Algorithmica (2019) 81:917–966
https://doi.org/10.1007/s00453-017-0319-z

Edge Bipartization Faster than 2^k

Marcin Pilipczuk[1] · **Michał Pilipczuk**[1] · **Marcin Wrochna**[1]

Received: 27 October 2016 / Accepted: 25 April 2017 / Published online: 12 May 2017

Abstract In the Edge Bipartization problem one is given an undirected graph G and an integer k, and the question is whether k edges can be deleted from G so that it becomes bipartite. Guo et al. (J Comput Syst Sci 72(8):1386–1396, 2006) proposed an algorithm solving this problem in time $\mathcal{O}(2^k \cdot m^2)$; today, this algorithm is a textbook example of an application of the iterative compression technique. Despite extensive progress in the understanding of the parameterized complexity of graph separation problems in the recent years, no significant improvement upon this result has been yet reported. We present an algorithm for Edge Bipartization that works in time $\mathcal{O}(1.977^k \cdot nm)$, which is the first algorithm with the running time dependence on the parameter better than 2^k. To this end, we combine the general iterative compression strategy of Guo et al. (2006), the technique proposed by Wahlström (in: Proceedings of SODA'14, SIAM, 2014) of using a polynomial-time solvable relaxation in the form of a Valued Constraint Satisfaction Problem to guide a bounded-depth branching algorithm, and an involved Measure&Conquer analysis of the recursion tree.

1 Introduction

The Edge Bipartization problem asks, for a given graph G and integer k, whether one can turn G into a bipartite graph using at most k edge deletions. Together with its

✉ Marcin Pilipczuk
malcin@mimuw.edu.pl

Michał Pilipczuk
michal.pilipczuk@mimuw.edu.pl

Marcin Wrochna
m.wrochna@mimuw.edu.pl

[1] University of Warsaw, Warsaw, Poland

close relative Odd Cycle Transversal (OCT), where one deletes vertices instead of edges, Edge Bipartization was one of the first problems shown to admit a fixed-parameter (FPT) algorithm using the technique of *iterative compression*. In a breakthrough paper [28] that introduces this methodology, Reed et al. showed how to solve OCT in time $\mathcal{O}(3^k \cdot kmn)$.[1] In fact, this was the first FPT algorithm for OCT. Following this, Guo et al. [14] applied iterative compression to show fixed-parameter tractability of several closely related problems. Among other results, they designed an algorithm for Edge Bipartization with running time $\mathcal{O}(2^k \cdot m^2)$. Today, both the algorithms of Reed et al. and of Guo et al. are textbook examples of the iterative compression technique [5,11].

Iterative compression is in fact a simple idea that boils down to an algorithmic usage of induction. In case of Edge Bipartization, we introduce edges of G one by one, and during this process we would like to maintain a solution F to the problem, i.e., $F \subseteq E(G)$ is such that $|F| \leq k$ and $G - F$ is bipartite. When the next edge e is introduced to the graph, we observe that $F \cup \{e\}$ is a solution of size at most $k + 1$, that is, at most one too large. Then the task reduces to solving Edge Bipartization Compression: given a graph G and a solution that exceeds the budget by at most one, we are asked to find a solution that fits into the budget.

Surprisingly, this simple idea leads to great algorithmic gains, as it reduces the matter to a cut problem. Guo et al. [14] showed that a simple manipulation of the instance reduces Edge Bipartization Compression to the following problem that we call Terminal Separation: We are given an undirected graph G with a set $\mathcal{T}$ of $k+1$ disjoint pairs of terminals, where each terminal is of degree 1 in G. The question is whether one can color one terminal of every pair white and the second black in such a way that the minimum edge cut between white and black terminals is at most k. Thus, the algorithm of Guo et al. [14] boils down to trying all the 2^{k+1} colorings of terminals and solving a minimum edge cut problem. For OCT, we similarly have a too large solution $X \subseteq V(G)$ of size $k + 1$, and we are looking for a partition of X into (L, R, Z), where the size of the minimum vertex cut between L and R in $G - Z$ is at most $k - |Z|$. Thus it suffices to solve 3^{k+1} instances of a flow problem.

The search for FPT algorithms for cut problems has been one of the leading directions in parameterized complexity in the recent years. Among these, Odd Cycle Transversal and Edge Bipartization play a central role; see for instance [14,16,18–20,22,25,28] and references therein. Of particular importance is the work of Kratsch and Wahlström [22], who gave the first (randomized) polynomial kernelization algorithms for Odd Cycle Transversal and Edge Bipartization. The main idea is to encode the cut problems that arise when applying iterative compression into a matroid with a representation that takes small space. The result of Kratsch and Wahlström sparked a line of further work on applying matroid methods in parameterized complexity.

Another thriving area in parameterized complexity is the *optimality program*, probably best defined by Marx in [27]. The goal of this program is to systematically investigate the optimum complexity of algorithms for parameterized problem by prov-

[1] Even though Reed et al. [28] state their running time as $\mathcal{O}(4^k \cdot kmn)$, it is not hard to adjust the analysis to show that the algorithm in fact works in time $\mathcal{O}(3^k \cdot kmn)$; see e.g. [16,26].

ing possibly tight lower and upper bounds. For the lower bounds methodology, the standard complexity assumptions used are the *Exponential Time Hypothesis* (*ETH*) and the *Strong Exponential Time Hypothesis* (*SETH*). In the recent years, the optimality program has achieved a number of successes. For instance, under the assumption of SETH, we now know the precise bases of exponents for many classical problems parameterized by treewidth [7,8,23]. To explain the complexity of fundamental parameterized problems for which natural algorithms are based on dynamic programming on subsets, Cygan et al. [3] introduced a new hypothesis resembling SETH, called the *Set Cover Conjecture* (*SeCoCo*). See [24,27] for more examples.

For our techniques, the most important is the line of work of Guillemot [13], Cygan et al. [9], Lokshtanov et al. [25], and Wahlström [30] that developed a technique for designing parameterized algorithm for cut problems called *LP-guided branching*. The idea is to use the optimum solution to the linear programming relaxation of the considered problem in order to measure progress. Namely, during the construction of a candidate solution by means of a backtracking process, the algorithm achieves progress not only when the budget for the size of the solution decreases (as is usual in branching algorithms), but also when the lower bound on the optimum solution increases. Using this concept, Cygan et al. [9] showed a $2^k n^{\mathcal{O}(1)}$-time algorithm for Node Multiway Cut. Lokshtanov et al. [25] further refined this technique and applied it to improve the running times of algorithms for several important cut problems. In particular, they obtained a $2.3146^k n^{\mathcal{O}(1)}$-time algorithm for Odd Cycle Transversal, which was the first improvement upon the classic $\mathcal{O}(3^k \cdot kmn)$-time algorithm of Reed et al. [28]. From the point of view of the optimality program, this showed that the base 3 of the exponent was not the final answer for Odd Cycle Transversal.

In works [9,25] it was essential that the considered linear programming relaxation is half-integral, which restricts the applicability of the technique. Recently, Wahlström [30] proposed to use stronger relaxations in the form of certain polynomial-time solvable Valued Constraint Satisfaction Problems (VCSPs). Using this idea, he was able to show efficient FPT algorithms for the node and edge deletion variants of Unique Label Cover, for which natural LP relaxations are not half-integral.

Despite substantial progress on the node deletion variant, for Edge Bipartization there has been no improvement since the classic algorithm of Guo et al. [14] that runs in time $\mathcal{O}(2^k \cdot m^2)$. The main technical contribution of Lokshtanov et al. [25] is a $2.3146^k n^{\mathcal{O}(1)}$-time algorithm for Vertex Cover parameterized by the excess above the value of the LP relaxation (VC-above-LP); the algorithm for OCT is a corollary of this result due to folklore reductions from OCT to VC-above-LP via the Almost 2- SAT problem. Thus the algorithm for OCT in fact relies on the LP relaxation for the Vertex Cover problem, which has very strong combinatorial properties; in particular, it is half-integral. No such simple and at the same time strong relaxation is available for Edge Bipartization. The natural question stemming from the optimality program, whether the 2^k term for Edge Bipartization could be improved, was asked repeatedly in the parameterized complexity community, for example by Daniel Lokshtanov at WorKer'13 [6], repeated later at [4].

1.1 Our Results and Techniques

In this paper we answer this question in affirmative by proving the following theorem.

Theorem 1.1 EDGE BIPARTIZATION *can be solved in time* $\mathcal{O}(1.977^k \text{ nm})$.

Thus, the 2 in the base of the exponent is not the ultimate answer for EDGE BIPARTIZATION.

To prove Theorem 1.1, we pursue the approach proposed by Guo et al. [14] and use iterative compression to reduce solving EDGE BIPARTIZATION to solving TERMINAL SEPARATION (see Sect. 2 for a formal definition of the latter). This problem has two natural parameters: $|\mathcal{T}|$, the number of terminal pairs, and p, the bound on the size of the cut between white and black terminals. The approach of Guo et al. is to use a simple $\mathcal{O}(2^{|\mathcal{T}|} \cdot pm)$ algorithm that tries all colorings of terminal pairs and computes the size of a minimum cut between the colors.

The observation that is crucial to our approach is that one can express TERMINAL SEPARATION as a very restricted instance of the EDGE UNIQUE LABEL COVER problem. More precisely, in this setting the task is to assign each vertex of G a label from $\{\mathbf{A}, \mathbf{B}\}$. Pairs of $\mathcal{T}$ present hard (of infinite cost) inequality constraints between the labels of terminals involved, while edges of G present soft (of unit cost) equality constraints between the endpoints. The goal is to minimize the cost of the labeling, i.e., the number of soft constraints broken. An application of the results of Wahlström [30] (with the further improvements of Iwata et al. [17] regarding linear dependency on the input size) immediately gives an $\mathcal{O}(4^p \cdot m)$ algorithm for TERMINAL SEPARATION.

Thus, we have in hand two substantially different algorithms for TERMINAL SEPARATION. If we plug in $|\mathcal{T}| = k + 1$ and $p = k$, as is the case in the instance that we obtain from EDGE BIPARTIZATION COMPRESSION, then we obtain running times $\mathcal{O}(2^k \cdot km)$ and $\mathcal{O}(4^k \cdot m)$, respectively. The idea now is that these two algorithms present two complementary approaches to the problem, and we would like to combine them to solve the problem more efficiently. To this end, we need to explain more about the approach of Wahlström [30].

The algorithm of Wahlström [30] is based on measuring the progress by means of the optimum solution to the relaxation of the problem in the form of a Valued CSP instance. In our case, this relaxation has the following form: We assign each vertex a label from $\{\perp, \mathbf{A}, \mathbf{B}\}$, where $\perp$ is an additional marker that should be thought of as *not yet decided*. The hard constraints have zero cost only for labelings $(\mathbf{A}, \mathbf{B})$, $(\mathbf{B}, \mathbf{A})$ and $(\perp, \perp)$, and infinite cost otherwise. The soft constraints have cost 0 for equal labels on the endpoints, 1 for unequal from $\{\mathbf{A}, \mathbf{B}\}$, and $\frac{1}{2}$ when exactly one endpoint is assigned $\perp$. Based on previous results of Kolmogorov et al. [21], Wahlström observed that this relaxation is polynomial-time solvable, and moreover it is *persistent*: whenever the relaxation assigns $\mathbf{A}$ or $\mathbf{B}$ to some vertex, then it is safe to perform the same assignment in the integral problem (i.e., only with the "integral" labels $\mathbf{A}, \mathbf{B}$). The algorithm constructs an integral labeling by means of a backtracking process that fixes the labels of consecutive vertices of the graph. During this process, it maintains an optimum solution to the relaxation that is moreover *maximal*, in the sense that one cannot extend the current labeling by fixing integral labels on some undecided vertices without increasing the cost. This can be done by dint of persistence and polynomial-time solvability: we can check in polynomial time whether a non-trivial extension exists, and then it is safe to fix the labels of vertices that get decided. Thus, when the algorithm considers the next vertex u and branches into two cases, fixing label **A** or **B** on it, the optimum cost of the relaxation increases by at least $\frac{1}{2}$ in each branch. Hence the recursion tree can be pruned at depth $2p$, and we obtain an $4^p n^{\mathcal{O}(1)}$-time algorithm.

Our algorithm for Terminal Separation applies a similar branching strategy, where at each point we maintain some labeling of the vertices with **A**, **B**, and $\bot$ (undecided). Every terminal pair is either already *resolved* [assigned (**A**, **B**) or (**B**, **A**)], or *unresolved* [assigned $(\bot, \bot)$]. Using the insight of Wahlström we can assume that this labeling is maximal. Intuitively, we look at the unresolved pairs from $\mathcal{T}$ and try to identify a pair (s, t) for which branching into labelings (**A**, **B**) and (**B**, **A**) leads to substantial progress. Here, we measure the progress in terms of a potential μ that is a linear combination of three components:

- t, the number of unresolved terminal pairs;
- k, the current budget for the cost of the sought integral solution;
- ν, the difference between k and the cost of the current solution to the relaxation.

These ingredients are taken with weights $\alpha_t = 0.59950$, $\alpha_\nu = 0.29774$, and $\alpha_k = 1 - \alpha_t - \alpha_\nu = 0.10276$. Thus, the largest weight is put on the progress measured in terms of the number of resolved terminal pairs. Indeed, we want to argue that if we can identify a possibility of recursing into two instances, where in each of them at least one new terminal pair gets resolved, but in one of them we resolve two terminal pairs, then we can pursue this branching step.

Therefore, we are left with the following situation: when branching on any terminal pair, only this terminal pair gets resolved in both branches. Then the idea is to find a branching step where the decrease of the auxiliary components of the potential, namely ν and k, is significant enough to ensure the promised running time of the algorithm. Here we apply an extensive combinatorial analysis of the instance to show that finding such a branching step is always possible. In particular, our analysis can end up with a branching not on a terminal pair, but on the label of some other vertex; however, we make sure that in both branches some terminal pair gets eventually resolved. Also, in some cases we localize a part of the input that can be simplified (a *reduction step*), and then the analysis is restarted.

To sum up, we would like to highlight two aspects of our contribution. First, we answer a natural question stemming from the optimality program, showing that 2^k is not the final dependency on the parameter for Edge Bipartization. Second, our algorithm can be seen as a "proof of concept" that the LP-guided branching technique, even in the more abstract variant of Wahlström [30], can be combined with involved Measure&Conquer analysis of the branching tree. Note that in the past Measure&Conquer and related techniques led to rapid progress in the area of moderately-exponential algorithms [12].

We remark that the goal of the current paper is clearly improving 2^k factor, and not optimizing the dependence of the running time on the input size. However, we do estimate it. Using the tools prepared by Iwata, Wahlström, and Yoishida [17], we are able to implement the algorithm so that it runs in time $\mathcal{O}(1.977^k \cdot nm)$. Naively, this seems like an improvement over the algorithm of Guo et al. [14] that had quadratic dependence on m, however this is not the case. We namely use the recent approximation algorithm for Edge Bipartization of Kolay et al. [19] that in time $\mathcal{O}(k^{\mathcal{O}(1)} \cdot m)$ either returns a solution F^{apx} of size at most $\mathcal{O}(k^2)$, or correctly concludes that there is no solution of size k. Then we start iterative compression from $G - F^{\text{apx}}$ and introduce edges of F^{apx} one by one, so we need to solve the Terminal Separation problem only $\mathcal{O}(k^2)$ times. In our case each iteration takes time $\mathcal{O}(1.977^k \cdot nm)$, but for the approach of Guo et al. it would take time $\mathcal{O}(2^k \cdot km)$. Thus, by using the same idea based on [19], the algorithm of Guo et al. can be adjusted to run in time $\mathcal{O}(2^k \cdot k^3 m)$. It is just that the newer algorithm of Kolay et al. [19] was not known at the time of writing [14].

1.2 Organization of the Paper

In Sect. 2 we give background on iterative compression and the VCSP-based tools borrowed from [17,30]. In particular, we introduce formally the Terminal Separation problem and reduce solving Edge Bipartization to it. In Sect. 3 we set up the Measure&Conquer machinery that will be used by our branching algorithm, and we introduce preliminary reductions. In Sect. 4 we prove some auxiliary results on *low excess set*, which is the key technical notion used in our combinatorial analysis. Finally, we present the whole algorithm in Sect. 5. Sect. 6 is devoted to some concluding remarks and open problems.

2 Preliminaries

2.1 Graph Notation

For all standard graph notation, we refer to the textbook of Diestel [10]. For the input instance (G, k) of Edge Bipartization, we denote $n = |V(G)|$ and $m = |E(G)|$. As isolated vertices are irrelevant for the Edge Bipartization problem, we assume that G does not contain any such vertices, and hence $n = \mathcal{O}(m)$.

2.2 Cuts and Submodularity

As edge cuts in a graph are the main topic of this work, let us introduce some convenient notation. In all graphs in this paper we allow multiple edges, but not loops, as they are irrelevant for the problem. For a graph G and two disjoint vertex sets $A, B \subseteq V(G)$, by $E_G(A, B)$ we denote the set of edges with one endpoint in A and the second endpoint in B. If any of the sets A or B is a singleton, say $A = \{a\}$, we write $E_G(a, B)$ instead of $E_G(\{a\}, B)$. We drop the subscript if the graph G is clear from the context.

For a set $A \subseteq V(G)$, we denote $d(A) = |E(A, V(G)\setminus A)|$. It is well known that the $d(\cdot)$ function is submodular, that is, for every $A, B \subseteq V(G)$ it holds that

$$d(A) + d(B) \geq d(A \cap B) + d(A \cup B). \tag{1}$$

In fact, a study of the proof of (1) allows us to state the difference in the inequality.

$$d(A) + d(B) = d(A \cap B) + d(A \cup B) + 2|E(A \setminus B, B \setminus A)|. \tag{2}$$

Since $d(\cdot)$ is symmetric (i.e., $d(A) = d(V(G) \setminus A)$), by applying (1) to A and the complement of B, we obtain a property sometimes called *posimodularity*: for every $A, B \subseteq V(G)$ it holds that

$$d(A) + d(B) \geq d(A \setminus B) + d(B \setminus A). \tag{3}$$

Or, with the error term:

$$d(A) + d(B) = d(A \setminus B) + d(B \setminus A) + 2|E\left(A \cap B, V(G) \setminus (A \cup B)\right)|. \tag{4}$$

2.3 Iterative Compression and the Compression Variant

Let (G, k) be an input Edge Bipartization instance. The opening step of our algorithm is the standard usage of iterative compression. We start by applying the approximation algorithm of [19] that, given (G, k), in time $\mathcal{O}(k^{\mathcal{O}(1)}m)$ either correctly concludes that it is a no-instance, or produces a set $Z \subseteq E(G)$ of size $\mathcal{O}(k^2)$ such that $G - Z$ is bipartite.

Let Z be the obtained set, $|Z| = r = \mathcal{O}(k^2)$ and $Z = \{e_1, e_2, \ldots, e_r\}$. Let $G_i = G - \{e_{i+1}, e_{i+2}, \ldots, e_r\}$ for $0 \leq i \leq r$; note that $G_r = G$ while $G_0 = G - Z$, which is bipartite. Our algorithm, iteratively for $i = 0, 1, \ldots, r$, computes a solution X_i to the instance (G_i, k), or concludes that no such solution exists. Clearly, since G_i is a subgraph of G, if we obtain the latter conclusion for some i, we can report that (G, k) is a no-instance.

For $i = 0$, $G_0 = G - Z$ is bipartite, thus $X_0 = \emptyset$ is a solution. Consider now an instance (G_i, k) for $1 \leq i \leq r$, and assume that a solution X_{i-1} has already been computed. Let $X'_i = X_{i-1} \cup \{e_i\}$. If $|X'_i| \leq k$, we can take $X_i = X'_i$ and continue. Otherwise, we can make use of the structural insight given by the set X'_i and solve the following problem.

Edge Bipartization Compression
Input: A graph G, and integer k, and a set $X' \subseteq E(G)$ of size $k + 1$ such that $G - X'$ is bipartite.
Goal: Compute a set $X \subseteq E(G)$ of size at most k such that $G - X$ is bipartite, or conclude that no such set exists.

If we could efficiently solve an Edge Bipartization Compression instance (G_i, k, X'_i), we can take the output solution as X_i and proceed to the next step of this iteration. Consequently, it suffices to prove the following theorem.

Theorem 2.1 Edge Bipartization Compression *can be solved in time* $\mathcal{O}(c^k nm)$ *for some constant* $c < 1.977$.

2.4 The Terminal Separation Problem

Following the algorithm of [14], we phrase Edge Bipartization Compression as a separation problem.

