ is exactly $\delta(V_i) + 1$. Thus, we obtain the desired bound

$$\mathrm{STC}(G) \geqslant \max_{i=1,\ldots,k} \delta(V_i) + 1 \geqslant \mathrm{MCMC}(G \setminus \{r\}, \Gamma) + 1 .$$

◀

▶ **Lemma 2.** *Let $G = (V, E)$ be a connected graph, $r \in V$ a vertex of G, $\Gamma = \{t_1, \ldots, t_l\} \subset V$ the set of neighbors of r in G, and $V_1, \ldots, V_l$ an optimal solution of the Min-Max Connected Multiway Cut instance (G, Γ). Then*

$$\mathrm{STC}(G) \leq \mathrm{MCMC}(G \setminus \{r\}, \Gamma) + \max\left\{1, \max_{i=1,\ldots,l} \mathrm{STC}(G[V_i])\right\} .$$

Proof. For each $i \in [l]$, let τ_i be an optimal spanning tree of subgraph $G[V_i]$; since each of these subgraphs is connected, all these trees exist. Let τ be the subgraph of G obtained as the union of the disjoint trees $\tau_1, \ldots, \tau_l$ with all the edges adjacent to r. Then, by construction, τ is a spanning tree of G. For each $i \in [l]$, the congestion of the edge $\{r, t_i\}$ in τ is $1 + \mathrm{MCMC}(G \setminus \{r\}, \Gamma)$, and the congestion of any edge in τ_i is at most $\mathrm{STC}(G[V_i]) + \mathrm{MCMC}(G \setminus \{r\}, \Gamma)$ which completes the proof. ◀

2 General Graphs

In this section, we show some NP-hardness results for the min-max connected multiway cut problem for general graphs. Among other results, we show that the problem remains NP-hard for graphs of tree-width two.

2.1 Min-Max Cut vs. Min-Max Connected Cut

For at most three terminals the min-max multiway cut and the min-max connected multiway cut problems are equivalent in the following sense.

▶ **Lemma 3.** *Let G be a connected graph and Γ a set of at most three terminals. For any multiway cut C for the instance (G, Γ), there exists a connected multiway cut C' for the same instance such that $cost(C') \leqslant cost(C)$.*

Proof. We prove the claim for three terminals. For two terminals, the proof is analogous.

Let C be a multiway cut and let V_1, V_2, V_3 be the corresponding partition of $V(G)$. If $G[V_i]$ is connected for each $i \in \{1, 2, 3\}$ the lemma holds with $C' = C$.

So, without loss of generality, we assume that $G[V_1]$ is not connected and let W_1 and W_2 be a partition of V_1 such that $t_1 \in W_1$ and $E(W_1, W_2) = \emptyset$. We are going to define a new connected partition V_1', V_2', V_3' of $V(G)$. Let $V_1' = W_1$. If $e(W_2, V_2) \geqslant e(W_2, V_3)$, we define $V_2' = V_2 \cup W_2$ and $V_3' = W_3$, and $e(W_2, V_2) < e(W_2, V_3)$, we define $V_2' = V_2$ and $V_3' = V_3 \cup W_2$. In both cases, we obtain a feasible solution of the min-max multiway cut that has fewer connected components and whose cost is not higher than the cost of (V_1, V_2, V_3). Applying this procedure repeatedly, we end up with a partition where each induced subgraph is connected and whose cost is at most $\mathrm{cost}(C)$. ◀

The above lemma implies that an optimal solution for the min-max multiway cut problem can always be assumed to be a connected multiway cut when we have at most three terminals.

For two terminals, the problem reduces to the standard $s-t$ cut and therefore can be solved in polynomial time [1].

For $k = 4$, there is a difference between the optimal solutions of the min-max multiway cut and the min-max connected multiway cut. Consider the example graph on five vertices with edge weights one and two as shown in Figure 1. In any connected multiway cut, the vertex u cannot belong to the component of t_4, and if it belongs to a component of any other terminal, the boundary size of this component is seven. However, the cost of the multiway cut $\{t_1\}, \{t_2\}, \{t_3\}, \{u, t_4\}$ is only six.

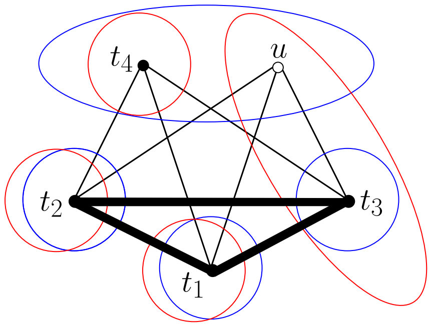

Figure 1 Min-max multiway cut vs. Min-max connected multiway cut. The fat edges are of weight two, the regular edges are of weight one. The optimal min-max cut is of cost six (the blue ovals) while the optimal min-max connected cut is of size seven (the red ovals).

2.2 NP-hardness

The weighted multiway cut problem was shown to be NP-hard for four terminals by Svitkina and Tardos [24]. The same proof without any modification also proves NP-hardness of the weighted min-max connected multiway cut problem because in their reduction, Svitkina and Tardos use (four) terminals that are connected to every non-terminal. So, any multiway cut in the resulting graph is also a connected multiway cut.

For two terminals, the min-max multiway cut problem reduces to the standard $s-t$ cut problem and can be solved in polynomial time. Furthermore, from a minimum cut, a minimum connected cut can be obtained as shown in Lemma 3.

This leaves the situation for three terminals unclear. We now prove that the weighted min-max connected multiway cut problem is weakly NP-hard for three terminals. Together with Lemma 3 this also shows weak NP-hardness of min-max multiway cut problem with three terminals. Recall that a problem is weakly NP-hard if it is NP-hard when the input weights are written in binary but polynomial time solvable when the input is written in unary.

▶ **Theorem 4.** *The weighted min-max connected multiway cut problem and the weighted min-max multiway cut problem with three terminals are weakly NP-hard.*

Proof. We show NP-hardness for the weighted min-max connected multiway problem. The NP-hardness of the weighted min-max multiway problem follows by Lemma 3.

We give a reduction from the three-way partitioning problem, which is a special case of the multiway number partitioning problem [18] (cf. multiprocessor scheduling problem [10]). In the multiway partitioning problem, we wish to separate n positive integers with total sum kN, into k parts in such a way that each part has sum N. This problem is weakly NP-hard for any fixed k. We use $k = 3$ in our reduction.

Given n positive integers $a_1, \ldots, a_n$ summing to $3N$, we construct a graph G on $n+3$ vertices as follows: for each $i \in [n]$, there is a vertex v_i, and in addition, there are three terminals t_1, t_2, t_3. Each of the n vertices v_i is connected to each of the terminals by an edge of weight $a_i/2$. In any multiway cut, let $S_j \subseteq \{v_1, \ldots, v_n\}$ be the set of non-terminals in the component of t_j. Then, the boundary $\delta(S_j)$ has total weight $\sum_{v_i \in S_j} 2 \cdot a_i/2 + \sum_{v_i \notin S_j} a_i/2 = 3N/2 + \sum_{v_i \in S_j} a_i/2$. Thus, as $\sum_{j=1}^{3} \sum_{v_i \in S_j} a_i = 3N$, the optimum min-max connected multiway cut in G with terminals t_1, t_2, t_3 has size at least $2N$, and it equals $2N$ if and only if there is a three-way partition of the input numbers with each part having sum exactly N. ◀

Note that the graphs used in the proof of Theorem 4 are the complete bipartite graphs $K_{3,n}$, and these graphs have tree-width three. This fact, together with Lemma 3, gives us the following corollary.

▶ **Corollary 5.** *The weighted min-max connected multiway cut and the weighted min-max multiway cut with three terminals are weakly NP-hard on graphs of tree-width at least three.*

If we allow more terminals, then the NP-hardness holds for simpler graphs, namely for graphs of tree-width two.

▶ **Theorem 6.** *The weighted min-max connected multiway cut on bipartite planar graphs of tree-width two is weakly NP-hard.*

Proof. We use a straightforward modification of the NP-hardness proof for the min-max multiway cut on trees by Svitkina and Tardos [24]. We give a reduction from the Partition problem which is weakly NP-hard [15].

Given n positive integers $a_1, \ldots, a_n$ with total sum $2B$, we want to check whether they can be partitioned into two subsets, each summing to B. For each $i \in [n]$, we create a tree τ_i with a non-terminal vertex v_i and six terminals, with edge weights as shown in Figure 2. Finally, we add two terminals t_1, t_2, and connect them with each non-terminal v_i with an edge of weight zero.

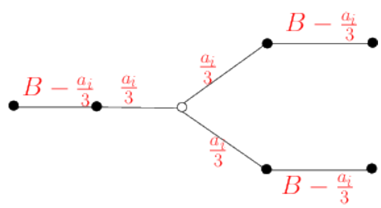

Figure 2 Gadget for NP-hardness for tree-width two

Note that $K_{2,n}$ has tree-width two, and attaching disjoint trees to it does not increase it. Thus, the resulting graph has tree-width two. By construction, it is clearly a bipartite planar graph.

In any connected multiway cut C, if a non-terminal v_i is assigned to any terminal other than t_1, t_2, then $\text{cost}(C) > B$. If every v_i is assigned to t_1 or t_2, then $\text{cost}(C) = \max\left\{\sum_{v_i \text{ assigned to } t_1} a_i, \sum_{v_i \text{ assigned to } t_2} a_i\right\}$. Thus, an optimal min-max connected multiway cut has cost B if and only if there is a desired partition of the given numbers. ◀

3 Trees

Given that the min-max connected multiway cut problem is strongly NP-hard on general graphs with four or more terminals, weakly NP-hard for three terminals on graphs of tree-width three, and weakly NP-hard on graphs of tree-width two, we restrict our attention now to trees.

3.1 Pseudopolynomial Algorithm

We now give a dynamic programming algorithm that solves the weighted min-max connected multiway cut problem on trees in pseudopolynomial time.