Consider a graph G with a family $\mathcal{T}$ of pairs of terminals in G. A pair (A, B) with $A, B \subseteq V(G)$ is a *terminal separation* if $A \cap B = \emptyset$ and, for every terminal pair P, either one of the terminals in P belongs to A and the second to B, or $P \subseteq V(G)\setminus(A \cup B)$. A terminal separation (A, B) is *integral* if $A \cup B = V(G)$.[2] A terminal separation (A', B') *extends* (A, B) if $A \subseteq A'$ and $B \subseteq B'$. The *cost* of a terminal separation (A, B) is defined as $c(A, B) = (d(A) + d(B))/2$. Note that if (A, B) is integral, then we have $c(A, B) = d(A) = d(B)$.

We will solve the following separation problem.

Terminal Separation
Input: A graph G with a set of terminal pairs $\mathcal{T}$ such that the pairs are pairwise disjoint and every terminal is of degree at most one in G; a terminal separation (A°, B°); and an integer k.
Goal: Find an integral terminal separation (A, B) extending (A°, B°) of cost at most k, or report that no such separation exists.

Lemma 2.2 *Given an* Edge Bipartization Compression *instance* (G, k, X'), *one can in polynomial time compute an equivalent instance* $(G', \mathcal{T}, (A^\circ, B^\circ), k')$ *of* Terminal Separation, *such that* $|E(G')| = |E(G)| + \mathcal{O}(|X'|)$, $|V(G')| = |V(G)| + \mathcal{O}(|X'|)$, $|\mathcal{T}| = |X'|$, $A^\circ = B^\circ = \emptyset$, *and* $k' = k$.

Proof Let G' be the graph obtained from G by replacing every edge uv in X' with two new vertices s, t and two pendant edges us, vt. Let $\mathcal{T}$ be the set of vertex pairs $\{s, t\}$ created this way, $A^\circ = B^\circ = \emptyset$, $k' = k$. We show this constructed instance is equivalent to the original instance.

If the constructed instance is a yes-instance, let A, B be an integral terminal separation of cost at most k. Take $X = E(A, B)$, and then, for every edge in X incident to a terminal s, replace this edge with uv, where uv is the edge of X' for which the terminal pair containing s was created. We claim X is a solution to the original instance (clearly $|X| \leq |E(A, B)| \leq k$). Indeed, let L', R' be a bipartition of $G - X'$. We show that $(L' \cap A) \cup (R' \cap B)$, $(R' \cap A) \cup (L' \cap B)$ gives a bipartition of $G - X$. Suppose that, to the contrary, there is an edge uv in $G - X$ with both endpoints in $(L' \cap A) \cup (R' \cap B)$ (the case of $(R' \cap A) \cup (L' \cap B)$ being symmetrical). Since all edges with one endpoint in $A \cap V(G)$ and the other in $B \cap V(G)$ were deleted by X, we may assume uv is an edge with both endpoints in $L' \cap A$ (the case of $R' \cap B$ being symmetrical).

Since L' is one side of a bipartition of $G - X'$, uv must be an edge in X'. Let us, vt be the corresponding edges to terminals created by the construction. Since both u and v are in A and exactly one of s, t is in A (as A, B is a terminal separation), one of us, vt must be an edge in $E(A, B)$. Hence uv was deleted by X', a contradiction.

[2] The word *integral* stems from the fact that an integral separation corresponds to a solution to the relaxed Terminal Separation problem that actually does not use the relaxed value $\perp$. In fact, it also corresponds to an integral solution of an LP formulation underlying the algorithmic results of [21].

For the other side, if the original instance is a yes-instance, let X be a set of at most k edges such that $G-X$ has a bipartition L, R. By taking X to be minimal, we can assert that every edge in X has both endpoints on one side of this bipartition. Let A contain all vertices in $(L \cap L') \cup (R \cap R')$, let B contain all vertices of $(L \cap R') \cup (R \cap L')$. For every edge uv in X, add the corresponding terminal vertices s, t to A and B so that s is in the same set as u and t is in the other one. Clearly (A, B) is an integral terminal separation extending $(\emptyset, \emptyset)$, it suffices to show that $|E_{G'}(A, B)| \leq |X|$.

Let $e \in E_{G'}(A, B)$. If neither endpoint of e is a terminal, then let $e = uv$ for $u \in (L \cap L') \cup (R \cap R')$ and $v \in (L \cap R') \cup (R \cap L')$. Since all edges with both endpoints in L' were in X' and hence replaced by edges to terminals in G', it cannot be that both u and v are in L'. Similarly, for R', so let us assume that $u \in L'$ and $v \in R'$ (the other case is symmetrical). Then $u \in (L \cap L')$ and $v \in (L \cap R')$, which means both u and v are on the same side of the (L, R) bipartition of $G - X$ but on different sides of the (L', R') bipartition of $G - X'$. Hence $e \in X \backslash X'$.

Otherwise, assume that some endpoint of e is a terminal. Recall that we have defined A and B in such a way that the edge us for a terminal pair $\{s, t\}$ is never cut. Hence, let $e = vt$, and without loss of generality assume that $v \in (L \cap L') \cup (R \cap R')$. (Here we keep the notation that s, t are two terminal with edges us and vt replacing in the construction an edge uv in X'.) In particular $v \in A$, so $t \in B$. By construction of the separation (A, B), it must be that s is in the same side as u and t is on the other side, so $s, u \in A$. Since u is not a terminal, $u, v \in (L \cap L') \cup (R \cap R')$. Hence $uv \in X' = E_G(L', L') \cup E_G(R', R')$ implies that u and v are on the same side of the (L', R') partition, thus also on the same side of the (L, R) partition and hence $uv \in X \cap X'$. Note also that of the two edges that replace uv in the construction, only e is in $E_{G'}(A, B)$, because $s, u \in A$.

Hence every edge in $E_{G'}(A, B)$ is either an edge in $X \backslash X'$ or an edge to a terminal uniquely corresponding to an edge in $X \cap X'$, which implies $|E_{G'}(A, B)| \leq |X|$, concluding the other side of the proof. □

We say that a terminal pair P is *resolved* in a Terminal Separation instance $(G, \mathcal{T}, (A^\circ, B^\circ), k)$ if $P \subseteq A^\circ \cup B^\circ$, and *unresolved* otherwise (i.e., $P \subseteq V(G) \backslash (A^\circ \cup B^\circ)$). Thus, our goal is to design an efficient branching algorithm for Terminal Separation, with parameters being k, the excess in the cutset $k - c(A^\circ, B^\circ)$, and the number t of unresolved terminal pairs. A precise statement of the result can be found in Sect. 3, where an appropriate progress measure is defined.

2.5 LP Branching

The starting point in designing an algorithm for Terminal Separation using the aforementioned parameters is the generic LP branching framework of Wahlström [30].

Observe that one can phrase an instance of Terminal Separation as a Valued CSP instance, with vertices being variables over the domain $\{\mathbf{A}, \mathbf{B}\}$, edges being soft (unit cost) equality constraints, terminal pairs being hard (infinite or prohibitive cost) inequality constraints, while membership in A° or B° translates to hard unary constraints on vertices of $A^\circ \cup B^\circ$. Observe that this Valued CSP instance is in fact a

Unique Label Cover instance over binary alphabet, with additional hard unary constraints.

In a relaxed instance, we add to the domain the "do not know" value $\perp$, and extend the cost function for an edge (equality) constraint to be 0 for both endpoints valued $\perp$, and $\frac{1}{2}$ for exactly one endpoint valued $\perp$; the hard inequality constraints on terminal pairs additionally allows both terminals to be valued $\perp$. Observe that now feasible solutions $f : V(G) \to \{\perp, \mathbf{A}, \mathbf{B}\}$ to the so-defined instance are in one-to-one correspondence with terminal separations $(A, B) = (f^{-1}(\mathbf{A}), f^{-1}(\mathbf{B}))$, and the cost of f equals exactly $c(f^{-1}(\mathbf{A}), f^{-1}(\mathbf{B}))$. Furthermore, as shown by Wahlström [30], the cost functions in this relaxation are bisubmodular, which implies the following two corollaries.

Theorem 2.3 (persistence [30]) *Let $(G, \mathcal{T}, (A^\circ, B^\circ), k)$ be a* Terminal Separation *instance, and let (A, B) be a terminal separation in G of minimum cost among separations that extend (A°, B°). Then there exists an integral separation (A^*, B^*) that has minimum cost among all separations extending (A°, B°), with the additional property that (A^*, B^*) extends (A, B).*

We say that a terminal separation (A, B) is *maximal* if every other separation extending it has strictly larger cost.

Theorem 2.4 (polynomial-time solvability [17,30]) *Given a* Terminal Separation *instance $(G, \mathcal{T}, (A^\circ, B^\circ), k)$ with $c(A^\circ, B^\circ) \leq k$, one can in $\mathcal{O}(k^{\mathcal{O}(1)}m)$ time find a maximal terminal separation (A, B) in G that has minimum cost among all separations extending (A°, B°).*

From Theorems 2.3 and 2.4 it follows that, while working on a Terminal Separation instance $(G, \mathcal{T}, (A^\circ, B^\circ), k)$, we can always assume that (A°, B°) is a maximal separation: If that is not the case, we can obtain an extending separation (A, B) via Theorem 2.4, and set $(A^\circ, B^\circ) := (A, B)$; the safeness of the last step is guaranteed by Theorem 2.3.

We remark here that, in the course of the algorithm, we will often merge sets of vertices in the processed graph. For a nonempty set $X \subseteq V(G)$, the operation of *merging X into a vertex* replaces X with a new vertex x, and replaces every edge $uv \in E(X, V(G)\setminus X)$, $u \in X$, $v \notin X$, with an edge xv. That is, in this process we do not supress multiple edges while identifying some vertices. However, we do supress loops, as they are irrelevant for the problem. Consequently, we allow the graph G to have multiple edges, but not loops; we remark that both theorems cited in this section work perfectly fine in this setting as well.

3 The Structure of the Branching Algorithm

In this section we describe the structure of the branching algorithm for Terminal Separation. Before we state the main result, we introduce the potential that will measure the progress made in each branching step.

Let $\mathcal{I} = (G, \mathcal{T}, (A^\circ, B^\circ), k)$ be a Terminal Separation instance, where (A_0, B_0) is a maximal terminal separation; we henceforth call such an instance *maximal*. We are interested in keeping track of the following partial measures:

- $t_{\mathcal{I}}$ is the number of unresolved terminal pairs;
- $\nu_{\mathcal{I}} = k - c(A^\circ, B^\circ)$;
- $k_{\mathcal{I}} = k$.

The $\mathcal{O}(2^k km)$-time algorithm used in [14] can be interpreted in our framework as an $\mathcal{O}(2^{t_{\mathcal{I}}} k_{\mathcal{I}} m)$-time algorithm for Terminal Separation, while the generic LP-branching algorithm for Edge Unique Label Cover of [30] can be interpreted as an $\mathcal{O}(4^{\nu_{\mathcal{I}}} m)$-time algorithm. As announced in the intro, our main goal is to blend these two algorithms, by analysing the cases where both these algorithms perform badly.

An important insight is that all these inefficient cases happen when A° and B° increase their common boundary. If this is the case, a simple reduction rule is applicable that also reduces the allowed budget k; in some sense, with this reduction rule the budget k represents the yet undetermined part of the boundary between A^* and B^* in the final integral solution (A^*, B^*). For this reason, we also include the budget k in the potential.

Formally, we fix three constants $\alpha_t = 0.59950$, $\alpha_\nu = 0.29774$, and $\alpha_k = 1 - \alpha_t - \alpha_\nu = 0.10276$ and define a potential of an instance $\mathcal{I}$ as

$$\mu_{\mathcal{I}} = \alpha_t \cdot t_{\mathcal{I}} + \alpha_\nu \cdot \nu_{\mathcal{I}} + \alpha_k \cdot k_{\mathcal{I}}.$$

Our main technical result, proved in the remainder of this paper, is the following.

Theorem 3.1 *A* Terminal Separation *instance $\mathcal{I}$ can be solved in time $\mathcal{O}(c^{\mu_{\mathcal{I}}} nm)$ for some $c < 1.977$.*

Observe that if $\mathcal{I}$ is an instance output by the reduction of Lemma 2.2, then $t_{\mathcal{I}} = |X'| = k + 1$, $\nu_{\mathcal{I}} = k$ since $A^\circ = B^\circ = \emptyset$, and $k_{\mathcal{I}} = k$. Consequently, $\mu_{\mathcal{I}} < k + 1$, and Theorem 1.1 follows from Theorem 3.1.

The algorithm of Theorem 3.1 follows a typical outline of a recursive branching algorithm. At every step, the current instance is analyzed, and either it is reduced, or some two-way branching step is performed. The potential $\mu_{\mathcal{I}}$ is used to measure the progress of the algorithm and to limit the size of the branching tree.

3.1 Reductions

We use a number of reductions in our algorithm. Every reduction decreases $|V(G)| + |\mathcal{T}| + k$, and after any application of any reduction we re-run Theorem 2.4 to ensure that the considered instance is maximal.

The first one is the trivial termination condition.

Reduction 1 (Terminator Reduction) If $k_{\mathcal{I}} < 0$ or $\nu_{\mathcal{I}} < 0$, then we terminate the current branch with the conclusion that there is no solution. If (A°, B°) is integral, return it as a solution.

Observe that if all terminals are resolved, then both $(A^\circ, V(G)\backslash A^\circ)$ and $(V(G)\backslash B^\circ, B^\circ)$ are integral separations, and one of them is of cost at most $c(A^\circ, B^\circ)$. Consequently, since $\mathcal{I}$ is maximal, in fact (A°, B°) is integral. We infer that if the Terminator

Reduction does not trigger, then there exists at least one unresolved terminal pair, i.e., $t_{\mathcal{I}} > 0$.

We now provide the promised reduction of the boundary between A° and B°.

Reduction 2 (Boundary Reduction) If there exists an edge ab with $a \in A^\circ$, $b \in B^\circ$, delete the edge ab and decrease k by one. If there exist two edges va, vb with $a \in A^\circ$, $b \in B^\circ$, and $v \notin A^\circ \cup B^\circ$, delete both edges va and vb, and decrease k by one.

Lemma 3.2 *Let $\mathcal{I} = (G, \mathcal{T}, (A^\circ, B^\circ), k)$ be a maximal* Terminal Separation *instance, and assume that the Boundary Reduction have been applied once, giving a graph G'. Then $\mathcal{I}' = (G', \mathcal{T}, (A^\circ, B^\circ), k - 1)$ is a maximal* Terminal Separation *instance, equivalent to $\mathcal{I}$. Furthermore, $\mu_{\mathcal{I}'} = \mu_{\mathcal{I}} - \alpha_k$.*

Proof Observe that whether (A, B) is a terminal separation extending (A°, B°) does not depend on the instance we are looking at: $\mathcal{I}$ and $\mathcal{I}'$ differ only in the edgeset of the graph and the budget. For such a separation, by $c(A, B)$ we denote its cost in $\mathcal{I}$, and by $c'(A, B)$ its cost in $\mathcal{I}'$.

We claim that for any terminal separation (A, B) extending (A°, B°) it holds that $c(A, B) = c'(A, B) + 1$. The claim is straightforward if an edge ab is deleted. For the second case, consider subcases depending on where the vertex v lies. If $v \in A$, then $d_G(A) = d_{G'}(A) + 1$ due to missing edge vb, while if $v \notin A$, then also $d_G(A) = d_{G'}(A) + 1$ due to missing edge va. Symmetrically, $d_G(B) = d_{G'}(B) + 1$, which proves the claim. Consequently, the instances $\mathcal{I}$ and $\mathcal{I}'$ are equivalent, and (A°, B°) remains a maximal separation. Furthermore, since $c(A^\circ, B^\circ) = c'(A^\circ, B^\circ) + 1$, we have $t_{\mathcal{I}'} = t_{\mathcal{I}}$ and $\nu_{\mathcal{I}'} = \nu_{\mathcal{I}}$, hence $\mu_{\mathcal{I}'} = \mu_{\mathcal{I}} - \alpha_k$. □

It is easy to observe that the Boundary Reduction can be applied exhaustively in linear time.

In a number of reductions in this section, in a few places in the analysis of different cases in the branching algorithm, as well as in the reduction rules defined in the next section, we find a set $X \subseteq V(G)$ of at least two vertices without any terminals, with at least one vertex of $V(G) \setminus (A^\circ \cup B^\circ)$, for which we can argue that there exists an integral solution (A^*, B^*) to $\mathcal{I}$ of minimum cost such that $X \subseteq A^*$ or $X \subseteq B^*$. In this case, we identify X into a single vertex (that belongs to A° if $X \cap A^\circ \neq \emptyset$ and to B° if $X \cap B^\circ \neq \emptyset$), and start from the beginning.

Note that after such reduction (A°, B°) may not be a maximal separation if the contracted set X contains at least one vertex of $A^\circ \cup B^\circ$, and we need to apply Theorem 2.4 to extend it to a maximal one. However, note that the operation of merging vertices only shrinks the space of all terminal separations, and thus the cost of (A°, B°) cannot decrease with such a reduction (and, consequently, $\nu_{\mathcal{I}}$ cannot increase).

We now introduce four simple rules. The first one reduces clearly superfluous pieces of the graph.

Reduction 3 (Pendant Reduction) If there exists a vertex set $X \subseteq V(G) \setminus (A^\circ \cup B^\circ)$ that does not contain any terminal and $|N(X)| \leq 1$, then delete X from G.

If there exists a vertex set $X \subseteq V(G) \setminus (A^\circ \cup B^\circ)$ that does not contain any terminal and $|N(X)| = 2$, then let λ be the size of the minimum (edge) cut between the two vertices of $N(X)$ in $G[N[X]]$. If $\lambda \leq k$, then replace X with λ edges between the two vertices of $N(X)$, and otherwise identify $N[X]$ into a single vertex.

The safeness of the Pendant Reduction is straightforward: in the first case, for any integral separation (A, B) of the reduced graph, one can add X to the set A or B that contains $N(X)$, without increasing the cost of the separation, while in the second case we can do exactly the same if $N(X)$ belongs to the same side A or B, and otherwise we can greedily cut $G[N[X]]$ along the minimum cut between the vertices of $N(X)$.

Observe that an application of a Pendant Reduction does not merge two terminals and does not spoil the invariant that every terminal in G is of degree at most one. If $|N(X)| \leq 1$, then clearly the deletion of X cannot spoil this property. Otherwise, if $|N(X)| = 2$, then λ is at most the degree of any vertex of $N(X)$ in $G[N[X]]$; in particular, if $N(X)$ contains a terminal, then $\lambda \leq 1$ and the vertices of $N(X)$ are not identified.

This reduction also does not decrease $c(A^\circ, B^\circ)$ (and thus does not increase $\nu_{\mathcal{I}}$). This is clear for $N(X) = \emptyset$. For $|N(X)| = 1$ or $\lambda > k$, it can be modelled as identifying $N[X]$ into a single vertex. Otherwise, for $|N(X)| = 2$ and $\lambda \leq k$, it can be modelled as identifying the sides of a minimum cut in $G[N[X]]$ between vertices of $N(X)$ onto the corresponding elements of $N(X)$.

Let us now argue that the Pendant Reduction can be applied efficiently.

Lemma 3.3 *One can in $\mathcal{O}(km)$ time find a set X on which the Pendant Reduction is applicable, or correctly conclude that no such set exists.*

Proof First, compute an auxiliary graph G' from G by adding a clique K on four vertices, and making K fully adjacent to $L := A^\circ \cup B^\circ \cup \mathcal{T}$. In this manner, the size of G' is bounded linearly in the size of G, while $G'[K \cup L]$ is three-connected. Compute the decomposition into three-connected components [2,29], which can be done in linear time [15]. It is easy to see that the Pendant Reduction is not applicable if and only if the decomposition consists of a single bag, and otherwise any leaf bag of the decomposition different than the bag containing $K \cup L$ equals $N[X]$ for some X to which the Pendant Reduction is applicable. Furthermore, for such a set X with $|N(X)| = 2$, one can compute $\min(\lambda, k+1)$ in time $\mathcal{O}(km)$ using $\mathcal{O}(k)$ rounds of the Ford–Fulkerson algorithm. □

The next three reduction rules consider some special cases of how terminals can lie in the graph.

Reduction 4 (Lonely Terminal Reduction) If there exists an unresolved terminal pair $P = \{s, t\}$ such that s is an isolated vertex, delete P from $\mathcal{T}$ and $V(G)$.