▶ **Theorem 7.** *The* weighted *min-max connected multiway cut on trees is solvable in pseudo-polynomial time.*

Proof. We define an auxiliary problem *Restricted connected partition (RCP)*. An instance of RCP consists of a weighted tree $T = (V, E)$ with a root $r \in V$, a set of terminals $\Gamma \subset V$ and an integer k. A feasible solution is a connected multiway cut such that the boundary weight of the component containing the root r, called a *root component*, is exactly k. The objective is to find a feasible solution minimizing the largest boundary weight, where the minimization is done over all components (including the root component); the weight of an optimal feasible solution for an instance (T, Γ, r, k) will be denoted by $\mathsf{RCP}(T, \Gamma, r, k)$; if no feasible solution exists, this value is ∞ by definition. For notational simplicity, we occasionally slightly abuse our notation and use $\mathsf{RCP}(T, \Gamma, r, k)$ for sets Γ that are not subsets of T; the meaning of this notation is then $\mathsf{RCP}(T, \Gamma \cap T, r, k)$.

Further, given in addition an integer c, by $\mathsf{RCP}'(T, \Gamma, r, c)$ we denote the weight of the optimal solution of a slightly modified problem RCP': the boundary weight restriction on the root component is omitted, and the objective is to minimize the largest *adjusted* boundary weight where the adjusted boundary weight of the root component is by c larger than its real weight, and the adjusted boundary weight of every other component is its real weight (i.e., as before).

Assume that we are given an instance of the Min-Max Connected Multiway Cut on trees, that is, a tree $T = (V, E)$ and a set of terminals $\Gamma \subseteq V$. We choose and fix an arbitrary vertex as a root of T; we denote it by r_T in the following text. For every (non-leaf) vertex u of T, we also fix the order of the children of u in an arbitrary way. For any node u of the tree, we denote by T_u the subtree of T consisting of u and all its descendants, and by T_u^i the subtree of T consisting of u, the subtrees of the first i children of u and the edges between u and the first i children; in particular, T_u^0 is the subtree consisting of the single vertex u.

▶ **Observation 8.** *For every non-leaf vertex u of the tree T with children $u_1, \ldots, u_\ell$, and every integers k and $i \in \{1, \ldots, \ell\}$, the following, where $c = w(uu_i)$, holds:*

1. *If $u, u_i \in \Gamma$, then*

$$\mathit{RCP}(T_u^i, \Gamma, u, k) = \max\left\{k, \mathit{RCP}(T_u^{i-1}, \Gamma, u, k - c), \mathit{RCP}'(T_{u_i}, \Gamma, u_i, c)\right\} .$$

2. *If $u \in \Gamma$ and $u_i \notin \Gamma$, then*

$$\mathit{RCP}(T_u^i, \Gamma, u, k) = \min\Bigg\{ \min_{j \in \{0,1,\ldots,k\}} \left\{ \max\left\{k, \mathit{RCP}(T_u^{i-1}, \Gamma, u, j), \mathit{RCP}(T_{u_i}, \Gamma \cup \{u_i\}, u_i, k - j)\right\}, \right. \\ \left. \max\left\{k, \mathit{RCP}(T_u^{i-1}, \Gamma, u, k - c), \mathit{RCP}'(T_{u_i}, \Gamma, u_i, c)\right\} \right\} .$$

3. If $u \notin \Gamma$ and $u_i \in \Gamma$, then

$$
\begin{aligned}
&\mathit{RCP}(T_u^i, \Gamma, u, k) = \\
&\quad = \min \Big\{ \min_{j \in \{1,\dots,k\}} \big\{ \max \{k, \mathit{RCP}(T_u^{i-1}, \Gamma \cup \{u\}, u, j), \mathit{RCP}(T_{u_i}, \Gamma, u_i, k-j)\}, \\
&\qquad \max \{k, \mathit{RCP}(T_u^{i-1}, \Gamma \cup \{u\}, u, k-c), \mathit{RCP}'(T_{u_i}, \Gamma, u_i, c)\} \Big\} .
\end{aligned}
$$

4. If $u, u_i \notin \Gamma$, then

$$
\begin{aligned}
&\mathit{RCP}(T_u^i, \Gamma, u, k) = \qquad (1)\\
&\quad = \min \Big\{ \min_{j \in \{1,\dots,k-1\}} \big\{ \max \{k, \mathit{RCP}(T_u^{i-1}, \Gamma \cup \{u\}, u, j), \mathit{RCP}(T_{u_i}, \Gamma, u_i, k-j)\}, \\
&\qquad \min_{j \in \{1,\dots,k-1\}} \big\{ \max \{k, \mathit{RCP}(T_u^{i-1}, \Gamma, u, j), \mathit{RCP}(T_{u_i}, \Gamma \cup \{u_i\}, u_i, k-j)\}, \\
&\qquad \max \{k, \mathit{RCP}(T_u^{i-1}, \Gamma, u, k-c), \mathit{RCP}'(T_{u_i}, \Gamma, u_i, c)\} \Big\} .
\end{aligned}
$$

Further, for every leaf u of T, and every integer c, $\mathit{RCP}'(T_u, \{u\}, u, c) = c$, and $\mathit{RCP}'(T_u, \emptyset, u, c) = \infty$.

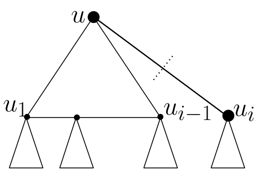

Figure 3 Case 1

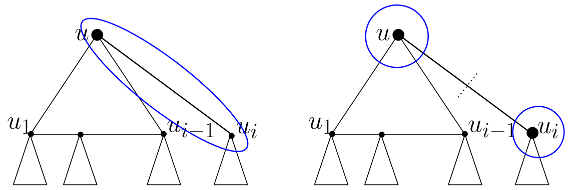

Figure 4 Case 2

Proof. Case 1 (cf. Fig. 3). Assume first that a feasible solution exists for the instance (T_u^i, Γ, u, k) and let $h \geqslant k$ denote the cost of an optimal solution. As both u and u_i are terminals, the edge between them has to be cut in every feasible solution of our instance. Therefore, every feasible solution is a union of partitions of the two trees T_u^{i-1} and T_{u_i}. Note that the corresponding partition of T_u^{i-1} is a feasible solution for the instance $(T_u^{i-1}, \Gamma \cap T_u^{i-1}, u, k-c)$ of cost at most h, and, similarly, the corresponding partition of T_{u_i} is a feasible solution for the instance $(T_{u_1}, \Gamma \cap T_u^{i-1}, u_i, c)$ of the modified problem RCP′, of cost at most h. As k is always a lower bound on RCP(T_u^i, Γ, u, k), we conclude that the left-hand side is at least as large as the right-hand side.

Now assume that a feasible solution exists for the instance $(T_u^{i-1}, \Gamma, u, k-1)$ of RCP and also for the instance $(T_{u_i}, \Gamma, u_i, c)$ of the modified problem RCP′, and let h_1 and h_2, resp., denote their costs. Then the union of these two partitions yields a feasible solution for the instance (T_u^i, Γ, u, k) of cost at most $\max\{k, h_1, h_2\}$. This proves that the right-hand side is at least as large as the left hand side, proving equality and completing the proof of Case 1.

Case 2. Assume, similarly as in Case 1, that a feasible solution exists for the instance (T_u^i, Γ, u, k) and let $\mathcal{S}$ denote an optimal solution and $h \geqslant k$ its cost. There are two subcases to consider. Either the vertices u and u_i belong to the same connected component in $\mathcal{S}$, or to different connected components.

In the first subcase, for some $j \in \{0, 1, \dots, k\}$, the solution $\mathcal{S}$ induces, in a similar way that we described in Case 1, a feasible solution for the instance $(T_u^{i-1}, \Gamma, u, j)$ of RCP of cost at most h, and also a feasible solution for the instance $(T_{u_i}, \Gamma \cup \{u_i\}, u_i, k-j)$ of RCP of cost at most h; on the other hand, for every $j \in \{0, \dots, k\}$, feasible solutions for RCP instances $(T_u^{i-1}, \Gamma, u, j)$ and $(T_{u_i}, \Gamma \cup \{u_i\}, u_i, k-j)$ of cost at most h' can be combined into a feasible solution for the RCP instance (T_u^i, Γ, u, k) of cost at most $\max\{h', k\}$.

In the second subcase, for the same reasons as in Case 1, we have $\mathsf{RCP}(T_u^i, \Gamma, u, k) = \max\left\{k, \mathsf{RCP}(T_u^{i-1}, \Gamma, u, k-c), \mathsf{RCP}'(T_{u_i}, \Gamma, u_i, c)\right\}$.

Case 3. The situation is symmetrical to Case 2 and therefore the same reasoning yields the claim.

Case 4. Assume, similarly as in Case 2, that a feasible solution exists for the instance (T_u^i, Γ, u, k) and let $\mathcal{S}$ denote an optimal solution and $h \geqslant k$ its cost. Again, there are a few subcases to consider. Either the vertices u and u_i belong to the same connected component in $\mathcal{S}$ and the terminal of this component is in T_{u_i}, or u and v belong to the same connected component and its terminal is in T_u^{i-1}, or u and u_i belong to different connected components. The first subcase is dealt with by the first line of the right-hand side of the formula (1), the second subcase by the second line, and the third subcase by the third line, by the same arguments as in the previous cases.

Regarding the last statement in the observation, if the set of terminals is of size one, then the optimal solution for the given RCP' instance consists of an empty set of edges, and the adjusted weight of this set is c by definition. If the set of terminals is empty, then there is no feasible solution and, by definition, $\mathsf{RCP}'(T_u, \emptyset, u, c) = \infty$. ◀

The Observation yields a dynamic programming algorithm that finds in time polynomial in $N = \sum_{e \in E} w(e)$ the smallest value $k \in \{0, 1, \dots, N\}$ for which $\mathsf{RCP}(T, \Gamma, r_t, k)$ is finite; this is the cost of the optimal weighted min-max connected cut. ◀

For unweighted trees, as $N = n - 1$, the algorithm runs in polynomial time in the size of the input. Thus, we get the following corollary.

▶ **Corollary 9.** *The* unweighted *min-max connected multiway cut on trees is solvable in polynomial time.*

3.2 FPTAS

For the weighted min-max connected multiway cut problem on trees we neither know of an NP-hardness proof, nor do we have a polynomial time algorithm. However, we show now that the weighted min-max connected multiway cut problem can be approximated efficiently on trees.