The safeness of the Lonely Terminal Reduction follows from the observation that in every terminal separation (A, B) of the reduced graph, we can always put t on the same side as its neighbor (if it exists) and s on the opposite side.

Reduction 5 (Adjacent Terminals Reduction) If there exist two neighboring unresolved terminals t_1 and t_2, then proceed as follows. If they belong to the same terminal pair, delete both of them from G and from $\mathcal{T}$, and reduce k by one. If they belong to different terminal pairs, say $\{s_1, t_1\}$ and $\{s_2, t_2\}$, then delete both these terminal pairs from $\mathcal{T}$, delete the vertices t_1 and t_2 from G, and add an edge $s_1 s_2$.

For safeness of the Adjacent Terminals Reduction, first recall that terminals are of degree one in G, thus $\{t_1, t_2\}$ is a connected component of G. If they belong to the same terminal pair, the edge t_1t_2 always belongs to the solution cut and can be deleted. If they belong to different terminal pairs $\{s_1, t_1\}$ and $\{s_2, t_2\}$, then the edge t_1t_2 is cut by a solution (A^*, B^*) if and only if s_1 and s_2 are on different sides of the solution, thus we can just as well account for it by replacing it with an edge s_1s_2.

Reduction 6 (Common Neighbor Reduction) If there exists an unresolved terminal pair $\{s, t\} \in \mathcal{T}$, such that s and t share a neighbor a, then delete both terminals from $\mathcal{T}$ and G, and decrease k by one.

For safeness of the Common Neighbor Reduction, note that in any solution, exactly one edge as or at is cut.

It is straightforward to check in linear time if any of the last three reductions is applicable, and apply one if this is the case. It follows from maximality of (A°, B°) and the above safeness arguments that none of these reductions increases $\nu_{\mathcal{I}}$: any potential extension (A, B) of (A°, B°) in the reduced graph can be translated to an extension in the original graph, with a cost larger than $c(A, B)$ by exactly the number of times the budget k has been decreased by the reduction. Consequently, every application of any of the last three reductions decreases the potential $\mu_{\mathcal{I}}$ by at least α_t, as each removes at least one terminal pair.

The last reduction is the following.

Reduction 7 (Majority Neighbour Reduction) If there exists two non-terminal vertices $u, v \in V(G)\setminus(A^\circ \cup B^\circ)$ such that at least half of the edges incident to u have the second endpoint in v, identify u and v.

The safeness of the Majority Neighbour Reduction is straightforward: in any integral separation that puts u and v on opposite sides, changing the side of u does not increase the cost of the separation. Also, it is straightforward to find vertices u, v for which the Majority Neighbour Reduction applies and execute it in linear time. Note that, since we require $u, v \notin A^\circ \cup B^\circ$, the considered instance remains maximal.

Two more reduction rules will be introduced in Sect. 4, where we study sets $A \supseteq A^\circ$ with small $d(A) - d(A^\circ)$.

3.2 Branching Step

In every branching step, we identify two terminal separations (A_1, B_1) and (A_2, B_2) extending (A°, B°), and branch into two subcases; in subcase i we replace (A°, B°) with (A_i, B_i). We always argue the *correctness* of a branch by showing that there exists a solution (A^*, B^*) extending (A°, B°) of minimum cost, with the additional property that (A^*, B^*) extends (A_i, B_i) for some $i = 1, 2$. In subcase i, we apply the algorithm of Theorem 2.4 to $(G, \mathcal{T}, (A_i, B_i), k)$ to obtain a maximal separation (A_i°, B_i°), and pass the instance $\mathcal{I}_i = (G, \mathcal{T}, (A_i^\circ, B_i^\circ), k)$ to a recursive call.

To show the *running time bound* for a branching step, we analyze how the measure $\mu_{\mathcal{I}}$ decreases in the subcases, taking into account the reductions performed in the subsequent recursive calls. More formally, we say that a branching case *fulfills a branching vector*

$$[t_1, \nu_1, k_1;\ t_2, \nu_2, k_2]$$

if, in subcase $i = 1, 2$, at least t_i terminal pairs become resolved or reduced with one of the reductions, the cost of the separation (A_i°, B_i°) grows by at least $\nu_i/2$, and the Boundary Reduction is applied at least k_i times in the instance $(G, \mathcal{T}, (A_i^\circ, B_i^\circ), k)$.

A branching vector $[t_1, \nu_1, k_1; t_2, \nu_2, k_2]$ is *good* if

$$1.977^{-\alpha_t t_1 - \alpha_\nu \nu_1/2 - \alpha_k k_1} + 1.977^{-\alpha_t t_2 - \alpha_\nu \nu_2/2 - \alpha_k k_2} < 1.$$

In other words, if in subcase $i = 1, 2$, the potential $\mu_\mathcal{I}$ of the instance $\mathcal{I}$ decreases by δ_i, then we require that $1.977^{-\delta_1} + 1.977^{-\delta_2} < 1$. A standard inductive argument for branching algorithms show that, if in every case we perform a branching step that fulfills some good branching vector, the branching tree originating from an instance $\mathcal{I}$ has $\mathcal{O}(c^{\mu_\mathcal{I}})$ leaves for some $c < 1.977$ (so that $c^{\mu_\mathcal{I} - \delta_1} + c^{\mu_\mathcal{I} - \delta_2} \leq c^{\mu_\mathcal{I}}$). To simplify further exposition, we gather in the next lemma good branching vectors used in the analysis; the fact that they are good can be checked by direct calculations.

Lemma 3.4 *The following branching vectors are good:*

$[1, 1, 0; 2, 1, 0]$	$[1, 1, 1; 1, 2, 3]$	$[1, 2, 0; 1, 3, 1]$	$[1, 1, 0; 1, 4, 3]$
$[1, 1, 2; 1, 2, 2]$	$[1, 1, 1; 1, 3, 2]$	$[1, 3, 0; 1, 3, 0]$	$[1, 1, 0; 1, 5, 2]$
$[1, 2, 1; 1, 2, 2]$	$[1, 1, 1; 1, 4, 1]$		

Let us stop here to comment that the vectors in Lemma 3.4 explain our choice of constants $\alpha_t, \alpha_\nu, \alpha_k$. The constant α_t is sufficiently large to make the vector $[1, 1, 0; 2, 1, 0]$ good; intuitively speaking, we are always done when in one branch we manage to resolve or reduce at least two terminal pairs. The choice of α_ν and α_k represents a very delicate tradeoff that makes both $[1, 1, 1; 1, 2, 3]$ and $[1, 2, 0; 1, 3, 1]$ good; note that setting $\alpha_\nu = 1 - \alpha_t$ and $\alpha_k = 0$ makes the first vector not good, while setting $\alpha_\nu = 0$ and $\alpha_k = 1 - \alpha_t$ makes the second vector not good.[3] In fact, arguably the possibility of a tradeoff that makes both the second and the third vector of Lemma 3.4 good at the same time is one of the critical insights in our work.

3.3 Running Time Bound

In the subsequent sections, we will only argue that

1. every single application of a reduction or a branching step is executed in $\mathcal{O}(k^{\mathcal{O}(1)} m)$ time;

[3] One can observe that the goodness of all other vectors mentioned in Lemma 3.4, can be easily deduced from the goodness of the first four vectors, by using the fact that $\alpha_t \geq \alpha_\nu \geq \alpha_k$ and that a branching vector cannot stop being good if one moves weight from the "heavier" side to the "lighter" one.

2. every reduction either terminates or reduces $k + |\mathcal{T}| + |V(G)|$ by at least one; note that this is true for the reductions defined so far;
3. every branching step is correct and fulfills one of the good vectors mentioned in Lemma 3.4.

Observe that these properties guarantee correctness and the claimed running time of the algorithm.

In a number of places in the branching algorithm, the algorithm *attempts* some branching $(A_1, B_1), (A_2, B_2)$, and withdraws this decision if the measure decrease is too small. A naive implementation of such behaviour would lead to an additional n factor in the running time bound, as exhaustive application of our reduction rules may take $\mathcal{O}(k^{\mathcal{O}(1)}nm)$ time, only to be later withdrawn. To maintain the $\mathcal{O}(nm)$ polynomial factor in our running time bound, we restrict such attempts to only the following procedure: for $i = 1, 2$, we apply Theorem 2.4 to obtain a minimum cost extension (A_i°, B_i°) of (A_i, B_i), and report:

1. the number of terminal pairs contained in $(A_i^\circ \cup B_i^\circ) \backslash (A^\circ \cup B^\circ)$, i.e., the immediate decrease in $t_{\mathcal{I}}$;
2. the difference $c(A_i^\circ, B_i^\circ) - c(A^\circ, B^\circ)$, i.e., the immediate decrease in $\nu_{\mathcal{I}}$;
3. the number of immediately applicable Boundary Reductions, defined as follows:

$$\rho_i := |E(A_i^\circ, B_i^\circ)| + \sum_{v \in V(G) \backslash (A_i^\circ \cup B_i^\circ)} \min(|E(v, A_i^\circ)|, |E(v, B_i^\circ)|).$$

Clearly, the aforementioned numbers are computable in $\mathcal{O}(k^{\mathcal{O}(1)}m)$ time.

4 Low Excess Sets

Let $\mathcal{I} = (G, \mathcal{T}, (A^\circ, B^\circ), k)$ be a maximal Terminal Separation instance. A set $A \subseteq V(G)$ is an A°-*extension* if $A^\circ \subseteq A \subseteq V(G) \backslash B^\circ$. It is *terminal-free* if $A \backslash A^\circ$ does not contain any terminal. We denote by $\Delta(A) := d(A) - d(A^\circ)$ the *excess* of an A°-extension A. An A°-extension A is *compact* if $A \backslash A^\circ$ is connected and $E(A \backslash A^\circ, A^\circ) \neq \emptyset$.

In this section we consider extensions of small excess, and show that their structure can be reduced to have a relatively simple picture. While in this section we focus on supersets of the set A°, by symmetry the same conclusion holds if we swap the roles of A° and B°. In our algorithm, we exhaustively apply the reduction rules defined in this section both for the A-side and B-side of the separation (A°, B°).

Before we start, let us first observe that we can efficiently enumerate all maximal sets of particular constant excess.

Lemma 4.1 *For every fixed constant r, one can in $\mathcal{O}(k^{\mathcal{O}(1)}(n + m))$ time enumerate all inclusion-wise maximal compact A°-extensions of excess at most r.*

Proof Our algorithm will in fact enumerate all compact A°-extensions A of excess at most r with the property that every compact A°-extension A' with $A \subsetneq A'$ satisfies

$\Delta(A') > \Delta(A)$. The approach closely follows the algorithm for enumerating important separators (see, e.g., [5, Chapter 8]).

By the maximality of (A°, B°), A° is the only such extension of excess 0. We initiate a queue Q with $Q = \{A^\circ\}$. Iteratively, until Q is not empty, we extract an extension A from Q, and proceed as follows. For every $v \in N(A)$, we compute a set A_v such that $E(A_v, V(G)\backslash A_v)$ is a minimum cut between $A \cup \{v\}$ and $B^\circ \cup (\mathcal{T}\backslash A^\circ)$, or take $A_v = \bot$ if for such a set $d(A_v)$ would be larger than $d(A^\circ) + r$. Such a set A_v can be computed using $\mathcal{O}(k + r)$ rounds of the Ford–Fulkerson algorithm, and furthermore it allows us to compute A_v being the unique inclusion-wise maximal set with the required properties.

If $A_v \neq \bot$, we insert A_v into the queue Q. Otherwise, if $A_v = \bot$ for every $v \in N(A)$, then we output A as one of the desired sets. For correctness, observe that every set A in the queue has excess at most r, and the described procedure uses the definition of compactness to check if there exists any other extension of excess at most r being a strict superset of A. For the time bound, observe that whenever a set A_v is inserted into the queue, it holds that $d(A_v) > d(A)$, while $d(A^\circ) \leq 2k$ (because of the Terminator Reduction). Hence, $\mathcal{O}((2k + r)^r)$ sets are inserted into the queue. Moreover, the computation for a single set A extracted from the queue takes $\mathcal{O}(k^{\mathcal{O}(1)}(n + m))$ time. □

We now proceed to the promised description of reductions. A straightforward corollary of the assumption that $\mathcal{I}$ is maximal is the following.

Lemma 4.2 *If A is a terminal-free A°-extension of excess zero or less, then $A = A^\circ$.*

We now study extensions of excess 1.

Lemma 4.3 *If A is a terminal-free A°-extension of excess* 1*, then there exists a minimum cost integral terminal separation (A^*, B^*) extending (A°, B°), such that $(A\backslash A^\circ)$ is either completely contained in A^* or completely contained in B^*.*

Proof Let (A^*, B^*) be a minimum cost integral terminal separation extending (A°, B°).

If $(A\backslash A^\circ)$ is completely contained in B^*, then (A^*, B^*) proves the claim, so let us assume the contrary: $(A\backslash A^\circ) \cap A^* \neq \emptyset$. Then $A \cap A^* \neq A^\circ$. We show that $(A^* \cup A, B^*\backslash A)$ is a minimum cost integral separation, proving the claim.

Indeed, since A is terminal-free, $(A^* \cup A, B^*\backslash A)$ is an integral terminal separation. It suffices to show that it is minimum, that is, $d(A^* \cup A) \leq d(A^*)$. By submodularity, $d(A^* \cup A) + d(A^* \cap A) \leq d(A^*) + d(A)$. Since $A \cap A^*$ is a terminal-free A°-extension and $A \cap A^* \neq A^\circ$, by Lemma 4.2 we have $\Delta(A \cap A^*) > 0$, which means $d(A \cap A^*) \geq 1 + d(A^\circ)$. By assumption $d(A) = 1 + d(A^\circ)$. Taking this together, $d(A^* \cup A) \leq d(A^*) + d(A) - d(A^* \cap A) \leq d(A^*)$, which concludes the proof. □

Lemma 4.3 proves safeness of the following reduction rule.

Reduction 8 (Excess-1 Reduction) If there exists a terminal-free A°-extension of excess 1 with $|A\backslash A^\circ| > 1$, merge all vertices of $A\backslash A^\circ$ into a single vertex.

The next lemma shows that one can apply the Excess-1 Reduction efficiently.

Lemma 4.4 *Given a maximal instance $\mathcal{I}$ for which none of the previously defined reduction rules is applicable, one can in $\mathcal{O}(k^{\mathcal{O}(1)}(n+m))$ time find a set A for which the Excess-1 Reduction rule is applicable, or correctly conclude that no such set exists.*

Proof Let A be a terminal-free A°-extension of excess 1. If $A \backslash A^\circ$ is disconnected, then for any connected component C of $A \backslash A^\circ$ we have that $d(A^\circ \cup C) + d(A \backslash C) = d(A^\circ) + d(A)$, hence either $d(A^\circ \cup C) \leq d(A^\circ)$ or $d(A \backslash C) \leq d(A^\circ)$, contradicting the maximality of (A°, B°). Thus, $A \backslash A^\circ$ is connected. If $E(A \backslash A^\circ, A^\circ)$ were empty, then $A \backslash A^\circ$ would be a terminal-free set with $d(A \backslash A^\circ) = 1$, and would hence be deleted by the Pendant Reduction.

Consequently, every terminal-free A°-extension of excess 1 is compact. We can enumerate all such inclusion-wise maximal extensions by Lemma 4.1, and apply the reduction for any such set A with $|A \backslash A^\circ| > 1$. □

We can henceforth assume that for every terminal-free A°-extension A of excess 1, the set $A \backslash A^\circ$ is a singleton.

We now move to an analysis of sets of excess 2.

Lemma 4.5 *Assume that the Pendant Reduction and Excess-1 Reduction have been exhaustively applied. If A is a terminal-free A°-extension of excess 2, then there exists a partition $A \backslash A^\circ = D \uplus C_1 \uplus C_2 \uplus \cdots \uplus C_r$ for some $r \geq 0$, such that:*

1. *there exists a minimum cost integral terminal separation (A^*, B^*) extending (A°, B°), such that one of the following holds:*
   - *$(A \backslash A^\circ) \cap A^* = \emptyset$;*
   - *$(A \backslash A^\circ) \cap A^* = C_i$ for some $1 \leq i \leq r$; or*
   - *$A \subseteq A^*$.*
2. *for every $1 \leq i \leq r$, the sets C_i and $E(C_i, A^\circ)$ are nonempty, and $A^\circ \cup C_i$ is a terminal-free A°-extension of excess 1;*
3. *if $D \neq \emptyset$, then for every $1 \leq i \leq r$ the set $E(C_i, D)$ is nonempty and $A \backslash A^\circ$ is connected;*
4. *if $D = \emptyset$, then $r = 2$;*
5. *for every $1 \leq i < j \leq r$, there are no edges between C_i and C_j.*

Proof Let $C'_1, \ldots, C'_r$ be all the inclusion-wise maximal subsets of A that are A°-extensions of excess 1. Let $C_i = C'_i \backslash A^\circ$ and let $D = A \backslash (A^\circ \cup C_1 \cup \cdots \cup C_r)$. We show the claim is true for these sets. Let $1 \leq i \neq j \leq r$.

The Excess-1 Reduction allows us to assume that C_i is a singleton and hence C_i is disjoint from C_j. Since $\Delta(C'_i) = \Delta(C'_j) = 1$ and $\Delta(C'_i \cup C'_j) \geq 2$ (by maximality of C'_i), there are no edges between C_i and C_j, proving point 5.

If $E(C_i, A^\circ)$ were empty, then $d(C_i) = 1$ and C_i would be deleted by the Pendant Reduction; this proves point 2. If $D \neq \emptyset$ but $A \backslash A^\circ$ was disconnected, then consider a component C of $A \backslash A^\circ$. Then $\Delta(A) = \Delta(A^\circ \cup C) + \Delta(A \backslash C)$, hence either $\Delta(A^\circ \cup C) = \Delta(A \backslash C) = 1$, which would contradict that $D \neq \emptyset$, or one of $A^\circ \cup C$, $A \backslash C$ has excess 0, which would contradict Lemma 4.2. Hence $A \backslash A^\circ$ is connected and as there are no edges between C_i and C_j, there must be edges between C_i and D, proving point 3. If $D = \emptyset$, then $\Delta(A) = \sum_{i=1}^{r} \Delta(A^\circ \cup C_i) = r$. Hence $r = 2$, proving point 4.

To prove point 1, consider a minimum cost integral terminal separation (A^*, B^*). Since C_i is a singleton, it is either completely contained in A^* or disjoint from it. If $A^* \cap (A \setminus A^\circ)$ is empty or equal to one of C_i, the claim follows. Otherwise, $A^* \cap (A \setminus A^\circ)$ contains a vertex of D or two of the C_i sets; by their maximality, the excess of $A^* \cap A$ is then at least 2, so $d(A^* \cap A) \geq d(A)$. By submodularity, $d(A^* \cup A) + d(A^* \cap A) \leq d(A^*) + d(A)$ and thus $d(A^* \cup A) \leq d(A^*)$. Therefore, since A is terminal-free, $(A^* \cup A, B^* \setminus A)$ is an integral terminal separation, concluding the proof. □

Lemma 4.5 ensures safeness of the following reduction rule.

Reduction 9 (Excess-2 Reduction) If there exists a terminal-free A°-extension A of excess 2 such that in the partition $D \uplus C_1 \uplus \cdots \uplus C_r$ defined by Lemma 4.5, $|D| > 1$, then merge D into a single vertex.

We are left with an efficient implementation of this rule.

Lemma 4.6 *Given a maximal instance $\mathcal{I}$ for which none of the previously defined reduction rules is applicable, one can in $\mathcal{O}(k^{\mathcal{O}(1)}(n+m))$ time find a set A for which the Excess-2 Reduction is applicable and compute the decomposition of $A \setminus A^\circ$ of Lemma 4.5, or correctly conclude that no such set A exists.*

Proof Let A be a terminal-free A°-extension of excess 2, and let $D, C_1, C_2, \ldots, C_r$ be the sets promised by Lemma 4.5 and let $|D| > 1$. The inapplicability of the Excess-1 Reduction ensures that every set C_i is a singleton, $C_i = \{c_i\}$.

Let us first deal with the corner case in which $r = 0$ and $E(D, A^\circ) = \emptyset$. Then, since A is of excess 2, we have $d(D) = 2$. However, as D does not contain any terminal, the Pendant Reduction is applicable to it.

In the remaining cases, Lemma 4.5 guarantees that A is compact. We enumerate all inclusion-wise maximal compact excess-2 extensions using Lemma 4.1. For every output extension A, we first identify the set $C \subseteq A \setminus A^\circ$ of all vertices v such that $A^\circ \cup \{v\}$ is of excess one. By Lemma 4.5, we have $D = A \setminus (A^\circ \cup C)$. If $|D| > 1$, then we can apply the reduction.