▶ **Theorem 10.** *For every $\varepsilon > 0$, there is an algorithm for the weighted min-max connected multiway cut on trees that runs in time polynomial in both the instance size and $1/\varepsilon$ and finds a $(1+\varepsilon)$-approximation solution.*

Proof. The algorithm uses the standard rounding and scaling approach.

Given $\varepsilon > 0$, a tree $T = (V, E)$ with a weight function $w : E \to \mathbb{N}$ and a set of terminals $S \subseteq V$, we start by checking whether there exists an optimal min-max connected cut of value at most n/ε; this can be done in time polynomial in n and $1/\varepsilon$ by the algorithm of Theorem 7. If there is such a cut, we have found an optimal solution in time polynomial in n and $1/\varepsilon$ and we stop.

From now on, we assume that $OPT > n/\varepsilon$. Let $W = \max_{e \in E} w(e)$ and $d = \frac{\varepsilon W}{n}$. For each edge $e \in E$, we define a new edge weight $w'(e) = \lceil \frac{w(e)}{d} \rceil$. Note that the new edge weights are integers in the range from 0 to $\lceil \frac{n}{\varepsilon} \rceil$. Then again the algorithm from Theorem 7 can be used to find, in time polynomial in the instance size n and $1/\varepsilon$, an optimal solution for the modified instance; this solution will be presented as an output for the original weighted instance of the problem.

Consider now the approximation ratio. Let $F \subseteq E$ be the optimal min-max connected cut and $F_{\max} \subseteq F$ be the largest (w.r.t. to the weights w) boundary of a component in the optimal solution. Similarly, let $A \subseteq E$ be the connected cut constructed by the algorithm of Theorem 7, and $A_{\max} \subseteq A$ the largest boundary of a component in this solution. Then, as for each edge $e \in E$, $w(e) \leq d \cdot w'(e) \leq w(e) + 1$, we have

$$\sum_{e \in A_{\max}} w(e) \leq d \cdot \sum_{e \in A_{\max}} w'(e) \leq \sum_{e \in F_{\max}} w'(e) \leq \sum_{e \in F_{\max}} w(e) + |F_{\max}| \leq OPT + n \leq (1+\varepsilon) \cdot OPT \, .$$

◀

3.3 Parametrized Complexity

Now we show that for a small number of terminals the min-max connected multiway cut problem on trees can be solved efficiently.

▶ **Theorem 11.** *The weighted min-max connected multiway cut on trees is fixed-parameter tractable with respect to the number of terminals.*

Proof. The essence of the proof is the kernelization method. Given an instance of the problem, that is, a tree $T = (V, E)$ and a set of terminals $\Gamma \subseteq V$, we first choose any of the terminals as the root of the tree. Without loss of generality, we can assume that every leaf of the tree T is a terminal; otherwise, we remove from T all vertices (and the adjacent edges) that do not have any terminal as their descendant.

Assuming that all leaves of $T = (V, E)$ are terminals, let W be the subset consisting of all terminals and all vertices of degree at least three. We say that two vertices $u, v \in W$ are *neighboring* if the unique path between u and v in T does not contain any vertex from W as an internal vertex. For any two neighboring vertices u and v, we replace the path between u and v by a single edge, and the weight of the edge is set to the minimum weight of the edges on the original $u - v$ path in T. Let $T' = (V', E')$ denote the resulting tree.

We observe two things:

- A connected multiway cut for T can be easily transformed into a connected multiway cut for T' of the same value, and vice versa.
- The tree T' has at most $2|\Gamma| - 1$ vertices and at most $2|\Gamma| - 2$ edges.

Thus, we can solve the weighted min-max connected multiway cut problem in T' optimally in time $O(2^{2|\Gamma|})$, and from the optimal solution obtain an optimal solution for T. ◀

3.4 Extension Complexity

Even though we do not know whether the weighted min-max connected multiway cut problem is NP-hard on trees, now we give some hardness results that are closely related. In particular, in this section we show that the problem cannot be formulated as a polynomial size linear program using a natural encoding of the solutions. More precisely, we now show that a natural polytope associated with the problem has superpolynomial extension complexity.

Let $P \subset \mathbb{R}^d$ be a polytope. A polytope $Q \subset \mathbb{R}^{d+r}$ is called an extended formulation if P is a projection of Q. That is,

$$P = \left\{ x \in \mathbb{R}^d \;\middle|\; \exists y \in \mathbb{R}^d, \begin{pmatrix} x \\ y \end{pmatrix} \in Q \right\} .$$

The extension complexity of a polytope P – denoted by $\text{xc}(P)$ – is the smallest number of inequalities needed to describe any extended formulation of P. A superpolynomial lower bound on the extension complexity of a polytope implies that one cannot obtain a polynomial size linear program to optimize a linear function over P using auxiliary variables [6, 9]. This rules out a wide range of algorithms even though it does not completely rule out polynomial algorithms [21].