To complete the proof, note that if the Excess-2 Reduction is applicable to some compact A°-extension A, then it is also applicable to any compact A°-extension A' of excess 2 being a superset of A: the corresponding set D for A is a subset of the corresponding set D' for A'. □

The set D of Lemma 4.5 is often a very convenient branching pivot: putting it into A° makes the boundary of A° extend by two, while putting it into B° triggers a number of Boundary Reductions. In the next few lemmata we summarize the properties of an excess-2 set after reductions, and outcomes on branching on the set D.

We start from a slightly more useful presentation of the properties promised by Lemma 4.5 (Fig. 1).

Lemma 4.7 *Assume that no reduction is applicable, and let A be a terminal-free A°-extension of excess 2. Then one can in $\mathcal{O}(k^{\mathcal{O}(1)}m)$ time compute a decomposition $A \setminus A^\circ = \{d, c_1, c_2, \ldots, c_r\}$ for some $r \geq 0$ or $A \setminus A^\circ = \{c_1, c_2\}$ with the following properties:*

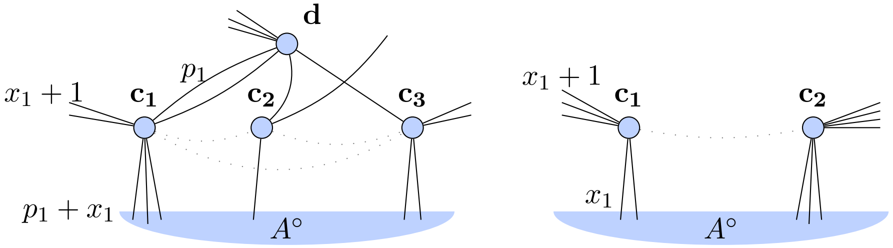

Fig. 1 Examples of sets of excess 2 after reductions (*dotted lines* are non-edges)

1. *if the vertex d exists, then A is compact and for every $1 \leq i \leq r$, there are p_i edges dc_i for some $p_i \geq 1$; we put $p_1 = p_2 = 0$ if the vertex d does not exists;*
2. *for every $1 \leq i \leq r$, the set $A^\circ \cup \{c_i\}$ is an A°-extension of excess 1, the vertex c_i has $x_i + 1 \geq 1$ edges towards $V(G)\backslash(A \cup B^\circ)$ and $p_i + x_i \geq 1$ edges towards A°, for some $x_i \geq 0$;*
3. *the vertices c_i are pairwise nonadjacent;*
4. *the set $A^\circ \cup \{d\}$ is an A°-extension of excess larger than 1.*

Proof Most of the enumerated properties are just repetitions of the points of Lemma 4.5, after each set of the partition has been identified into a single vertex. Recall that noncompact A°-extensions of excess 2 are completely reduced by the Pendant Reduction.

For the count on the number of edges incident to a vertex c_i, define p_i as claimed and $x_i := |E(c_i, V(G)\backslash A)|-1$; clearly $x_i \geq -1$. Since $A^\circ \cup \{c_i\}$ is of excess 1, and no two vertices c_i are adjacent, we have $|E(c_i, A^\circ)| = p_i + x_i$. Furthermore, note that no edge may connect c_i and B°, as it would trigger a Boundary Reduction. It remains to refute the case $x_i = -1$, i.e., $E(c_i, V(G)\backslash A) = \emptyset$. In this case $p_i + x_i = |E(c_i, A^\circ)| \geq 0$ implies $p_i \geq 1$, so the vertex d exists. However, the Majority Neighbour Reduction then applies to c_i and d, a contradiction.

If $A^\circ \cup \{d\}$ is an A°-extension of excess at most 1, then $r \geq 1$ as A has excess 2, but then an edge count shows that $A^\circ \cup \{d, c_1\}$ would be an A°-extension of nonpositive excess, a contradiction to the maximality of A°.

Finally, the decomposition of $A\backslash A^\circ$ can be identified by inspecting the edges incident to every vertex $v \in A\backslash A^\circ$ to check whether $A^\circ \cup \{v\}$ is of excess 1 or larger. □

We now investigate what happens in a branch when we put the vertex d onto the A-side.

Lemma 4.8 *Assume that no reduction is applicable, and let A, A' be two terminal-free A°-extensions of excess 2 with $A \subsetneq A'$. Then $A'\backslash A^\circ$ decomposes as $\{d, c_1, c_2, \ldots, c_r\}$ for some $r \geq 2$, and $A\backslash A^\circ$ consists of two vertices c_i of this decomposition.*

Proof If $A'\backslash A^\circ = \{c_1, c_2\}$, then there is no choice for the set A, as $A^\circ \cup \{c_i\}$ is of excess 1 for $i = 1, 2$. Hence, $A'\backslash A^\circ = \{d, c_1, c_2, \ldots, c_r\}$ for some $r \geq 1$; note that $|A'\backslash A^\circ| \geq 2$ as $A^\circ \subsetneq A \subsetneq A'$. A direct edge count using Lemma 4.7 shows that for every $C \subseteq \{c_1, c_2, \dots, c_r\}$ we have $\Delta(A^\circ \cup C) = |C|$ and $\Delta(A^\circ \cup C \cup \{d\}) \geq 2 + (r - |C|)$. Hence, the only option to get excess 2 is to have $A = A^\circ \cup C$ for some $|C| = 2$. □

Lemma 4.9 *Assume that no reduction is applicable, and let A be a terminal-free A°-extension of excess 2 with $A \setminus A^\circ = \{d, c_1, c_2, \dots, c_r\}$ for some $r \geq 0$. If we furthermore consider a branch (A_1, B_1) such that $d \in A_1$, but $A_1 \setminus A^\circ$ does not contain any terminal, then*

1. *if B_1 contains at least one vertex c_i, then there does not exist any minimum cost integral terminal separation (A^*, B^*) extending (A°, B°) that also extends (A_1, B_1);*
2. *$d(A_1) \geq d(A^\circ) + 2$;*
3. *if $d(A_1) = d(A^\circ) + 2$, then $A_1 = A$.*

Proof Define $A' := A_1 \cup A$ and $B' := B_1 \setminus A$; note that $A' \setminus A^\circ$ is terminal-free and (A', B') is a terminal separation as well.

Observe that if (A_1, B_1) is a terminal separation extending (A°, B°) with $d \in A_1$ but $c_i \notin A_1$ for some $1 \leq i \leq r$, then a direct edge count from Lemma 4.7 shows that $d(A_1 \cup \{c_i\}) < d(A_1)$, $d(B_1 \setminus \{c_i\}) \leq d(B_1)$, hence $c(A_1 \cup \{c_i\}, B_1 \setminus \{c_i\}) < c(A_1, B_1)$. This proves the first point, and shows that $d(A') \leq d(A_1)$, $d(B') \leq d(B_1)$, thus $c(A', B') \leq c(A_1, B_1)$, and the equality holds only if $(A', B') = (A_1, B_1)$.

Since $A \subseteq A'$, the Excess-1 Reduction is inapplicable, and $\Delta(A) = 2$, we have $\Delta(A') \geq 2$. Consequently, $d(A_1) \geq d(A') \geq d(A^\circ) + 2$, and $d(A_1) = d(A^\circ) + 2$ only if $d(A_1) = d(A') = d(A^\circ) + 2$. As discussed in the previous paragraph, this can only happen if $A' = A_1$ and $\Delta(A') = 2$. By Lemma 4.8, this implies $A' = A$, finishing the proof of the lemma. □

In the last lemma we study what happens in a branch when we put the vertex d onto the B-side (Fig. 2).

Lemma 4.10 *Assume that no reduction is applicable, and let A be a terminal-free A°-extension of excess 2 with $A \setminus A^\circ = \{d, c_1, c_2, \dots, c_r\}$ for some $r \geq 0$. Furthermore, if we consider a branch (A_1, B_1) such that $d \in B_1$, then at least one Boundary Reduction is immediately triggered. If only one is triggered, then one of the following holds:*

1. *$r = 0$, $A \setminus A^\circ = \{d\}$, and the vertex d is of degree four, with one incident edge having second endpoint in A° and the remaining three edges having second endpoint in $V(G) \setminus (A \cup B^\circ)$; or*

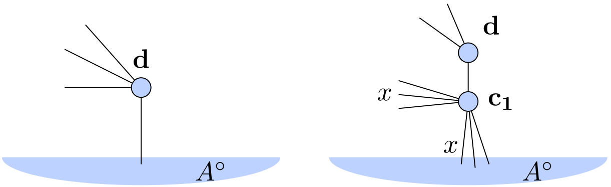

Fig. 2 The two cases when putting d on the unnatural side triggers only one Boundary Reduction

2. *$r = 1$, $A \backslash A^\circ = \{d, c_1\}$, the vertex d is of degree three, with one incident edge being c_1d and the remaining two edges having second endpoint in $V(G)\backslash A$, and the vertex c_1 is of degree $2x + 1$ for some $x \geq 1$, with one incident edge being c_1d, x incident edges having second endpoint in A°, and x incident edges having second endpoint in $V(G)\backslash(A \cup B^\circ)$.*

Proof In the branch (A_1, B_1), a Boundary Reduction is immediately triggered for every edge in $E(d, A^\circ)$, and every vertex c_i triggers $\min(p_i, p_i + x_i) = p_i$ Boundary Reductions. Note that $r \geq 1$ or $E(D, A^\circ) \neq \emptyset$, as A is compact by Lemma 4.7. Hence at least one reduction is triggered. If only one reduction is triggered, then $|E(d, A^\circ)| + \sum_{i=1}^{r} p_i = 1$. In particular r is either 0 or 1.

If $r = 0$, then $|E(d, A^\circ)| = 1$ and the assumption that A is of excess 2 implies that $|E(d, V(G)\backslash A^\circ)| = 3$. No edge incident to d may have a second endpoint in B°, as it would trigger the Boundary Reduction together with the edge in $E(d, A^\circ)$. Thus the first case of the claim holds.

If $r = 1$, then $|E(d, A^\circ)| = 0$ and $p_1 = 1$. Since c_1 has $p_1 + x_1$ edges to A° and $x_1 + 1$ edges to $V(G)\backslash A$, the assumption that A is of excess 2 implies that d has exactly two edges to $V(G)\backslash A$. No edge incident to c_1 can have the second endpoint in B°, as otherwise it would trigger the Boundary Reduction with any edge in $E(c_1, A^\circ)$. Thus the second case of the claim holds. □

5 The Detailed Cases of the Branching Algorithm

In this section we assume we have a maximal instance $\mathcal{I} = (G, \mathcal{T}, (A^\circ, B^\circ), k)$ for which none of the previously defined reduction rules is applicable. Our goal is to find a *branching step* that fulfils a good vector, or a set of vertices to merge (a *reduction step*). Recall that when we consider a branching into terminal separations (A_1, B_1) and (A_2, B_2) that extend (A°, B°), then t_i, ν_i, k_i for $i = 1, 2$ measure respectively the number of terminals resolved in branch i, two times the growth of the cost of the separation in branch i (i.e., $2(c(A_i, B_i) - c(A^\circ, B^\circ))$), and the decrease in the budget k after applying all the reduction rules when recursing into branch i.

Assume that we have identified a branching step into separations (A_1, B_1) and (A_2, B_2) that both extend, but are different than (A°, B°). Then, from the maximality of (A°, B°) we infer than $\nu_1, \nu_2 \geq 1$. Since $[1, 1, 0; 2, 1, 0]$ is a good vector, any branching step in which in both cases we resolve or reduce at least one terminal pair, while in at least one case we resolve or reduce at least *two* terminal pairs, is fine for our purposes.

5.1 Basic Branching and Reductions

Let $\mathcal{T}' \subseteq \mathcal{T}$ be the set of unresolved terminal pairs (not in $A^\circ \cup B^\circ$). For every terminal pair $\{s, t\} \in \mathcal{T}'$, we apply the algorithm of Theorem 2.4 twice: once for terminal separation $(A^\circ \cup \{s\}, B^\circ \cup \{t\})$, and the second time for terminal separation $(A^\circ \cup \{t\}, B^\circ \cup \{s\})$. In this manner we obtain two maximal terminal separations (A_s, B_t) and (A_t, B_s) that extend $(A^\circ \cup \{s\}, B^\circ \cup \{t\})$ and $(A^\circ \cup \{t\}, B^\circ \cup \{s\})$ respectively. Of course, the number of unresolved pairs decreases by at least one in both (A_s, B_t) and (A_t, B_s), due to resolving $\{s, t\}$. If the number of unresolved pairs either in (A_s, B_t) or in (A_t, B_s) decreases by more than one, then, as we argued, performing a branching step $(A_1, B_1) = (A_s, B_t)$ and $(A_2, B_2) = (A_t, B_s)$ leads to the branching vector $[1, 1, 0; 2, 1, 0]$ or a better one, which is good. We can test in $\mathcal{O}(k^{\mathcal{O}(1)}m)$ time whether this holds for any pair $\{s, t\} \in \mathcal{T}'$, and if so then we pursue the branching step.

Branching step 1 If in either (A_s, B_t) or in (A_t, B_s), more than one terminal pair gets resolved, then perform branching into $(A_1, B_1) = (A_s, B_t)$ and $(A_2, B_2) = (A_t, B_s)$.

Hence, if this branching step cannot be performed, then we assume the following:

Assumption 1 For every pair $\{s, t\} \in \mathcal{T}'$ in both (A_s, B_t) and (A_t, B_s) only the pair $\{s, t\}$ gets resolved.

We now proceed with some structural observations about the instance at hand.

Lemma 5.1 $G[A_s \setminus A^\circ]$, $G[A_t \setminus A^\circ]$, $G[B_s \setminus B^\circ]$, $G[B_t \setminus B^\circ]$ *are connected.*

Proof We prove the statement for $G[A_s \setminus A^\circ]$, since the other statements are symmetric. Suppose $G[A_s \setminus A^\circ]$ is disconnected, and let C be any of its connected component that does not contain s. Then C is terminal-free, so by the maximality of (A°, B°) we infer that $d(C \cup A^\circ) > d(A^\circ)$. But then $d(A_s \setminus C) < d(A_s)$, which contradicts the optimality of (A_s, B_s). □

Lemma 5.2 *Let* $\{s, t\} \in \mathcal{T}'$, *and let* (A_s, B_t) *and* (A_t, B_s) *be any optimum-cost terminal separations extending* $(A^\circ \cup \{s\}, B^\circ \cup \{t\})$ *and* $(A^\circ \cup \{t\}, B^\circ \cup \{s\})$, *respectively. Suppose that* (A_s, B_t) *and* (A_t, B_s) *do not resolve any terminal pair apart from* $\{s, t\}$. *Then for any set* A *with* $A^\circ \cup \{s\} \subseteq A \subseteq V(G) \setminus B^\circ$ *that has only* s *among the terminals of* $\mathcal{T}'$, *it holds that* $\Delta(A) \geq \Delta(A_s)$. *Symmetrically, for any set* B *with* $B^\circ \cup \{s\} \subseteq B \subseteq V(G) \setminus A^\circ$ *that has only* s *among the terminals of* $\mathcal{T}'$, *it holds that* $\Delta(B) \geq \Delta(B_s)$.

Proof We prove only the first claim for the second one is symmetric. Let A be such a set, and for the sake of contradiction suppose $\Delta(A) < \Delta(A_s)$. Then $d(A) + d(B_t) < 2c(A_s, B_t)$. However, from posimodularity of cuts it follows that either $d(B_t \setminus A) + d(A) \leq d(B_t) + d(A)$ or $d(B_t) + d(A \setminus B_t) \leq d(B_t) + d(A)$. Both $(A, B_t \setminus A)$ and $(A \setminus B_t, B_t)$ are terminal separations that extend $(A^\circ \cup \{s\}, B^\circ \cup \{t\})$, and one of them has strictly smaller cost than (A_s, B_t). This is a contradiction with the optimality of (A_s, B_t). □

5.1.1 Pushing A_s and B_s

The problem that we will soon face is that separations (A_s, B_t) and (A_t, B_s) are not uniquely defined. For instance, there can be some set of vertices $Z \subseteq A_s \setminus A^\circ$ that could be moved from A_s to B_t without changing the cost of the separation. We now make an adjustment of these separations so that we can assume that A_s, resp. B_s, is maximal. For this, we need the following technical results.

Lemma 5.3 *Suppose that (A_s, B_t) and (A'_s, B'_t) are maximal terminal separations of minimum cost among separations that extend $(A^\circ \cup \{s\}, B^\circ \cup \{t\})$. Suppose further that they do not resolve any other terminal pair from $\mathcal{T}'$. Then*

(a) *$d(A_s) = d(A'_s)$ and $d(B_t) = d(B'_t)$;*
(b) *$(A_s \cap A'_s, B_t \cup B'_t)$ and $(A_s \cup A'_s, B_t \cap B'_t)$ are also terminal separations of minimum cost among separations that extend $(A^\circ \cup \{s\}, B^\circ \cup \{t\})$;*
(c) *$A_s \cup B_t = A'_s \cup B'_t$.*

Proof (a) Let $C = c(A_s, B_t) = c(A'_s, B'_t)$ be the minimum cost of a terminal separation extending $(A^\circ \cup \{s\}, B^\circ \cup \{t\})$. Suppose w.l.o.g. that $d(A_s) < d(A'_s)$, then we have that $d(B_t) > d(B'_t)$. By posimodularity, we have that

$$d(A_s \backslash B'_t) + d(B'_t \backslash A_s) \leq d(A_s) + d(B'_t) < 2C. \tag{5}$$

Observe that $(A_s \backslash B'_t, B'_t)$ is a terminal separation that extends $(A^\circ \cup \{s\}, B^\circ \cup \{t\})$, and hence

$$d(A_s \backslash B'_t) + d(B'_t) = 2c(A_s \backslash B'_t, B'_t) \geq 2C. \tag{6}$$

Symmetrically, by considering terminal separation $(A_s, B'_t \backslash A_s)$ we obtain that

$$d(A_s) + d(B'_t \backslash A_s) = 2c(A_s, B'_t \backslash A_s) \geq 2C. \tag{7}$$

Thus, from (5), (6), and (7) we obtain that

$$4C \leq d(A_s) + d(B'_t) + d(A_s \backslash B'_t) + d(B'_t \backslash A_s) < 4C,$$

which is a contradiction.

(b) Observe that $d(A_s \cap A'_s) \geq d(A_s)$, because otherwise A_s could have been replaced with $A_s \cap A'_s$ in separation (A_s, B_t). By submodularity of cuts we have that $d(A_s \cap A'_s) + d(A_s \cup A'_s) \leq d(A_s) + d(A'_s)$, and hence $d(A_s \cup A'_s) \leq d(A'_s) = d(A_s)$. By posimodularity, we have that

$$\begin{aligned} d\left((A_s \cup A'_s) \backslash B_t\right) + d\left(B_t \backslash (A_s \cup A'_s)\right) &\leq d(A_s \cup A'_s) + d(B_t) \\ &\leq d(A_s) + d(B_t) = 2C \end{aligned} \tag{8}$$

On the other hand, for terminal separation $((A_s \cup A'_s) \backslash B_t, B_t)$ we have that

$$d\left((A_s \cup A'_s) \backslash B_t\right) + d(B_t) = 2c\left((A_s \cup A'_s) \backslash B_t, B_t\right) \geq 2C, \tag{9}$$

and for terminal separation $(A_s \cup A'_s, B_t \backslash (A_s \cup A'_s))$ we have that

$$d(A_s \cup A'_s) + d\left(B_t \backslash (A_s \cup A'_s)\right) = 2c\left((A_s \cup A'_s, B_t) \backslash (A_s \cup A'_s)\right) \geq 2C. \tag{10}$$

Thus, from (8), (9), and (10)

$$4C \geq d\left((A_s \cup A'_s) \backslash B_t\right) + d(B_t) + d(A_s \cup A'_s) + d\left(B_t \backslash (A_s \cup A'_s)\right) \geq 4C,$$

which means that all the inequalities above are in fact equalities. In particular:

- $d(A_s \cap A'_s) = d(A_s) = d(A_s \cup A'_s)$, and
- $c((A_s \cup A'_s)\backslash B_t, B_t) = C$.

Symmetric arguments can be used to show that:

- $d(B_t \cap B'_t) = d(B_t) = d(B_t \cup B'_t)$,
- $c((A_s \cup A'_s)\backslash B'_t, B'_t) = C$,
- $c(A_s, (B_t \cup B'_t)\backslash A_s) = C$, and
- $c(A'_s, (B_t \cup B'_t)\backslash A'_s) = C$.

Therefore, both $(A_s \cap A'_s, B_t \cup B'_t)$ and $(A_s \cup A'_s, B_t \cap B'_t)$ have cost C.
(c) For the sake of contradiction, assume that $A_s \cup B_t \neq A'_s \cup B'_t$. Suppose first that there is an element $u \in A_s$ such that $u \notin A'_s \cup B'_t$. In the proof of (b) we have showed that $c((A_s \cup A'_s)\backslash B'_t, B'_t) = C$. Note that $((A_s \cup A'_s)\backslash B'_t, B'_t)$ is a terminal separation that extends (A'_s, B'_t), and moreover its left side is has at least one additional element u. Since its cost is the same as the cost of (A'_s, B'_t), we obtain a contradiction with the maximality of (A'_s, B'_t). □

Lemma 5.4 *Let $\mathcal{F}$ be the family of all maximal terminal separations (A_s, B_t) of minimum cost among separations that extend $(A^\circ \cup \{s\}, B^\circ \cup \{t\})$. Suppose that all separations from $\mathcal{F}$ resolve only the pair $\{s, t\}$ among the pairs from $\mathcal{T}'$. Then there exists a unique maximal terminal separation $(A_s^{\max}, B_t^{\min})$ such that $A_s^{\max} \supseteq A_s$ and $B_t^{\min} \subseteq B_t$ for each $(A_s, B_t) \in \mathcal{F}$. Moreover, if A is such that $A^\circ \cup \{s\} \subseteq A$, $A \cap B^\circ = \emptyset$, $A \cap \bigcup \mathcal{T}' \subseteq \{s\}$, but $A\backslash A_s^{\max} \neq \emptyset$, then $d(A) > d(A_s^{\max})$.*

Proof We set

$$\left(A_s^{\max}, B_t^{\min}\right) = \left(\bigcup_{(A_s, B_t) \in \mathcal{F}} A_s, \bigcap_{(A_s, B_t) \in \mathcal{F}} B_t \right).$$

From Lemma 5.3 it follows that $(A_s^{\max}, B_t^{\min}) \in \mathcal{F}$.

We are left with proving the last statement. Take any such A, and suppose for the sake of contradiction that $d(A) \leq d(A_s^{\max})$. Let $\overline{A} = A_s^{\max} \cup A$ and $\overline{B} = B_t^{\min}\backslash A$. Observe that $(\overline{A}, \overline{B})$ is a terminal separation that extends $(A^\circ \cup \{s\}, B^\circ \cup \{t\})$. Since $\overline{A}$ has at least one more element than $A_s^{\max}$, from the properties of $(A_s^{\max}, B_t^{\min})$ we infer that $c(\overline{A}, \overline{B}) > C$, where C is the cost of every separation from $\mathcal{F}$. Observe that $d(A_s^{\max} \cap A) \geq d(A_s^{\max})$, because otherwise we would substitute $A_s^{\max}$ with $A_s^{\max} \cap A$ in separation $(A_s^{\max}, B_t^{\min})$ and obtain a separation of smaller cost that extends $(A^\circ \cup \{s\}, B^\circ \cup \{t\})$. Hence, from the submodularity of cuts we infer that $d(\overline{A}) \leq d(A)$, so in particular $d(\overline{A}) \leq d(A_s^{\max})$.

Now, by posimodularity we obtain that

$$d\left(\overline{A}\backslash B_t^{\min}\right) + d\left(B_t^{\min}\backslash\overline{A}\right) \leq d(\overline{A}) + d\left(B_t^{\min}\right) \leq d\left(A_s^{\max}\right) + d\left(B_t^{\min}\right).$$

On the other hand, observe that $d(\overline{A}\backslash B_t^{\min}) \geq d(A_s^{\max})$, because otherwise we could substitute $A_s^{\max}$ with $\overline{A}\backslash B_t^{\min}$ in the terminal separation $(A_s^{\max}, B_t^{\min})$ and obtain a terminal separation that extends $(A^\circ \cup \{s\}, B^\circ \cup \{t\})$ and has strictly smaller cost. Thus we infer that $d(B_t^{\min} \setminus \overline{A}) \leq d(B_t^{\min})$. As $B_t^{\min} \setminus \overline{A} = B_t^{\min} \setminus A = \overline{B}$, we conclude that $d(\overline{A}) \leq d(A_s^{\max})$, $d(\overline{B}) \leq d(B_t^{\min})$, and hence $c(\overline{A}, \overline{B}) \leq C$. This is a contradiction. □

We modify now separation (A_s, B_t) as follows. For every terminal pair $\{s', t'\} \in \mathcal{T}'$ that is different from $\{s, t\}$, we verify using Theorem 2.4 whether (A_s, B_t) can be chosen so that it has a minimum possible cost among the separations that extend $(A^\circ \cup \{s\}, B^\circ \cup \{t\})$, but it also resolves $\{s', t'\}$. If this is possible, then we pursue Branching Step 1 with appropriate (A_s, B_t). Otherwise, every minimum-cost separation extending $(A^\circ \cup \{s\}, B^\circ \cup \{t\})$ resolves only $\{s, t\}$, and the assumptions of Lemma 5.4 are satisfied. Let $(A_s^{\max}, B_t^{\min})$ be the terminal extension whose existence is asserted by Lemma 5.4. Observe that we can construct $(A_s^{\max}, B_t^{\min})$ in time $\mathcal{O}(k^{\mathcal{O}(1)} m)$: we start with any (A_s, B_t) given by Theorem 2.4, and observe that Lemma 5.4 implies that $A_s^{\max}$ is the unique inclusion-wise maximal set containing A_s such that $E(A_s^{\max}, V(G) \setminus A_s^{\max})$ is a minimum cut between A_s and $B^\circ \cup (\mathcal{T} \setminus A_s)$; such a set can be computed using $\mathcal{O}(k)$ rounds of the Ford–Fulkerson algorithm.

Hence, we proceed further with the assumption that we have chosen (A_s, B_t) to be $(A_s^{\max}, B_t^{\min})$. We do symmetrically in the second branch, assuming that (A_t, B_s) is chosen to be $(A_t^{\min}, B_s^{\max})$, that is, the extension of B that contains terminal s is chosen to be maximum possible. Hence, by Lemma 5.4, we can from now on use the following assumption.

Assumption 2 For any set A with $A^\circ \subseteq A \subseteq V(G) \setminus B^\circ$ that contains only s from the terminals of $\mathcal{T}'$ and has at least one vertex outside A_s, it holds that $\Delta(A) > \Delta(A_s)$. Symmetrically, for any set B with $B^\circ \subseteq B \subseteq V(G) \setminus A^\circ$ that contains only s from the terminals of $\mathcal{T}'$ and has at least one vertex outside B_s, it holds that $\Delta(B) > \Delta(B_s)$.

5.1.2 Analyzing $A_s \cap B_s$, $A_s \setminus B_s$, and $B_s \setminus A_s$

Suppose now that for some pair $\{s, t\} \in \mathcal{T}'$, we have that $|(A_s \cap B_s) \setminus \{s\}| \geq 2$. Then, by Assumption 1 $Z = (A_s \cap B_s) \setminus \{s\}$ is a terminal-free set. Since pair $\{s, t\}$ has to be resolved one way or the other, then by persistence (Theorem 2.3) we infer that there is some minimum integral terminal separation (A^*, B^*) such that $Z \subseteq A^*$ or $Z \subseteq B^*$. Therefore, it is a safe reduction to merge Z into a single vertex.

Reduction step 2 For every $\{s, t\} \in \mathcal{T}'$, compute $Z_s = (A_s \cap B_s) \setminus \{s\}$ and $Z_t = (A_t \cap B_t) \setminus \{t\}$. Provided Z_s (Z_t) contains more than one vertex, merge it.

We apply this reduction to all terminal pairs from $\mathcal{T}'$, which takes time $\mathcal{O}(k^{\mathcal{O}(1)} m)$. Hence, using Lemma 5.1 from now on we can assume the following:

Assumption 3 For every pair $\{s, t\} \in \mathcal{T}'$, either $A_s \cap B_s = \{s\}$ or $A_s \cap B_s = \{s, s'\}$, where s' is the only neighbor of s. Moreover, either $A_t \cap B_t = \{t\}$ or $A_t \cap B_t = \{t, t'\}$, where t' is the only neighbor of t.

As every terminal has degree one, for a pair $\{s, t\} \in \mathcal{T}'$ we have that $d(A_s) \leq d(A^\circ \cup \{s\}) \leq d(A^\circ) + 1$, since otherwise replacing A_s with $A^\circ \cup \{s\}$ would decrease the cost of (A_s, B_s). On the other hand, we have that $d(A_s) \geq d(A^\circ)$, since otherwise $(A_s, B^\circ \cup \{t\})$ would be a terminal separation extending (A°, B°) of not larger cost, which would contradict the maximality of (A°, B°). Then, we have three possible cases for $(\Delta(A_s), \Delta(B_s))$: $(0, 0)$, $(1, 0)$ and $(1, 1)$; the omitted case $(0, 1)$ is symmetric to $(1, 0)$. The algorithm behaves differently in each of these cases. Before we proceed to the description of handling each case separately, we prove some useful observations first.

Let us now fix one pair $\{s, t\}$, and let $\tilde{A} = A_s \backslash B_s$ and $\tilde{B} = B_s \backslash A_s$. Observe that since branching on $\{s, t\}$ did not resolve any additional terminal pair, then both $\tilde{A} \backslash A^\circ$ and $\tilde{B} \backslash B^\circ$ are terminal-free. Hence, by the maximality of (A°, B°) we have that

$$d(\tilde{A}) \geq d(A^\circ) \quad \text{and} \quad d(\tilde{B}) \geq d(B^\circ), \tag{11}$$

and the equality holds if and only if $\tilde{A} = A^\circ$ or $\tilde{B} = B^\circ$, respectively. Let $R = V(G) \backslash (A_s \cup B_s)$.

Lemma 5.5 *One of the following two cases holds:*

- $|E(A_s \cap B_s, R)| = 1$, $\tilde{A} = A^\circ$, $\tilde{B} = B^\circ$, *and* $(\Delta(A_s), \Delta(B_s)) = (1, 1)$*; or*
- $|E(A_s \cap B_s, R)| = 0$, *and* $2 \geq \Delta(A_s) + \Delta(B_s) = \Delta(\tilde{A}) + \Delta(\tilde{B}) \geq 0$.

Proof By applying posimodularity of cuts to the sets A_s and B_s, we obtain:

$$\begin{aligned} d(A_s) + d(B_s) = d(\tilde{A}) + d(\tilde{B}) + 2|E(A_s \cap B_s, R)| &\geq d(A^\circ) + d(B^\circ) \\ &+ 2|E(A_s \cap B_s, R)|. \end{aligned} \tag{12}$$

On the other hand, we have that $d(A_s) \leq d(A^\circ) + 1$ and $d(B_s) \leq d(B^\circ) + 1$. Hence we have that $|E(A_s \cap B_s, R)| \leq 1$ and the claimed case distinction follows from (11) and (12). □

5.1.3 Decomposing Sets of Excess 2

Finally, we make a useful observation that will show a generic setting when Lemma 4.7 can be applied.

Lemma 5.6 *Suppose* $\Delta(A_s) = 1$ *and* $A_s \neq A^\circ \cup \{s\}$. *Then* $A_s \backslash \{s\} \supsetneq A^\circ$ *is a terminal-free excess-2 set, and* $(A_s \backslash A^\circ) \setminus \{s\}$ *has a decomposition* $\{d, c_1, c_2, \ldots, c_r\}$ *given by Lemma 4.7. Moreover,* $d = s'$ *is the unique neighbor of* s *in* G.

Proof The fact that $A_s \backslash \{s\}$ is an excess-2 set follows from the assumption that s has degree exactly 1 (due to the inapplicability of the Lonely Terminal Reduction), and its unique neighbor s' does not belong to $A^\circ \cup B^\circ$ and does belong to A_s (because $G[A_s \backslash A^\circ]$ is connected by Lemma 5.1). Since $(A_s \backslash A^\circ) \backslash \{s\}$ is nonempty and terminal-free (by Assumption 1), it follows from Lemma 4.7 that it has a decomposition of the form $\{c_1, c_2\}$ or $\{d, c_1, c_2, \ldots, c_r\}$, where c_i-s are pairwise nonadjacent and $A^\circ \cup \{c_i\}$ are excess-1 sets. Suppose $s' = c_i$ for some i. Then since $A^\circ \cup \{c_i\}$ is an excess-1 set, we would have that $A^\circ \cup \{c_i, s\}$ is an excess-0 set, and hence $(A^\circ \cup \{c_i, s\}, B_t)$

would be an extension of $(A^\circ \cup \{s\}, B^\circ \cup \{t\})$ of strictly smaller cost than (A_s, B_t), contradicting the definition of (A_s, B_t). Hence $(A_s \backslash A^\circ) \setminus \{s\}$ has a decomposition of the form $\{d, c_1, c_2, \ldots, c_r\}$ and $s' = d$. □

We will need one more lemma that resolves corner cases when we apply Lemma 5.6.

Lemma 5.7 *Suppose s satisfies the conditions of Lemma 5.6, and let $\{d = s', c_1, c_2, \ldots, c_r\}$ be the obtained decomposition of $(A_s \backslash A^\circ) \setminus \{s\}$. Let t' be the unique neighbor of t. Then $s' \neq t'$, and if $t' = c_i$ for some $i \in \{1, 2, \ldots, r\}$, then there exists an optimum integral terminal separation (A^*, B^*) that extends (A°, B°) and has $s \in B^*$ and $t \in A^*$.*

Proof The fact that $s' \neq t'$ follows from the inapplicability of the Common Neighbour Reduction. Suppose then that $t' = c_i$. From Lemma 4.7 it follows that for some $p_i \geq 1$ and $x_i \geq 0$, there are p_i edges between s' and c_i, $p_i + x_i$ edges between c_i and A°, and $x_i + 1$ edges between c_i and $V(G) \backslash A_s$; one of these $x_i + 1$ edges connects $c_i = t'$ with t.

Take any optimum integral separation (A^*, B^*) extending (A°, B°) and suppose that $s \in A^*$ and $t \in B^*$. We can further assume that $s' \in A^*$ and $t' = c_i \in B^*$, because otherwise switching the sides of s and t would result in an integral separation of not larger cost that already fulfills the property we aim for. Recall that c_i has p_i edges to s' (which is assigned to A^*), $p_i + x_i$ edges to A°, and $x_i + 1$ edges to other vertices of the graph. Since $p_i \geq 1$, we see that a strict majority of neighbors of c_i are in A^*. Hence switching the side of c_i from A^* to B^* strictly decreases the cost of the separation, a contradiction. □

Lemma 5.7 enables us to perform a reduction step whenever a corner case appears in the analysis of vertices close to s and t. We choose not to perform this reduction exhaustively, but rather to execute it on demand when such a case appears during branching.

5.1.4 Fixing an Edge ss' or tt'

In a few cases, we consider an improved branching set, when in one branch we fix $\{s', s\}$ to belong to the left part and t to belong to the right part, whereas in the second branch we fix vice versa. More precisely, we consider branches $(A_{ss' \to A}, B_{ss' \to A})$ and $(A_{ss' \to B}, B_{ss' \to B})$ that are minimum-cost terminal separations extending $(A^\circ \cup \{s, s'\}, B^\circ \cup \{t\})$ and $(A^\circ \cup \{t\}, B^\circ \cup \{s, s'\})$, computed using Theorem 2.4. Observe that there is some optimum solution that extends one of these branches: If in some optimum solution the vertices s' and s were assigned to different sides, then we could modify this solution by swapping the sides of s and t. After this modification then solution has no larger cost due to t having degree one, whereas the edge ss' ceases to be cut by the solution. This justifies the correctness of this branching step; we shall henceforth call it *branching on $\{s, t\}$ with fixing the edge ss'*. Symmetrically, we can define branching on $\{s, t\}$ with fixing the edge tt'.

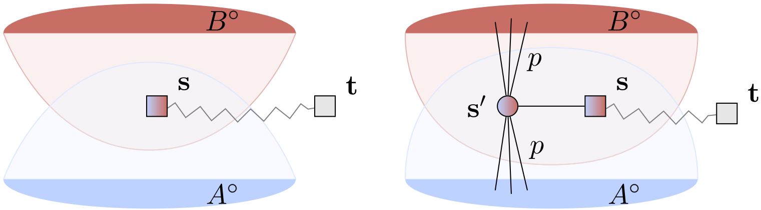

Fig. 3 Case $(\Delta(A_s), \Delta(B_s)) = (0, 0)$: a reduction is always immediately applicable. Terminal nodes are *squares*, paired with zig–zags. Extensions A_s and B_s are highlighted with *light blue* and *red*, respectively (Color figure online)

5.2 Case $(\boldsymbol{\Delta}(A_s), \boldsymbol{\Delta}(B_s)) = (\mathbf{0}, \mathbf{0})$

We show that this case in fact never happens. From Lemma 5.5 we infer that $E(A_s \cap B_s, R) = \emptyset$, $\tilde{A} = A^\circ$, and $\tilde{B} = B^\circ$. Hence, $A_s \backslash A^\circ = B_s \setminus B^\circ = A_s \cap B_s$. As we argued earlier, we can assume that $A_s \cap B_s = \{s\}$ or $A_s \cap B_s = \{s, s'\}$ for s' being the only neighbor of s'.

In the first case, since the degree of s is at most one, from $E(A_s \cap B_s, R) = \emptyset$ and $\Delta(A_s) = \Delta(B_s) = 0$ we can infer that s is an isolated terminal, which should have been removed by the Lonely Terminal Reduction. This contradicts the assumptions that no reduction rule is applicable.

In the second case, by $E(A_s \cap B_s, R) = \emptyset$ and $\Delta(A_s) = \Delta(B_s) = 0$, we infer that $|E(s', A^\circ)| = |E(s', B^\circ)| = x$ for some $x \geq 0$. If $x = 0$, then s' should have been reduced by the Pendant Reduction. On the other hand, if $x > 0$ then the Boundary Reduction would have been triggered on s'. In both cases this is a contradiction (Fig. 3).

5.3 Case $(\boldsymbol{\Delta}(A_s), \boldsymbol{\Delta}(B_s)) = (\mathbf{1}, \mathbf{0})$

From Lemma 5.5 we infer that $E(A_s \cap B_s, R) = \emptyset$ and $\Delta(\tilde{A}) + \Delta(\tilde{B}) = 1$. We have two subcases: either (a) $(\Delta(\tilde{A}), \Delta(\tilde{B})) = (1, 0)$, or (b) $(\Delta(\tilde{A}), \Delta(\tilde{B})) = (0, 1)$.

5.3.1 Subcase (a): $(\Delta(\tilde{A}), \Delta(\tilde{B})) = (1, 0)$

By the equality condition in (11) we have that $\tilde{B} = B^\circ$, while $\tilde{A} \supsetneq A^\circ$ is a terminal-free set of excess 1. By the inapplicability of the Excess-1 Reduction, we infer that $\tilde{A} = A^\circ \cup \{a\}$ for some nonterminal vertex a.

Set A_s satisfies the conditions of Lemma 5.6, so we can decompose $(A_s \backslash A^\circ) \setminus \{s\}$ into $\{d, c_1, c_2, \ldots, c_r\}$, where $d = s'$ is the unique neighbor of s. Since $\Delta(\tilde{A}) = 1$, we have that $B_s \supsetneq B^\circ \cup \{s\}$ and hence by Lemma 5.1 it follows that $s' \in B_s$. By Assumption 3 we infer that $A_s \cap B_s = \{s, s'\}$ and thus $\{a\} = \tilde{A} \backslash A^\circ = \{c_1, c_2, \ldots, c_r\}$. Therefore $r = 1$ and $c_1 = a$.

Since $\tilde{B} = B^\circ$, $B_s \backslash B^\circ = B_s \cap A_s = \{s, s'\}$.