Let $G = (V, E)$ be a graph, $w : E \to \mathbb{R}$ be a weight function on the edges, and $\Gamma \subseteq V$ be a set of terminals. We denote by $\mathcal{C}$ the set of all connected multiway cuts in G for T, and for a connected cut $C \in \mathcal{C}$ we denote its cost, that is, the maximum over the sizes of the boundaries of its connected components, by $\nu(C)$, and by $\chi^C \in \{0,1\}^{|E|}$ we denote the characteristic vector of C . We define the Min-Max Connected cut polytope $P(G, \Gamma, w)$ to be the convex hull of $\{(\chi^C, \nu(C) \in \mathbb{R}^{|E|+1} \mid C \in \mathcal{C}\}$. In other words, $P(G, \Gamma, w)$ is the convex hull of the characteristic vectors of all connected cuts in G extended with a coordinate specifying its value[1].

We will prove a superpolynomial lower bound on the extension complexity of $P(G, \Gamma, w)$. To do this, we will use a superpolynomial lower bound on the extension complexity of the convex hull of feasible solutions of the partition problem.

Let $S = \{a_1, \ldots, a_n\}$ be a set of positive integers, and let $B \in \mathbb{N}$ such that $\sum_{i\in[n]} a_i = 2B$. The partition polytope $\mathrm{PARTITION(S, B)}$ is the convex hull of the characteristic vectors of subsets $I \subset [n]$ such that $\sum_{i\in I} a_i = B$.

▶ **Theorem 12.** *For infinitely many integers $n \in \mathbb{N}$ there exists a set $S = \{a_1, \ldots, a_n\}$ of positive integers, $B \in \mathbb{N}$ with $\sum_{i\in[n]} a_i = 2B$ such that the extension complexity of* $\mathrm{PARTITION(S, B)}$ *is superpolynomial in n.*

Proof. Let $\hat{S} = \{a_1, \ldots, a_n\}$ be a set of positive integers, and let $\hat{B} \in \mathbb{N}$. The SUBSET-SUM problem asks whether there exists a subset $I \subseteq [n]$ such that $\sum_{i\in I} a_i = \hat{B}$. We will call the convex hull of the characteristic vectors of the feasible solutions of the instance $(\hat{S}, \hat{B})$ the polytope $\mathrm{SUBSETSUM(\hat{S}, \hat{B})}$. It is known [3] that there are infinitely many values of $n \in \mathbb{N}$ so that there exists an instance $(\hat{S}, \hat{B})$ of SUBSET-SUM problem with $|\hat{S}| = n$ such that $\text{xc}(\text{SUBSETSUM}(\hat{S}, \hat{B}))$ is superpolynomial in n.

Given an instance $(\hat{S}, \hat{B})$ of the SUBSET-SUM problem we use a standard reduction [10] to PARTITION problem to create an instance (S, B) of the partition problem such that $\text{xc}(\text{SUBSETSUM}(\hat{S}, \hat{B})) \leqslant \text{xc}(\text{PARTITION}(S, B))$, thus proving the theorem.

Given $(\hat{S}, \hat{B})$ define $S = \hat{S} \cup \{a_{n+1}, a_{n+2}\}$ where $a_{n+1} = \sum_{i=1}^{n} a_i$ and $a_{n+2} = 2\hat{B}$, and $B = \hat{B} + \sum_{i=1}^{n} a_i$. The partition instance (S, B) has a solution if and only if the subset sum instance $(\hat{S}, \hat{B})$ has a solution. If $S' \subseteq [n]$ is a feasible solution of the subset sum instance, then $S' \cup \{n + 1\}$ is a feasible solution of the partition instance. Furthermore, if $S' \subseteq [n + 2]$ is a feasible solution of the partition instance such that $n + 1 \in S'$ then $S' \setminus \{n+1\}$ is a feasible solution of the subset sum instance. Therefore, $\text{SUBSETSUM}(\hat{S}, \hat{B})$ is a face of $\text{PARTITION}(S, B)$ obtained using the valid inequality $x_{n+1} \leqslant 1$. Therefore, $\text{xc}(\text{SUBSETSUM}(\hat{S}, \hat{B})) \leqslant \text{xc}(\text{PARTITION}(S, B))$. ◀

▶ **Theorem 13.** *For infinitely many integers $n \in \mathbb{N}$ there exists a tree T on n vertices, a set of terminals Γ, and a weight function w on the edges such that the extension complexity of $P(T, \Gamma, w)$ is superpolynomial in n.*

---

[1] This extra-coordinate as well as definition of P to include the weight function, seems unavoidable as the min-max objective function is not linear.

Proof. Let $S = \{a_1, \ldots, a_n\}$ be a set of positive integers, and let $B \in \mathbb{N}$ be such that $\sum_{i\in[n]} s_i = 2B$. So (S, B) is an instance of the partition problem. That is, we are interested in subsets $S' \subset [n]$ such that $\sum_{i\in S'} a_i = B$. We will show that PARTITION(S, B) is a face of $P(G, \Gamma, w)$ for a suitable graph G, a set of terminals Γ, and weights on its edges $w : E(G) \to \mathbb{R}$. In fact, the graph that we build is a tree. Thus, for each instance (S, B) of the partition problem for which PARTION(S, B) has superpolynomial extension complexity, we get a polytope for the min-max connected multiway cut that has superpolynomial extension complexity.