By Lemma 4.7 we have that a has: p edges to s', $x + 1$ edges to $V(G) \backslash (A_s \cup B^\circ)$, $p + x$ edges to A° and no other edges, for some $p \geq 1$, $x \geq 0$. Since $B_s = B^\circ \cup \{s', s\}$ is an excess-0 set and $E(s', R) = \emptyset$, we have that $|E(s', B^\circ)| = p + |E(s', A^\circ)|$. In

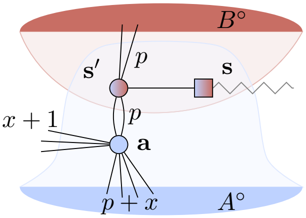

Fig. 4 Case (1,0)(a): $(\Delta(A_s), \Delta(B_s)) = (\Delta(\tilde{A}), \Delta(\tilde{B})) = (1, 0)$. Extensions A_s, B_s are highlighted

particular $|E(s', B^\circ)| > 0$, so since Boundary Reductions do not apply to s', we have $E(s', A^\circ) = \emptyset$ and hence $|E(s', B^\circ)| = p$ (Fig. 4).

Consider now case $x = 0$. Then a has a unique edge aa' with $a' \in R$. Consider first the case when a' is a terminal, so in particular aa' is the only edge incident to a'. If $a' = t$, then it is easy to see that $(A^\circ \cup \{a, t\}, B^\circ \cup \{s', s\})$ would be an extension of (A°, B°) of the same cost, which contradicts the maximality of (A°, B°). However, if a' belonged to some other pair $\{a', a''\} \in \mathcal{T}'$, then terminal separation $(A_s \cup \{a'\}, B_t \cup \{a''\})$ would have the same cost as (A_s, B_t), which contradicts the maximality of (A_s, B_t). In either case we obtain a contradiction, which means that a' is a nonterminal.

We claim that it is a safe reduction to contract the edge aa'; to prove this claim, it suffices to show that there exists an optimum integral terminal separation extending (A°, B°) where a and a' belong to the same side. Take any such integral terminal separation (A^*, B^*), and assume that a and a' are on opposite sides. Clearly it cannot happen that $a \in B^*$ and $a' \in A^*$, because then moving a from B^* to A^* would decrease the cost of the separation. Hence $a \in A^*$ and $a' \in B^*$. If $s' \in B^*$, then moving a from A^* to B^* would decrease the cost of the separation, so also $s' \in A^*$. Construct a new integral separation (A^*_m, B^*_m) from (A^*, B^*) by moving $\{a, s'\}$ from A^* to B^*. Then the cost of (A^*_m, B^*_m) is not larger than that of (A^*, B^*) (we could have broken the edge $s's$ instead of aa'), while both endpoints of aa' belong to A^*_m.

This reasoning proves the correctness of the following step.

Reduction step 3 Suppose $x = 0$ and let a' be the unique neighbor of a in R; then a' is a non-terminal. Merge a with a' and restart.

Henceforth we assume that $x > 0$. We claim that now branching on the membership of a leads to a good branch. More precisely, we perform the following branching.

Branching step 4 If $x \geq 1$, recurse into two branches $(A_{a\to A}, B_{a\to A})$ and $(A_{a\to B}, B_{a\to B})$ that are minimum-cost maximal terminal separations extending $(A^\circ \cup \{a\}, B^\circ)$ and $(A^\circ, B^\circ \cup \{a\})$, respectively.

Of course, $(A_{a\to A}, B_{a\to A})$ and $(A_{a\to B}, B_{a\to B})$ are computed using the algorithm of Theorem 2.4 in time $\mathcal{O}(k^{\mathcal{O}(1)}m)$. We are left with proving that after applying all the immediate reductions in each branch, we arrive at a good branching vector. For $X \in \{A, B\}$, let $t_{a\to X}$, $\nu_{a\to X}$, $k_{a\to X}$ be the changes of the components of the potential in respective branches, as we denote them in branching vectors.

Consider first the branch $(A_{a\to A}, B_{a\to A})$. Then p Boundary Reductions are triggered on vertex s' (regardless of whether it is added or not to one of the sets

$A_{a\to A}, B_{a\to A})$. Hence $k_{a\to A} \geq p$. Moreover, the terminal pair $\{s, t\}$ either is already resolved by $(A_{a\to A}, B_{a\to A})$ or gets reduced by the Lonely Terminal Reduction after applying the Boundary Reductions. Hence $t_{a\to A} \geq 1$. Finally, since (A°, B°) was maximal, we have that $\nu_{a\to A} \geq 1$. So the part of the branching vector corresponding to the branch $(A_{a\to A}, B_{a\to A})$ is $[1, 1, p]$, or better.

Consider now the second branch $(A_{a\to B}, B_{a\to B})$. Then at least $|E(a, A^\circ)| = p+x$ Boundary Reductions are triggered, hence $k_{a\to B} \geq p + x$. Since $p \geq 1$ and t is of degree 1, $s' \in B_{a\to B}$ and without loss of generality we can assume $s \in B_{a\to B}$ and $t \in A_{a\to B}$. Hence $t_{a\to B} \geq 1$. If actually $t_{a\to A} \geq 2$ or $t_{a\to B} \geq 2$, then we arrive at a branching vector $[1, 1, p; 2, 1, p]$ or better, which is good, so assume that $t_{a\to A} = t_{a\to B} = 1$, that is, only the pair $\{s, t\}$ gets resolved.

We now claim that $\Delta(A_{a\to B}) \geq 1$ and $\Delta(B_{a\to B}) \geq 1$. The latter claim follows from Assumption 2, since then $B_{a\to B}$ contains only s among the terminals (due to $t_{a\to B} = 1$) and $a \in B_{a\to B}\setminus B_s$. For the former claim, suppose for the sake of contradiction that $d(A_{a\to B}) = d(A^\circ)$. Recall that also $d(B_s) = d(B^\circ)$, which means that $d(A_{a\to B}) + d(B_s) = c(A^\circ, B^\circ)$. From the posimodularity of cuts it now follows that one of the terminal separations $(A_{a\to B}\setminus B_s, B_s)$ and $(A_{a\to B}, B_s\setminus A_{a\to B})$ has cost not larger than (A°, B°), while both of them resolve the terminal pair $\{s, t\}$. This is a contradiction with the maximality of (A°, B°). Hence we infer that $\Delta(A_{a\to B}) \geq 1$ and $\Delta(B_{a\to B}) \geq 1$, and so $\nu_{a\to B} \geq 2$.

Thus, branching into separations $(A_{a\to A}, B_{a\to A})$ and $(A_{a\to B}, B_{a\to B})$ leads to a branching vector $[1, 1, p; 1, 2, p + x]$ or better. Recalling that $p, x > 0$, observe that this branching vector can be not good only if $p = x = 1$ and $\Delta(B_{a\to B}) = 1$. Hence, from now on let us analyze this case.

Since $\Delta(B_{a\to B}) = 1$, we have that $B_{a\to B}\setminus\{s\}$ is a terminal-free set of excess 2, and hence we can apply Lemma 4.7 to it: We have that $B_{a\to B}\setminus\{s\}$ has a decomposition of the form $\{c_1, c_2\}$ or $\{d, c_1, \ldots, c_r\}$. Note that $B^\circ \cup \{s'\}$ is an excess-1 set, so $s' = c_i$ for some i. As $a \in B_{a\to B}$, a is adjacent to s', and c_i-s are pairwise non-adjacent, we must have that $a = d$ and we are dealing with a decomposition of the form $\{d, c_1, \ldots, c_r\}$. Observe that $B^\circ \cup \{a, s'\}$ is a B°-extension of excess at least $1 + x + 1 = 3$; hence $B_{a\to B} \supsetneq B^\circ \cup \{a, s', s\}$, and in particular $r > 1$. Hence there exists some vertex $c_j \neq c_i = s'$. By Lemma 4.7 we have that c_j is adjacent both to B° and to a. Hence, in the branch $(A_{a\to A}, B_{a\to A})$ at least one Boundary Reduction is applied to c_j, regardless whether c_j is assigned to $A_{a\to A}$, or $B_{a\to A}$, or neither of these sets. We did not include this Boundary Reduction in the previous calculations; this shows that we in fact pursue a branch with a branching vector $[1, 1, 2; 1, 2, 2]$ or better, which is a good branching vector.

5.3.2 Subcase (b): $(\Delta(\tilde{A}), \Delta(\tilde{B})) = (0, 1)$

By the equality condition in (11) we have that $\tilde{A} = A^\circ$, while $\tilde{B} \supsetneq B^\circ$ is a terminal-free set of excess 1. By the inapplicability of the Excess-1 Reduction, we infer that $\tilde{B} = B^\circ \cup \{b\}$ for some nonterminal vertex b. In particular $B_s \supsetneq B^\circ \cup \{s\}$, so by Lemma 5.1 the unique neighbor s' of s belongs to B_s. Since $\Delta(B_s) = 0$, we have that $B_s\setminus\{s\}$ is a terminal-free set of excess 1, so it consists of a single vertex. However, this set already contains b. Hence we infer that $b = s'$ is the unique neighbor of s,

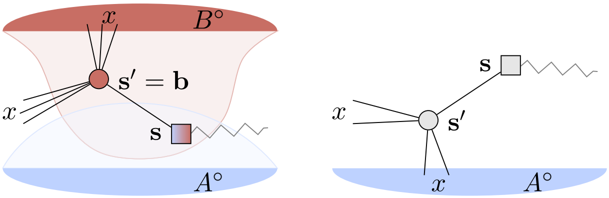

Fig. 5 Case (1,0)(b): $(\Delta(\tilde{A}), \Delta(\tilde{B})) = (0, 1)$. This gives rise to an *antenna*, which has to be analyzed together with the other terminal. The *right side* shows an antenna with natural side A°; note the definition does not mention A_s, B_s, it only relies on the behaviour of extensions containing s or s'

$\tilde{B} = \{s'\} \cup B^\circ$, $B_s = \{s, s'\} \cup B^\circ$. In particular $s' \notin A_s$, so by Lemma 5.1 it follows that $A_s = A^\circ \cup \{s\}$.

Let $x = |E(s', B^\circ)|$. Since $\Delta(B_s) = 0$, we also have $x = |E(s', V(G) \backslash B_s)|$. If $x = 0$ then s' would be only adjacent to s and thus reducible by the Pendant Reduction. Hence, $x > 0$. In particular, we infer that $E(s', A^\circ) = \emptyset$, since otherwise the Boundary Reduction could be applied to s'.

Let us now examine two possible branching steps. Firstly, consider just branching into two branches (A_s, B_t) and (A_t, B_s). In both cases, only one terminal pair $\{s, t\}$ gets resolved. In branch (A_t, B_s), when s is assigned to B, we pessimistically have no Boundary Reduction and no increase in the cost of the separation. In branch (A_s, B_t), however, when s is assigned to A, we have that $\Delta(A_s) = 1$ and one Boundary Reduction is triggered on vertex s' due to having both an edge to s and to B°.

We now investigate the components of the branching vector when branching on $\{s, t\}$ with fixing ss'. If in one of the branches at least one more terminal pair gets resolved, then as argued in the beginning of this section we can just pursue the branching step, because it leads to a good branching vector. Hence, assume from now on that in both branches only the pair $\{s, t\}$ gets resolved. Since $A_s = \{s\} \cup A^\circ$ and $A_{ss' \to A}$ contains only s among the terminals of $\mathcal{T}'$, by Assumption 2 we have that $\Delta(A_{ss' \to A}) \geq 2$. Also, at least one Boundary Reduction is triggered on an edge between s' and B°. In branch $(A_{ss' \to B}, B_{ss' \to B})$, again we pessimistically have no Boundary Reduction and no increase in the cost of the separation (Fig. 5).

A terminal s with the behaviour as described above will be actually the most problematic case for our branching algorithm. Let us define this setting formally.

Definition 5.8 A terminal s is called an *antenna* if the following conditions hold:

- The only neighbor s' of s is a nonterminal, has $x > 0$ edges to one of the sets A° or B°, no edge to the second one, and x edges to $V(G) \backslash (A^\circ \cup B^\circ \cup \{s\})$. The side $S \in \{A^\circ, B^\circ\}$ to which s' is adjacent is called the *natural* side of s, and the second one is called the *unnatural* side of s.
- Let S and $\overline{S}$ be the natural and unnatural side of s, respectively. Then
  - For any X with $S \cup \{s, s'\} \subsetneq X \subseteq V(G) \backslash \overline{S}$ that contains only s among the terminals from $\mathcal{T}'$, it holds that $\Delta(X) \geq 1$.
  - For any Y with $\overline{S} \cup \{s\} \subseteq Y \subseteq V(G) \backslash S$ that contains only s among the terminals from $\mathcal{T}'$, it holds that $\Delta(Y) \geq 1$. If moreover Y contains at least one more vertex than $\overline{S} \cup \{s\}$, then $\Delta(Y) \geq 2$.

The discussion above together with Lemmas 5.2 and Assumption 2 shows that in this case s is an antenna with natural side B°. Obviously, in the symmetric subcase when $(\Delta(A_s), \Delta(B_s)) = (0, 1)$ and $(\Delta(\tilde{A}), \Delta(\tilde{B})) = (1, 0)$ we obtain that s is an antenna with natural side A°.

The idea now is *not* to perform any branching step on an antenna, but rather to branch on the situation around the second terminal t, i.e., swap the roles of t and s and restart the analysis. In other words, we will show that if the analysis of the second terminal t does not reveal that it is an antenna [it conforms to cases (0, 0), $(1, 0)a$, or (1, 1)], then a branching step leading to a good branching vector can be found on that side. We will be thus left with the case when both s and t are antennas, which we aim to resolve now by exposing a branching strategy leading to a good branching vector.

Therefore, assume that s and t are both antennas, and let s' and t' be their unique neighbors, respectively. By the inapplicability of the Common Neighbor Reduction, $s' \neq t'$. First, suppose that s and t have different natural sides, say s has natural side A° and t has natural side B°. However, then $(A^\circ \cup \{s, s'\}, B^\circ \cup \{t, t'\})$ would be a terminal separation that has the same cost as (A°, B°), which contradicts the maximality of (A°, B°).

Hence, assume that s and t have the same natural side. W.l.o.g. suppose that it is B°. Let $x = |E(s', B^\circ)|$ and $y = |E(t', B^\circ)|$; recall that $x, y \geq 1$. Consider two possible branching steps: we can branch on $\{s, t\}$ with fixing ss' or with fixing tt'. Consider first fixing edge ss', and let $(A_{ss'\to A}, B_{ss'\to A})$ and $(A_{ss'\to B}, B_{ss'\to B})$ be the branches. By the definition of the antenna we have that $\Delta(A_{ss'\to A}) \geq 2$ and $\Delta(A_{ss'\to B}) \geq 1$. Also, in branch $(A_{ss'\to A}, B_{ss'\to A})$ we have at least x Boundary Reductions triggered on edges incident to s', whereas in branch $(A_{ss'\to B}, B_{ss'\to B})$ we have at least one Boundary Reduction triggered edges incident to t'. Thus we obtain a branching vector $[1, 2, x; 1, 1, 1]$ or better, and a symmetric reasoning for fixing tt' leads to branching vector $[1, 1, 1; 1, 2, y]$, or better. Note that one of these vectors is good if $\max(x, y) \geq 3$. Furthermore, such a branching also leads to a good vector if s' or t' is adjacent to some terminal other than s or t, respectively, as then in at least one branch a second terminal pair would be resolved. Hence, if this is the case, we pursue the respective branching step.

Branching step 5 If $\max(x, y) \geq 3$, or there is a terminal in $\mathcal{T}'$ different than s or t adjacent to s' or t', then pursue branching on $\{s, t\}$ with fixing the respective edge ss' or tt'.

From now on we assume that $x, y \leq 2$ and that no other terminal than s and t is adjacent to s' nor t' (Fig. 6).

Consider now the case when there is a vertex a such that all edges of $E(s', V(G)\setminus(B^\circ \cup \{s\}))$ have a as the endpoint different than s'. This encompasses the cases when $x = 1$ and when $x = 2$ but the considered edges connecting s' with $V(G)\setminus(A^\circ \cup \{s\})$ have the same second endpoint. We claim that then it is a safe reduction to merge a and s'. To prove this claim, we need to show that there exists an optimum integral terminal separation (A^*, B^*) extending (A°, B°) where a and s' are on the same side. Take any such integral terminal separation (A^*, B^*), and assume that a and s' are on opposite sides. Clearly it cannot happen that $a \in B^*$ and $s' \in A^*$, because then moving s' from A^* to B^* would decrease the cost of the separation.

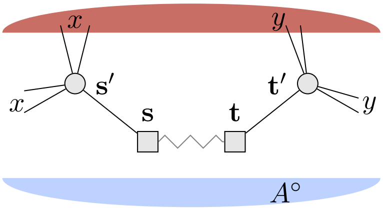

Fig. 6 Case (1,0)(b) on both s and t, where furthermore the antennas have the same natural side

Hence $a \in A^*$ and $s' \in B^*$. This implies that $s \in B^*$, since otherwise we could improve the cost of the separation by moving s' from B^* to A^*. Therefore $t \in A^*$. Consider modifying (A^*, B^*) into (A^*_m, B^*_m) by

- moving s' and s from B^* to A^*, and
- moving t and t' from A^* to B^*, provided t' was not already included in B^*.

It is easy to see that (A^*_m, B^*_m) is still an integral terminal separation extending (A°, B°) and its cost is no larger than that of (A^*, B^*). Hence, it is optimum as well. However, in (A^*_m, B^*_m) it holds that s' and a are on the same side.

This reasoning and its symmetric version for t' imply the correctness of the following reduction step. Note that a is not a terminal, as we have already excluded this case in the previous branching step.

Reduction step 6 If $|N(s') \setminus (B^\circ \cup \{s\})| = 1$, then merge s' with its unique neighbor in $V(G) \setminus (B^\circ \cup \{s\})$ and restart. If $|N(t') \setminus (B^\circ \cup \{t\})| = 1$, then merge t' with its unique neighbor in $V(G) \setminus (B^\circ \cup \{t\})$ and restart.

We are left with the case when $x = y = 2$ and both s' and t' have two neighbors outside $B^\circ \cup \{s, t\}$; these neighbors will be called *external*. We claim that then just pursuing branching on $\{s, t\}$ with fixed ss' leads to a good branching vector.

Branching step 7 If $x = y = 2$ and $|N(s') \setminus (B^\circ \cup \{s\})| = |N(t') \setminus (B^\circ \cup \{t\})| = 2$, then pursue branching on $\{s, t\}$ with fixing ss'.

Let the branches be $(A_{ss' \to A}, B_{ss' \to A})$ and $(A_{ss' \to B}, B_{ss' \to B})$. Recall that $\Delta(A_{ss' \to A}) \geq 2$. If actually $\Delta(A_{ss' \to A}) \geq 3$, then we would already have a good branching vector $[1, 3, 2; 1, 1, 1]$ or better, so assume henceforth that $\Delta(A_{ss' \to A}) = 2$. Consider now set $A' = A_{ss' \to A} \setminus \{s, s'\}$. If A' did not contain both external neighbors of s', then a simple edge count shows that A' would be a terminal-free set of excess at most -2 (if it contains no external neighbor of s') or 0 (if it contains one external neighbor of s'). In both cases this is a contradiction with the maximality of (A°, B°). Hence, A' contains both external neighbors of s', and A' is a terminal-free set of excess 2. By Lemma 4.7, we can decompose A' as $\{c_1, c_2\}$ or $\{d, c_1, \ldots, c_r\}$. Since s' has two different neighbors in A', at least one of them is c_i for some i. However, by Lemma 4.7 each c_i is adjacent to A°, and hence in branch $(A_{ss' \to B}, B_{ss' \to B})$ at least one Boundary Reduction is triggered on vertex c_i (regardless whether this vertex is assigned to $A_{ss' \to B}$, or to $B_{ss' \to B}$, or to neither of these sets). In our earlier calculations we did not account for this Boundary Reduction, so in fact we obtain branching vector $[1, 2, 2; 1, 1, 2]$ or better, which is a good branching vector.

5.4 Case $(\Delta(A_s), \Delta(B_s)) = (1, 1)$

By Lemma 5.5, we have three non-symmetric subcases:

(a) $|E(A_s \cap B_s, R)| = 1$, $\tilde{A} = A^\circ$, $\tilde{B} = B^\circ$;
(b) $E(A_s \cap B_s, R) = \emptyset$, $\Delta(\tilde{A}) = \Delta(\tilde{B}) = 1$;
(c) $E(A_s \cap B_s, R) = \emptyset$, $\Delta(\tilde{A}) = 0$, $\Delta(\tilde{B}) = 2$.

The case when $E(A_s \cap B_s, R) = \emptyset$, $\Delta(\tilde{A}) = 2$, $\Delta(\tilde{B}) = 0$, is symmetric to case (c).