We build a weighted tree as follows. We have a special vertex called the root terminal in T. For each element $a_i \in S$ we attach a five-vertex subtree consisting of four terminals and one non-terminal, as shown in Figure 5. We label the edges $e_i^0, e_i^1, e_i^2, e_i^3$, and e_i^4 as shown. The

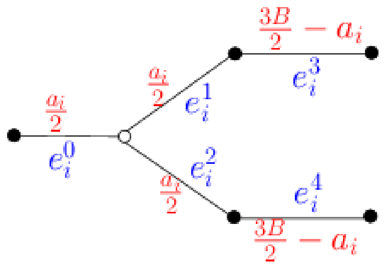

Figure 5 Superpolynomial extension complexity gadget; solid circles are terminals, hollow circles are non-terminals

weights of the edges are as follows: $w(e_i^0) = w(e_i^1) = w(e_i^2) = a_i/2, w(e_i^3) = w(e_i^4) = 3B/2 - a_i$.

Each connected cut in T has a value at least $3B/2$, so $z = 3B/2$ is a face of $P(T, \Gamma, w)$ where z is the last coordinate. Let us call this face F. For any connected cut C, consider the subset $S(C) = \{i \mid e_i^0 \notin C\}$. For the cut C we see that $\sum_{i=1}^n (w(e_i^1)x_{e_i^1} + w(e_i^2)x_{e_i^2}) = B + w(S(C))/2$ which also equals the size of the boundary of the root terminal which for any vertex in F is at most $3B/2$. Therefore, consider the face G of F given by the hyperplane $\sum_{i=1}^n (w(e_i^1)x_{e_i^1} + w(e_i^2)x_{e_i^2}) = 3B/2$. G contains exactly those connected cuts in T that correspond to subsets with sum exactly equal to B. The projection map given by $y_i = 1 - x_i^0$, $\forall i \in [n]$, maps G to the partition polytope of instance (S, B). Therefore PARTITION(S, B) is projection of a face of $P(T, \Gamma, w)$ and so by Theorem 12 we have for infinitely many n a tree and weights so that the extension complexity of the corresponding min-max connected multiway cut polytope superpolynomial in n. ◀

3.5 NP-hardness: Weighted Exact Min-Max Cut

Finally, we show that finding a connected multiway cut of specified size is NP-hard for trees.

▶ **Theorem 14.** *Given a weighted tree and $B \in \mathbb{N}$ it is NP-hard to decide whether there is a connected cut of cost exactly B.*

Proof. Given an instance of the partition problem (S, B), we consider the tree used in the proof of Theorem 13. By adding an additional terminal attached to the root terminal by an edge with weight $2B$, we can ensure that the terminal with the maximum boundary size is the root terminal. Furthermore, for any connected cut C and the corresponding subset $S(C)$, the size of the boundary of the root terminal is exactly $3B + w(S(C))/2$. Thus, one can check whether the partition instance is a yes instance or not by checking whether there is a connected sum with value $7B/2$. ◀

4 Concluding Remarks

We know that the min-max connected multiway cut problem is strongly NP-hard for four terminals, weakly NP-hard for three terminals on graphs of tree-width three, and weakly NP-hard on graphs of tree-width two. This naturally leads to the following problem.

▶ **Problem 1.** *Show that the min-max connected multiway cut problem is strongly NP-hard on graphs of tree-width two or three, or alternatively give a pseudopolynomial algorithm for the weighted version of the problem.*

For trees we have shown that the problem can be solved in pseudopolynomial time, yet we do not have a proof that the problem is weakly NP-hard. This leads to the following problem.

▶ **Problem 2.** *Show that the min-max connected multiway cut problem is weakly NP-hard on trees, or alternatively give a polynomial time algorithm for the weighted version of the problem.*

References

1 Ravindra K. Ahuja, Thomas L. Magnanti, and James B. Orlin. *Network flows - theory, algorithms and applications.* Prentice Hall, 1993.

2 Sunny Atalig, Marek Chrobak, Christoph Dürr, Petr Kolman, Huong Luu, Jiří Sgall, and Gregory Zhu. Two complexity results on spanning-tree congestion problems. *CoRR, arXiv 2601.10881. To appear at LATIN*, 2026. URL: `https://arxiv.org/abs/2601.10881`.

3 David Avis and Hans Raj Tiwary. On the extension complexity of combinatorial polytopes. *Math. Program.*, 153(1):95–115, 2015. `doi:10.1007/S10107-014-0764-2`.

4 Nikhil Bansal, Uriel Feige, Robert Krauthgamer, Konstantin Makarychev, Viswanath Nagarajan, Joseph Seffi Naor, and Roy Schwartz. Min-max graph partitioning and small set expansion. *SIAM Journal on Computing*, 43(2):872–904, 2014.

5 Rodolfo Carvajal, Miguel Constantino, Marcos Goycoolea, Juan Vielma, and Andres Weintraub. Imposing connectivity constraints in forest planning models. *Operations Research*, 61:824–836, 08 2013. `doi:10.2307/23481799`.