The algorithm proceeds as follows: It investigates every terminal pair $\{s, t\} \in \mathcal{T}'$, and investigates the case given by this terminal pair when considered as $\{s, t\}$ (i.e., looking from the side of s), and when considered as $\{t, s\}$ (i.e., looking from the side of t). If in any of these checks, for any terminal pair, case $(0, 0)$ or $(1, 0)(a)$ is discovered, the algorithm pursues the respective Reduction Step or Branching Step, as described in the previous sections. Otherwise, we can assume the following:

Assumption 4 Every terminal of $\mathcal{T}'$ is either an antenna, or investigating the basic branch of the respective terminal pair from its side yields case $(1, 1)$ (has *type (1,1)*).

In the following we will use this property heavily in order to be able to reason about the total increase in the cost of the separation, also on the side of the second terminal from the pair we are currently investigating.

5.4.1 Case (a): $|E(A_s \cap B_s, R)| = 1$, $\tilde{A} = A^\circ$, $\tilde{B} = B^\circ$

Let $Z = A_s \cap B_s = A_s \backslash A^\circ = B_s \setminus B^\circ$. By Assumption 3, we have that $Z = \{s\}$ or $Z = \{s, s'\}$, where s' is the unique neighbor of s.

Suppose first that $Z = \{s, s'\}$. Let $x = |E(s', A^\circ)|$ and $y = |E(s', B^\circ)|$. Since $|E(s', R)| = |E(Z, R)| = 1$ and both A_s and B_s are excess-1 sets, we infer that $x = y$. Consequently it must hold that $x = y = 0$, because otherwise the Boundary Reduction would apply to s'. Thus, s' is a vertex of degree 2 with one neighbor r in R and the second being s. Then the Pendant Reduction would apply to $X = \{s'\}$, a contradiction.

Therefore, we have that $Z = \{s\}$. Let s' be the unique neighbor of s. We pursue branching on the pair $\{s, t\}$ with fixing edge ss', i.e., branch into two subcases $(A_{ss' \to A}, B_{ss' \to A})$ and $(A_{ss' \to B}, B_{ss' \to B})$ that are minimum-cost terminal separations extending $(A^\circ \cup \{s, s'\}, B^\circ \cup \{t\})$ and $(A^\circ \cup \{t\}, B^\circ \cup \{s, s'\})$, respectively.

Branching step 8 Pursue branching on $\{s, t\}$ with fixing ss'.

Obviously, as explained in the beginning of this section, if any of the resulting branches resolves one more terminal pair, then the branching vector is good. Therefore, suppose that in both branches only the pair $\{s, t\}$ gets resolved. By Assumption 2, we have that $\Delta(A_{ss' \to A}) \geq 2$ and $\Delta(B_{ss' \to B}) \geq 2$. By Assumption 4, terminal t is either of type $(1, 1)$ or is an antenna. In the former case, by Lemma 5.2 we have that $\Delta(B_{ss' \to A}) \geq 1$ and $\Delta(A_{ss' \to B}) \geq 1$. Hence we arrive at branching vector $[1, 3, 0; 1, 3, 0]$ or better, which is a good branching vector. In the latter case, by the definition of an antenna we have that $\Delta(B_{ss' \to A}) \geq 1$ or $\Delta(A_{ss' \to B}) \geq 1$, depending

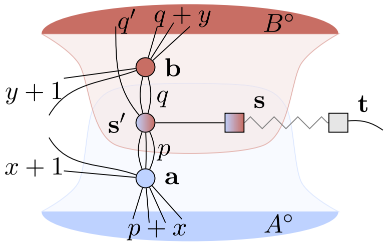

Fig. 7 Case (1,1)(b): $\Delta(B_s) = \Delta(A_s) = \Delta(\tilde{A}) = \Delta(\tilde{B}) = 1$. Note the edges counted in $x + 1$ and $y + 1$ may both include a common edge between a and b

on whether A° or B° is natural for s. Also, in the same branch where respective inequality holds, one Boundary Reduction gets applied on the unique neighbor of t. Thus we arrive at branching vector $[1, 2, 0; 1, 3, 1]$, or $[1, 3, 1; 1, 2, 0]$ or better (depending on which side is natural for t), which is a good branching vector.

5.4.2 Case (b): $E(A_s \cap B_s, R) = \emptyset$, $\Delta(\tilde{A}) = \Delta(\tilde{B}) = 1$

Since $\tilde{A}$ and $\tilde{B}$ are terminal-free sets of excess 1, by the inapplicability of the Excess-1 Reduction we infer that $\tilde{A} = A^\circ \cup \{a\}$ and $\tilde{B} = B^\circ \cup \{b\}$ for some distinct nonterminal vertices a, b. In particular, both $A_s \backslash A^\circ$ and $B_s \backslash B^\circ$ contain at least one more vertex than s. Hence, by Lemma 5.1 we infer that if s' is the unique neighbor of s, then $s' \in A_s$ and $s' \in B_s$. From Assumption 3 it follows that $A_s \cap B_s = \{s, s'\}$. Hence $A_s \backslash A^\circ = \{a, s, s'\}$ and $B_s \backslash B^\circ = \{b, s, s'\}$.

Since $\Delta(A_s) = 1$ and $A_s \neq A^\circ \cup \{s\}$, we can apply Lemma 5.6 to it and infer that $A_s \backslash \{s\}$ has a decomposition $\{d, c_1\}$ with $s' = d$ and $c_1 = a$. Consequently, by Lemma 4.7 we infer that for some $p \geq 1$ and $x \geq 0$, we have $|E(s', a)| = p$, $|E(a, A^\circ)| = p + x$, and $|E(a, V(G) \backslash A_s)| = x + 1$. Also, there is no edge between a and B°, because then the Boundary Reduction would be applicable to a. A symmetric reasoning shows that for some $q \geq 1$ and $y \geq 0$, we have $|E(s', b)| = q$, $|E(b, B^\circ)| = q + y$, $|E(b, V(G) \backslash B_s)| = y + 1$, and there is no edge between b and A°.

Vertex s' cannot be connected both to A° and to B°, because then the Boundary Reduction would be applicable to it. Hence, w.l.o.g. assume that $E(s', A^\circ) = \emptyset$. Let $q' = |E(s', B^\circ)|$. Since $E(A_s \cap B_s, R) = \emptyset$ and both A_s and $A^\circ \cup \{a\}$ are sets of excess 1, we infer that $p = q + q'$ (Fig. 7).

For the sake of further argumentation, we now resolve the case when $t' = a$ or $t' = b$, where t' is the unique neighbor of t. Then, Lemma 5.7 and its symmetric variant imply that the pair $\{s, t\}$ can be assigned greedily. More precisely, the following reduction step is correct.

Reduction step 9 If $t' = a$ then assign s to the B-side and t to the A-side, i.e., proceed with instance (A_t, B_s). If $t' = b$ then assign s to the A-side and t to the B-side, i.e., proceed with instance (A_s, B_t).

Henceforth we assume that $t' \neq a$ and $t' \neq b$. Since $A_s = \{a, s, s'\}$ and $B_s = \{b, s, s'\}$, by the inpplicability of Common Neighbor Reduction we infer that $t' \notin A_s$ and $t' \notin B_s$.

The crucial observation now is that we can fix both edges ss' and tt' at the same time.

Lemma 5.9 *There exists an optimum integral terminal separation* (A^*, B^*) *where either* $\{s, s'\} \subseteq A^*$ *and* $\{t, t'\} \subseteq B^*$*, or* $\{s, s'\} \subseteq B^*$ *and* $\{t, t'\} \subseteq A^*$.

Proof Let us take any optimum integral terminal separation (A^*, B^*). If the condition of the lemma is not satisfied, then swapping the sides of s and t does not change the cost of the separation. Let us then assume that the edge tt' is not cut in the solution. Hence, without loss of generality we assume that $s', t, t' \in B^*$ and $s \in A^*$; the rest of the reasoning will be independent of the choice we made earlier that there are no edges between s and A°, so we are indeed not losing generality here.

Consider $A = A^* \cup A_s$. By the submodularity of cuts we have that

$$d(A^* \cap A_s) + d(A) \leq d(A^*) + d(A_s).$$

However, we have that $d(A_s) \leq d(A^* \cap A_s)$ because otherwise we would be able to replace A_s with $A^* \cap A_s$ in separation (A_s, B_t) thus decreasing its cost while preserving the fact that it extends $(A^\circ \cup \{s\}, B^\circ \cup \{t\})$. Hence, we infer that $d(A^*) \geq d(A)$. Observe that since $A_s \setminus (A^\circ \cup \{s\})$ is terminal-free, then $(A, V(G)\setminus A)$ is also an integral terminal separation, and its cost is $d(A) \leq d(A^*) = c(A^*, B^*)$. Hence, $(A, V(G)\setminus A)$ is also an optimum integral terminal separation. Since $s' \in A_s$ and $t' \notin A_s$, we infer that edges ss' and tt' are not cut in $(A, V(G)\setminus A)$, as was requested. □

Lemma 5.9 justifies the correctness of branching on $\{s, t\}$ with both ss' and tt' fixed. More precisely, we branch into separations $(A_{ss' \to A}, B_{ss' \to A})$ and $(A_{ss' \to B}, B_{ss' \to B})$ that are minimum-cost terminal separations extending $(A^\circ \cup \{s, s'\}, B^\circ \cup \{t, t\})$ and $(A^\circ \cup \{t, t'\}, B^\circ \cup \{s, s'\})$, computed using Theorem 2.4.

Branching step 10 Pursue branching on $\{s, t\}$ with fixing both ss' and tt'.

As we argued at the beginning of this section, if in any of these branches at least one more terminal pair gets resolved, then we arrive at a good branching vector; hence assume that this is not the case.

By Lemma 5.2 we have that $\Delta(A_{ss' \to A}) \geq 1$ and $\Delta(B_{ss' \to B}) \geq 1$. Also, in both branches the Boundary Reduction will be applied at least p times: either p times on a (provided s' is assigned to the B-side), or q times on b and q' times on edges between s' and B° (provided s' is assigned to the A-side).

We now calculate the branching vectors when performing this branching.

Suppose first that t is an antenna, then by the definition of the antenna we have that $\Delta(A_{ss' \to B}) \geq 2$ or $\Delta(B_{ss' \to A}) \geq 2$, depending whether B° or A° is the natural side of t. Moreover, in the same branch, one Boundary Reduction is triggered on an edge between t' and the natural side of t in the branch. By $t' \notin A_s \cup B_s$, we know that this Boundary Reduction was not accounted for in the previous calculations. Hence, we obtain branching vector $[1, 1, p; 1, 3, p + 1]$, or $[1, 3, p + 1; 1, 1, p]$, or better, depending on the natural side of t. Since $p \geq 1$, these branching vectors are good.

Suppose now that t is of type $(1, 1)$, and moreover that investigation of its situation also leads to the same case (b). Then we have that $\Delta(B_{ss' \to A}) \geq 1$ and $\Delta(A_{ss' \to B}) \geq 1$

by Lemma 5.2. Moreover, in both of the branches, at least one Boundary Reduction is triggered that reduces some edge incident to t'. It is easy to see that the applicability of this Boundary Reduction could not be spoiled by the application of the p Boundary Reductions on the side of s, because $t' \notin A_s \cup B_s$ and s' and t' are assigned to different sides. Thus, we arrive at branching vector $[1, 2, p+1; 1, 2, p+1]$, which is $[1, 2, 2; 1, 2, 2]$ or better, and hence good.

Finally, we are left with the case when t is of type $(1, 1)$, and the investigation of its situation also leads to case (c). Similarly as in the previous paragraph, we have that $\Delta(B_{ss' \to A}) \geq 1$ and $\Delta(A_{ss' \to B}) \geq 1$. Moreover, as we shall see in the next section, in at least one branch, one additional Boundary Reduction will be triggered that will reduce an edge incident to t'. Moreover, the applicability of this Boundary Reduction will not be spoiled by the application of the previous p Boundary Reductions on the side of s, for the same reason as in the previous paragraph; that is, $t' \notin A_s \cup B_s$ and s' and t' are assigned to different sides. Hence we arrive at branching vector $[1, 2, p+1; 1, 2, p]$, or $[1, 2, p; 1, 2, p+1]$, or better. All these vectors are good for $p > 0$.

5.4.3 Case (c): $E(A_s \cap B_s, R) = \emptyset$, $\tilde{A} = A^\circ$, $\Delta(\tilde{B}) = 2$

Since $\Delta(B_s) = 1$, we have that $B_s \backslash \{s\}$ is a terminal-free extension of B° of excess 2 and we can apply Lemma 5.6 to decompose it as $\{d, c_1, c_2, \ldots, c_r\}$, where $d = s'$ is the unique neighbor of s. Let $p_i = |E(c_i, s')|$, for $i = 1, 2, \ldots, r$. Recalling Lemma 4.10, let $\sigma = |E(s', B^\circ)| + \sum_{i=1}^{r} p_i = |E(s', \{c_1, \ldots, c_r\} \cup B^\circ)|$ be the number of Boundary Reductions that are immediately triggered within $B_s \backslash \{s\}$ in any branch when s' is assigned to the A-side. By Lemma 4.10 we have that $\sigma > 0$. This justifies the claim that was left in our analysis of Case $(1, 1)b$, where we argued for the applicability of one additional Boundary Reduction.

Before we proceed, let us exclude the corner case when $t' = c_i$ for some $i \in \{1, 2, \ldots, r\}$, where t' is the unique neighbor of t. Lemma 5.7 justifies the correctness of the following reduction step.

Reduction step 11 If $t' = c_i$ for some $i \in \{1, 2, \ldots, r\}$, then assign s to the A-side and t to the B-side, i.e., proceed with instance (A_s, B_t).

Since $t' \neq s'$ by the inapplicability of the Common Neighbor Reduction, henceforth we can assume that $t' \notin B_s$.

Since $\tilde{A} = A^\circ$, by Assumption 3 we have two cases: either $A_s \backslash A^\circ = \{s, s'\}$ or $A_s \backslash A^\circ = \{s\}$.

Subcase (c.i) $A_s \backslash A^\circ = \{s, s'\}$. Let $p = |E(s', A^\circ)|$. Since $E(A_s \cap B_s, R) = \emptyset$ and A_s has excess 1, we infer that there are $p + 1$ edges from s' to $\tilde{B} = \{c_1, c_2, \ldots, c_r\} \cup B^\circ$, and hence $\sigma = p + 1$. Observe that $p \geq 1$, because if $p = 0$ the s' would be adjacent only to s and to a vertex in $\tilde{B}$, and hence the Pendant Reduction would be applicable to $\{s'\}$. Hence in this case $\sigma \geq 2$ (Fig. 8).

Having this structure, it is natural to make the following branching.

Branching step 12 If $A_s \backslash A^\circ = \{s, s'\}$, then pursue branching on $\{s, t\}$ with fixing ss'.

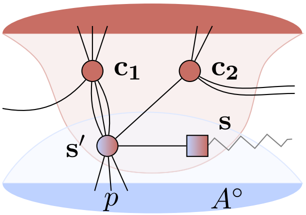

Fig. 8 Case (1,1)(c.i): $\tilde{A} = A^\circ$, $\Delta(\tilde{B}) = 1$, $A_s \cap B_s = \{s, s'\}$. A careful reader might notice that since the excess of B_s is 1, an edge count implies $r = 2$

Let $(A_{ss'\to A}, B_{ss'\to A})$ and $(A_{ss'\to B}, B_{ss'\to B})$ be the respective branches, i.e., minimum-cost maximal terminal separations extending $(A^\circ \cup \{s, s'\}, B^\circ \cup \{t\})$ and $(A^\circ \cup \{t\}, B^\circ \cup \{s, s'\})$, respectively. Of course, if in any of these branches one more terminal pair got resolved, then we have a good branching vector. Assume therefore that this is not the case. In branch $(A_{ss'\to A}, B_{ss'\to A})$ from Lemma 5.2 we have that $\Delta(A^\circ \cup \{s, s'\}) \geq 1$ and $\sigma \geq 2$ Boundary Reductions are triggered within $B_s \backslash \{s\}$. In branch $(A_{ss'\to B}, B_{ss'\to B})$ we again have that $\Delta(B^\circ \cup \{s, s'\}) \geq 1$, and $p \geq 1$ Boundary Reduction are triggered for edges between s' and A°.

We now calculate the obtained branching vector depending on whether t is an antenna or is of type $(1, 1)$.

If t is an antenna with natural side A°, then in branch $(A_{ss'\to A}, B_{ss'\to A})$ we have $\Delta(B_{ss'\to A}) \geq 1$ and one Boundary Reduction is triggered on edges incident to t'. Since $t' \notin B_s$, it is easy to see that the execution of the σ previous Boundary Reductions on the side of s could not spoil the applicability of this Boundary Reduction. In branch $(A_{ss'\to B}, B_{ss'\to B})$, we do not account for any gain on the side of t. Thus we arrive at branching vector $[1, 2, 1 + \sigma; 1, 1, p]$, which is $[1, 2, 3; 1, 1, 1]$ or better, and hence good.

If t is an antenna with natural side B°, then in branch $(A_{ss'\to A}, B_{ss'\to A})$ we do not account for any gain on the side of t. However, in branch $(A_{ss'\to B}, B_{ss'\to B})$ we have that $\Delta(A_{ss'\to B}) \geq 1$ and one Boundary Reduction is triggered on edges incident to t'. Since $t' \neq s'$, again it is easy to see that the execution of the p previous Boundary Reductions on the side of s could not spoil the applicability of this Boundary Reduction. Thus we arrive at branching vector $[1, 2, \sigma; 1, 1, 1 + p]$, which is $[1, 2, 2; 1, 1, 2]$ or better, and hence good.

Finally, suppose t is of type $(1, 1)$. Then by Lemma 5.2 we infer that $\Delta(B_{ss'\to A}) \geq 1$ and $\Delta(A_{ss'\to B}) \geq 1$. Hence we have a branching vector $[1, 2, \sigma; 1, 2, p]$ or better, which is good because $\sigma \geq 2$ and $p \geq 1$.

Subcase (c.ii) $A_s \backslash A^\circ = \{s\}$. We again investigate the branch on $\{s, t\}$ with fixing ss'; for now we do not state that we indeed perform it, because its execution will take place only if the progress will be large enough. Let $(A_{ss'\to A}, B_{ss'\to A})$ and $(A_{ss'\to B}, B_{ss'\to B})$ be the respective branches, i.e., minimum-cost maximal terminal separations extending $(A^\circ \cup \{s, s'\}, B^\circ \cup \{t\})$ and $(A^\circ \cup \{t\}, B^\circ \cup \{s, s'\})$, respectively. Of course, if in any of these branches an additional terminal pair gets resolved, then we already have a good branching vector, so assume henceforth that this is not the case. Since $A_s = A^\circ \cup \{s\}$, by Assumption 2 we infer that $\Delta(A_{ss'\to A}) > \Delta(A_s) = 1$, because in $A_{ss'\to A}$ at least one more vertex (namely s') is assigned to the A-side. As before, in branch

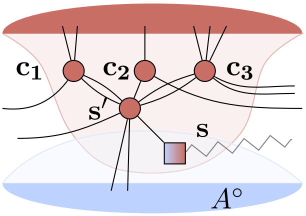

Fig. 9 Case (1,1)(c.ii): $\tilde{A} = A^\circ$, $\Delta(\tilde{B}) = 1$, $A_s \cap B_s = \{s\}$

$(A_{ss'\to A}, B_{ss'\to A})$ we have that $\sigma \geq 1$ Boundary Reductions are triggered inside B_s. In branch $(A_{ss'\to B}, B_{ss'\to B})$, by Lemma 5.2 we have that $\Delta(B_{ss'\to B}) \geq 1$, and we do not account for any Boundary Reductions (Fig. 9).

Let us now investigate what happens in respective branches on the side of terminal t, depending on the type of t. Suppose first that t is of type $(1, 1)$. Then, by Lemma 5.2 it follows that $\Delta(B_{ss'\to A}) \geq 1$ and $\Delta(A_{ss'\to B}) \geq 1$, and hence together with the account of the progress on the side of s, we obtain a branching vector $[1, 3, \sigma; 1, 2, 0]$ or better, which is good because $\sigma \geq 1$.

We are left with the case when t is an antenna, where it can be easily verified that the reasoning as above does not lead to a good branching vector without any deeper analysis. We distinguish two subsubcases, depending on the natural side of t.

Subsubcase (c.ii.A) the natural side of t is A°. Let t' be the unique neighbor of t. Since t is an antenna, there are x edges from t' to A° and x edges from t' to $V(G)$, for some $x \geq 1$.