6 Michele Conforti, Gérard Cornuéjols, and Giacomo Zambelli. Extended formulations in combinatorial optimization. *Ann. Oper. Res.*, 204(1):97–143, 2013. `doi:10.1007/S10479-012-1269-0`.

7 Daniel Delling, Andrew V. Goldberg, Thomas Pajor, and Renato F. Werneck. Customizable route planning. In Panos M. Pardalos and Steffen Rebennack, editors, *Experimental Algorithms*, page 376–387. Springer Berlin Heidelberg, 2011. `doi:10.1007/978-3-642-20662-7_32`.

8 Gabriel L. Duarte, Hiroshi Eto, Tesshu Hanaka, Yasuaki Kobayashi, Yusuke Kobayashi, Daniel Lokshtanov, Lehilton L. C. Pedrosa, Rafael C. S. Schouery, and Uéverton S. Souza. Computing the largest bond and the maximum connected cut of a graph. *Algorithmica*, 83(5):1421–1458, 2021. `doi:10.1007/S00453-020-00789-1`.

9 Samuel Fiorini, Serge Massar, Sebastian Pokutta, Hans Raj Tiwary, and Ronald de Wolf. Exponential lower bounds for polytopes in combinatorial optimization. *J. ACM*, 62(2):17:1–17:23, 2015. `doi:10.1145/2716307`.

10 M. R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman, 1979.

11 Naveen Garg. *Multicommodity Flows and Approximation Algorithms.* PhD thesis, Indian Institute of Technology, Delhi, 1994.

12 Veronika Grimm, Thomas Kleinert, Frauke Liers, Martin Schmidt, and Gregor Zöttl. Optimal price zones of electricity markets: a mixed-integer multilevel model and global solution approaches. *Optim. Methods Softw.*, 34(2):406–436, 2019. `doi:10.1080/10556788.2017.1401069`.

13 MohammadTaghi Hajiaghayi, Guy Kortsarz, Robert MacDavid, Manish Purohit, and Kanthi K. Sarpatwar. Approximation algorithms for connected maximum cut and related problems. *Theor. Comput. Sci.*, 814:74–85, 2020.

14 Ibrahim Abou Hamad, Per Arne Rikvold, and Svetlana V. Poroseva. Floridian high-voltage power-grid network partitioning and cluster optimization using simulated annealing. In *Proc. of the 24th Workshop on Computer Simulation Studies in Condensed Matter Physics (CSP2011)*, volume 15, page 2–6, 2011. URL: https://www.sciencedirect.com/science/article/pii/S1875389211003221, doi:10.1016/j.phpro.2011.05.051.

15 Richard M. Karp. Reducibility among combinatorial problems. In *Proceedings of a Symposium on the Complexity of Computer Computations*, The IBM Research Symposia Series, pages 85–103. Plenum Press, New York, 1972. doi:10.1007/978-1-4684-2001-2_9.

16 Petr Kolman. Approximation of Spanning Tree Congestion Using Hereditary Bisection. In *42nd International Symposium on Theoretical Aspects of Computer Science (STACS)*, volume 327 of *LIPIcs*, pages 63:1–63:6, Dagstuhl, Germany, 2025. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.STACS.2025.63.

17 Petr Kolman and Hans Raj Tiwary. Bond polytope under vertex- and edge-sums, 2026. URL: https://arxiv.org/abs/2601.11119, arXiv:2601.11119.

18 Richard E. Korf. Multi-way number partitioning. In *Proc. of the 21st International Joint Conference on Artificial Intelligence (IJCAI)*, pages 538–543, 2009. URL: http://ijcai.org/Proceedings/09/Papers/096.pdf.

19 Mikhail I. Ostrovskii. Minimal congestion trees. *Discrete Mathematics*, 285(1):219–226, 2004. doi:10.1016/j.disc.2004.02.009.

20 Yota Otachi. A survey on spanning tree congestion. In *Treewidth, Kernels, and Algorithms: Essays Dedicated to Hans L. Bodlaender on the Occasion of His 60th Birthday*, volume 12160 of *Lecture Notes in Computer Science*, pages 165–172, 2020. doi:10.1007/978-3-030-42071-0_12.

21 Thomas Rothvoss. The matching polytope has exponential extension complexity. *J. ACM*, 64(6):41:1–41:19, 2017. doi:10.1145/3127497.

22 Baruch Schieber and Soroush Vahidi. Approximating connected maximum cuts via local search. In *Proc. of 31st Annual European Symposium on Algorithms (ESA)*, volume 274 of *LIPIcs*, pages 93:1–93:17, 2023.

23 David B. Shmoys. Cut problems and their application to divide-and-conquer. In D. S. Hochbaum, editor, *Approximation Algorithms for NP-hard Problems*, pages 192–235. PWS Publishing Company, 1997.

24 Zoya Svitkina and Éva Tardos. Min-max multiway cut. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX)*, pages 207–218. Springer Berlin Heidelberg, 2004.

25 Sara Vicente, Vladimir Kolmogorov, and Carsten Rother. Graph cut based image segmentation with connectivity priors. In *Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008. doi:10.1109/CVPR.2008.4587440.