We now introduce a new type of a branching step that we shall call *skewed branching*. Namely, we will branch into separations $(A_{\mathrm{nt}}, B_{\mathrm{nt}})$ and $(A_{\mathrm{unt}}, B_{\mathrm{unt}})$ that are minimum-cost terminal separations extending $(A^\circ \cup \{t\}, B^\circ \cup \{s\})$ and $(A^\circ \cup \{s, s'\}, B^\circ \cup \{t, t'\})$, respectively. It is easy to see that this branching step is correct, because there is always an optimum integral separation (A^*, B^*) extending (A°, B°) where (1) $s \in B^*$ and $t \in A^*$, or (2) $\{s, s'\} \in A^*$ and $\{t, t'\} \in B^*$. Namely, if neither the first nor the second property is satisfied, then swapping the sides of s and t does not increase the cost of the separation (because the second property is not satisfied), but it makes the first property satisfied.

The reader should think of the skewed branching in the following way. For terminal t, the side A° is the natural side to be assigned to, whereas for s it is B° that is more natural. More precisely, in the branch where we have such assignment, we are not able to reason about any Boundary Reductions being triggered. We do, however, hope for a large decrease in the potential in the opposite branch, where both terminals are assigned to their unnatural sides. Therefore, in this unnatural branch we fix both edges ss' and tt' to maximize the progress measured in the potential function, while in the natural branch we do not fix anything, because this would not lead to any profit in the analysis.

Let us now calculate the branching vector that we obtain when we perform the described skewed branching; of course we assume that no other terminal pair gets resolved in either of the branches, because then we immediately obtain a good branching vector. In branch $(A_{\mathrm{nt}}, B_{\mathrm{nt}})$, by Lemma 5.2 we have that $\Delta(A_{\mathrm{nt}}) \geq 0$

and $\Delta(B_{\mathrm{nt}}) \geq 1$, and we do not account for any applications of the Boundary Reduction. In branch $(A_{\mathrm{unt}}, B_{\mathrm{unt}})$, however, we have $\Delta(A_{\mathrm{unt}}) \geq 2$ by Assumption 2, because $A_s = A^\circ \cup \{s\}$, and $\Delta(B_{\mathrm{unt}}) \geq 2$, by the definition of an antenna and the fact that B° is the unnatural side of t. Moreover, in this branch $x \geq 1$ Boundary Reductions are applicable to the edges between t' and A° and $\sigma \geq 1$ Boundary Reductions are applicable within B_s. Since $t' \notin B_s$, these applications do not interfere with each other. Thus, we arrive at a branching vector $[1, 1, 0; 1, 4, x + \sigma]$, or better. This branching vector is good unless $x = \sigma = 1$. Also, even if $x = \sigma = 1$ but $\Delta(B_{\mathrm{unt}}) \geq 3$, then this leads to branching vector $[1, 1, 0; 1, 5, 2]$ or better, which is good. Thus, we can state the following branching step.

Branching step 13 Unless $x = \sigma = 1$ and $\Delta(B_{\mathrm{unt}}) = 2$, pursue skewed branching into separations $(A_{\mathrm{nt}}, B_{\mathrm{nt}})$ and $(A_{\mathrm{unt}}, B_{\mathrm{unt}})$.

Henceforth we assume that $x = \sigma = 1$ and $\Delta(B_{\mathrm{unt}}) = 2$. Therefore, the degree of t' in G is equal to 3, and it is adjacent to t, one vertex in A°, and one vertex in $V(G)\backslash(A^\circ \cup B^\circ)$ that shall be whence called v.

Consider now the branch $(A_{\mathrm{unt}}, B_{\mathrm{unt}})$, and suppose there is some optimum terminal separation (A^*, B^*) extending (A°, B°) that conforms to this branch, i.e., it also extends $(A_{\mathrm{unt}}, B_{\mathrm{unt}})$. Suppose that $v \in A^*$. Then this is clearly a contradiction with the optimality of (A^*, B^*), because t' has 2 neighbors in A^* and 1 in B^*, so moving it from B^* to A^* would decrease the cost of the separation. Hence we can assign v greedily to the B-side. More precisely, instead of $(A_{\mathrm{unt}}, B_{\mathrm{unt}})$ we will from now on consider terminal separation $(A^{\mathrm{ext}}_{\mathrm{unt}}, B^{\mathrm{ext}}_{\mathrm{unt}})$ defined as the minimum-cost terminal separation extending $(A^\circ \cup \{s, s'\}, B^\circ \cup \{t, t', v\})$. In case $s' = v$, the reasoning above shows that the branch where $\{s, s'\}$ is assigned to the A-side and $\{t, t'\}$ is assigned to the B-side cannot lead to an optimum solution, so we can greedily pursue the branch where s and t are assigned to respective natural sides.

Reduction step 14 If $v = s'$, then recurse into terminal separation (A_t, B_s).

Hence, from now on we assume that $v \neq s'$ and we branch into $(A_{\mathrm{nt}}, B_{\mathrm{nt}})$ and $(A^{\mathrm{ext}}_{\mathrm{unt}}, B^{\mathrm{ext}}_{\mathrm{unt}})$, where the latter is defined as above. As usual, we assume that $(A^{\mathrm{ext}}_{\mathrm{unt}}, B^{\mathrm{ext}}_{\mathrm{unt}})$ does not resolve any new terminal pair, because then we would have a good branching vector. The same reasoning as for $(A_{\mathrm{unt}}, B_{\mathrm{unt}})$ shows that $\Delta(A^{\mathrm{ext}}_{\mathrm{unt}}) \geq 2$ and $\Delta(B^{\mathrm{ext}}_{\mathrm{unt}}) \geq 2$. As before, if we had that $\Delta(B^{\mathrm{ext}}_{\mathrm{unt}}) \geq 3$, then branching into $(A_{\mathrm{nt}}, B_{\mathrm{nt}})$ and $(A^{\mathrm{ext}}_{\mathrm{unt}}, B^{\mathrm{ext}}_{\mathrm{unt}})$ would lead to a branching vector $[1, 1, 0; 1, 5, 2]$ or better. Hence, we can again assume that $\Delta(B^{\mathrm{ext}}_{\mathrm{unt}}) = 2$.

Therefore, a straightforward edge count shows that $B_q = B^{\mathrm{ext}}_{\mathrm{unt}}\backslash\{t, t'\}$ is a terminal-free B°-extension of excess 2. Hence, we can apply Lemma 4.7 to decompose it. By the inapplicability of the Excess-2 Reduction, we have that $B_q\backslash B^\circ$ has a decomposition of the form $\{c_1, c_2\}$ or $\{d, c_1, \ldots, c_r\}$ (from now on we drop the earlier notation for the decomposition of $B_s\backslash(B^\circ \cup \{s\})$, and use the notation $d, c_1, \ldots, c_r$ for the decomposition of $B_q\backslash B^\circ$). Suppose first that $v = c_i$ for some i. Then this is a contradiction with the optimality of $(A^{\mathrm{ext}}_{\mathrm{unt}}, B^{\mathrm{ext}}_{\mathrm{unt}})$, because then $\{t, t', c_i\}$ would be a set of excess 1, so replacing $B^{\mathrm{ext}}_{\mathrm{unt}}$ with it would decrease the cost of separation $(A^{\mathrm{ext}}_{\mathrm{unt}}, B^{\mathrm{ext}}_{\mathrm{unt}})$. Therefore, B_q has a decomposition of the form $\{d, c_1, \ldots, c_r\}$ where $v = d$.

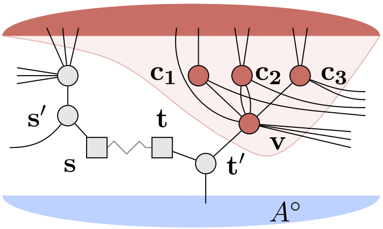

Fig. 10 Case (1,1)(c.ii.A), where furthermore $x = \sigma = 1$ and $\Delta(B_{\mathrm{unt}}) = 2$. The set $B_q = B^{\mathrm{ext}}_{\mathrm{unt}} \backslash \{t, t'\}$ of excess 2 is highlighted in *red* (Color figure online)

By Lemma 4.7, we have that $|E(v, c_i)| = p_i$, $|E(c_i, B^\circ)| = p_i + x_i$ and $|E(c_i, V(G)\backslash B_q)| = x_i + 1$, for some integers $p_i \geq 1$ and $x_i \geq 0$. Let $\sigma_2 = |E(v, B^\circ)| + \sum_{i=1}^{r} p_i$ be the number of Boundary Reductions triggered within B_q when the vertex v is assigned to the A-side. By Lemma 4.10, $\sigma_2 \geq 1$.

Before we proceed, we need to resolve a corner case when $s' \in B_q$. We claim that then it is safe to greedily assign t to the A-side and s to the B-side.

Reduction step 15 If $s' \in B_q$, then recurse with terminal separation (A_t, B_s).

To argue the correctness of this reduction step, we need to prove that there exists an optimum terminal separation extending (A°, B°) where s is assigned to the B-side and t is assigned to the A-side. Let us take any optimum terminal separation (A^*, B^*) that satisfies point 1 of Lemma 4.5. Assume $s \in A^*$ and $t \in B^*$, as otherwise we are done. We can further assume that $s' \in A^*$ and $t' \in B^*$, because otherwise switching the sides of s and t would not increase the cost of the separation, however it would make it satisfy the condition we seek. Suppose first that $s' = v$; then we have an immediate contradiction, because moving t' from A^* to B^* would decrease the cost. Suppose then that $s' = c_i$ for some $i = \{1, 2, \ldots, r\}$. Since (A^*, B^*) satisfies point 1 of Lemma 4.5, we infer that $B^* \cap B_q = B^\circ$ or $B^* \cap B_q = B^\circ \cup \{c_j\}$ for some $j \neq i$. In particular, $v \in A^*$. This is, however, a contradiction, because moving t' from B^* to A^* would decrease the cost of the separation. This justifies the correctness of Reduction Step 15.

Whence we assume that $s' \notin B_q$ (Fig. 10).

We will now branch on vertex v. More precisely, we recurse into branches $(A_{v\to A}, B_{v\to A})$ and $(A_{v\to B}, B_{v\to B})$, defined as minimum-cost terminal separations extending $(A^\circ \cup \{v\}, B^\circ)$ and $(A^\circ, B^\circ \cup \{v\})$, respectively.

Branching step 16 Pursue branching on v, that is, recurse into branches $(A_{v\to A}, B_{v\to A})$ and $(A_{v\to B}, B_{v\to B})$.

The remainder of the description of this subcase is devoted to proving that the execution of this branching step leads to a good branching vector.

Consider first branch $(A_{v\to A}, B_{v\to A})$. By the optimality of $(A_{v\to A}, B_{v\to A})$ we infer that $t' \in A_{v\to A}$, because otherwise assigning it to $A_{v\to A}$, or moving from $B_{v\to A}$ to $A_{v\to A}$, would decrease the cost of the separation. Also, we can assume that $t \in A_{v\to A}$ and $s \in B_{v\to A}$ for the following reason. If this terminal pair was not resolved in $(A_{v\to A}, B_{v\to A})$, then assigning t to the A-side and s to the B-side would not increase the cost of the separation while extending $(A_{v\to A}, B_{v\to A})$, a contradiction with the maximality of $(A_{v\to A}, B_{v\to A})$. However, if $t \in B_{v\to A}$ and $s \in A_{v\to A}$, then we can modify separation $(A_{v\to A}, B_{v\to A})$ by switching the sides of s and t, and because $t' \in A_{v\to A}$, then this modification does not increase the cost.

Hence, in branch $(A_{v\to A}, B_{v\to A})$ the terminal pair $\{s, t\}$ gets resolved. Since $\{v, t, t'\} \subseteq A_{v\to A}$ and $\{s\} \subseteq B_{v\to A}$, by Lemma 5.2 and the definition of an antenna we obtain that $\Delta(A_{v\to A}) \geq 1$ and $\Delta(B_{v\to A}) \geq 1$. Notice also that at least $\sigma_2 \geq 1$ Boundary Reductions are triggered within B_q.

Consider the second branch $(A_{v\to B}, B_{v\to B})$. In it, one Boundary Reduction is triggered on the edges incident to t', and the terminal pair $\{s, t\}$ is either resolved by this branch or is immediately removed by the Lonely Terminal Reduction (possibly preceded by the Pendant Reduction that removes t'). Hence, $\{s, t\}$ also gets resolved in this branch.

From now on, we assume that neither of the considered branches resolves any terminal pair other than $\{s, t\}$, because then, as argued at the beginning of this section, we would immediately achieve a good branching vector. With this assumption in mind, we now claim that $\Delta(B_{v\to B}) \geq 2$.

Claim 5.10 $\Delta(B_{v\to B}) \geq 2$.

Proof If $B_{v\to B}$ is a terminal-free extension of B°, then this follows from Lemma 4.9. We have two cases left to investigate: either $t \in B_{v\to B}$ or $s \in B_{v\to B}$.

In the first case, since $v \in B_{v\to B}$ by the optimality of $(A_{v\to B}, B_{v\to B})$ it follows that also $t' \in B_{v\to B}$. But then if $B_{v\to B}$ was an extension of excess at most 1, then $B_{v\to B}\setminus\{t, t'\}$ would be a terminal-free extension of excess at most 1, a contradiction with Lemma 4.9.

In the second case, assume for the sake of contradiction that $\Delta(B_{v\to B}) = 1$ (it cannot happen that $\Delta(B_{v\to B}) = 0$, because then $(A^\circ \cup \{t, t'\}, B_{v\to B})$ would be an extension of (A°, B°) of the same cost, a contradiction with the maximality of (A°, B°)). Then $B_{v\to B}\setminus\{s\}$ is a terminal-free set of excess 2. Since $v \in B_{v\to B}$, by the optimality of $(A_{v\to B}, B_{v\to B})$ we obtain that $c_i \in B_{v\to B}$ for each $i \in \{1, 2, \ldots, r\}$, so $B_q \subseteq B_{v\to B}\setminus\{s\}$.

On the other hand, $s' \in B_{v\to B}$ because $G[B_{v\to B}\setminus B^\circ]$ is connected by the same reasoning as in Lemma 5.1. But we are currently working with the assumption that $s' \notin B_q$, so B_q is a strict subset of $B_{v\to B}\setminus\{s\}$.

Thus, we obtain a contradiction with Lemma 4.8. Indeed, from this lemma it follows that B_q consists of two vertices that are adjacent to B° and each of them forms an excess-1 extension of B°, but we know that $v \in B_q$ is the vertex d of the decomposition of B_q and by Lemma 4.7, $\Delta(B^\circ \cup \{v\}) > 1$. □

We conclude that the considered branching leads to branching vector $[1, 2, \sigma_2; 1, 2, 1]$, or better. This vector is good unless $\sigma_2 = 1$. Also, if in fact $\Delta(A_{v\to A}) \geq 3$, then we also arrive at a good branching vector $[1, 3, 1; 1, 2, 1]$, or better. Hence, from now on assume that $\sigma_2 = 1$ and $\Delta(A_{v\to A}) = 2$.

If $\Delta(A_{v\to A}) = 2$, then $A_{v\to A}\setminus\{t, t'\}$ is a terminal-free extension of A° of excess 2. Let us apply Lemma 4.7 to it. Regardless of the form of the decomposition, from Lemma 4.10 we infer that in the branch $(A_{v\to B}, B_{v\to B})$ at least one Boundary Reduction will be triggered within $A_{v\to A}\setminus\{t, t'\}$. This Boundary Reduction is applied independently of the Boundary Reduction triggered on edges incident to t' that we previously counted in branch $(A_{v\to B}, B_{v\to B})$. This gives one additional Boundary Reduction that we did not account for previously, which leads to a good branching vector $[1, 2, 1; 1, 2, 2]$, or better.

Subsubcase (c.ii.B) the natural side of t **is** B°. Recall that we investigated the branch $(A_{ss'\to A}, B_{ss'\to A})$ and $(A_{ss'\to B}, B_{ss'\to B})$, and we concluded that $\Delta(A_{ss'\to A}) \geq 2$, $\Delta(B_{ss'\to B}) \geq 1$, and $\sigma \geq 1$ Boundary Reductions are triggered inside B_s in branch $(A_{ss'\to A}, B_{ss'\to A})$. From the definition of an antenna we have that $\Delta(A_{ss'\to B}) \geq 1$ and one Boundary Reduction is triggered on edges incident to t' in branch $(A_{ss'\to B}, B_{ss'\to B})$, when t is assigned to the A-side. This gives us branching vector $[1, 2, \sigma; 1, 2, 1]$ or better, which is good unless $\sigma = 1$. Also, note that this branching is good if s' is adjacent to a second terminal different than s: due to inapplicability of the Common Neighbor Reduction, this terminal would belong to a second terminal pair that would get resolved in the branching. This justifies the execution of the following branching step.

Branching step 17 If $\sigma > 1$ or s' is adjacent to a second terminal different than s, then recurse into branches $(A_{ss'\to A}, B_{ss'\to A})$ and $(A_{ss'\to B}, B_{ss'\to B})$.

Recall that from Lemma 5.6 we obtained a decomposition $\{d, c_1, c_2, \ldots, c_r\}$ of $(B_s\backslash\{s\}) \setminus B^\circ$, where $d = s'$ is the unique neighbor of s. By Lemma 4.10, if $\sigma = 1$ then we have two cases:

- either $r = 0$ and s' has degree 4: one edge to B°, one edge to s, and two edges to $V(G)\backslash(B^\circ \cup \{s\})$;
- or $r = 1$ and s' has degree 3: one edge to c_1, one edge to s, and one edge to $V(G)\backslash(B^\circ \cup \{s, c_1\})$. Moreover, c_1 is incident exactly on the following edges: 1 edge to s', x edges to B° and x edges to $V(G)\backslash B^\circ \cup (\{s, s'\})$, for some $x \geq 1$.

Let $y = |E(t', B^\circ)|$; then $y \geq 1$. We now investigate the cases separately.

Subsubsubcase (c.ii.B.1) $r = 0$. We first investigate the possibility of branching on $\{s, t\}$ with fixing tt'. That is, we examine branches $(A_{tt'\to B}, B_{tt'\to B})$ and $(A_{tt'\to A}, B_{tt'\to A})$ that are minimum-cost maximal extensions of $(A^\circ\cup\{s\}, B^\circ\cup\{t, t'\})$ and $(A^\circ \cup \{t, t'\}, B^\circ \cup \{s\})$, respectively. Of course if any of these branches resolves some terminal pair other than $\{s, t\}$, then we obtain a good branching vector. Hence, assume this is not the case.

Consider the first branch $(A_{tt'\to B}, B_{tt'\to B})$. By Lemma 5.2 we have that $\Delta(A_{tt'\to B}) \geq 1$. Also, one Boundary Reduction is triggered on edges incident to s'.

Consider the second branch $(A_{tt'\to A}, B_{tt'\to A})$. By the definition of an antenna, we have that $\Delta(A_{tt'\to A}) \geq 2$ and by Lemma 5.2 we have that $\Delta(B_{tt'\to A}) \geq 1$. Also, y Boundary Reductions are triggered on edges between t' and B° in this branch.

This leads to a branching vector $[1, 1, 1; 1, 3, y]$ or better, which is good unless $y = 1$. This justifies executing the following step (Fig. 11).

Branching step 18 If $y > 1$, then recurse into branches $(A_{tt'\to B}, B_{tt'\to B})$ and $(A_{tt'\to A}, B_{tt'\to A})$ (Fig. 12).

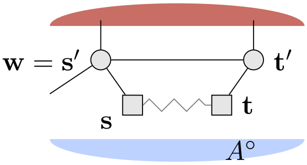

Fig. 11 Case (1,1)(c.ii.B.1), where furthermore s' is a neighbor of t'

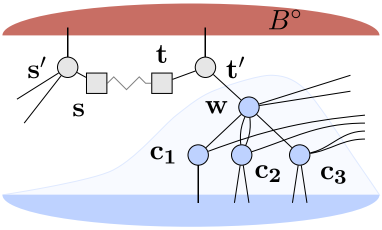

Fig. 12 Case (1,1)(c.ii.B.1), where furthermore $w \neq s'$ and $\Delta(A^{\mathrm{ext}}_{tt' \to A}) = 2$. The set $A_q = A^{\mathrm{ext}}_{tt' \to A} \setminus \{t, t'\}$ of excess 2 is highlighted in *blue* (Color figure online)

From now on we assume that $y = 1$, and, consequently, we have that t' has degree 3: it neighbors t, one vertex in B°, and one vertex w outside $B^\circ \cup \{t\}$.

We now resolve the corner case when $w = s'$. Then t' is adjacent to t, s' and one vertex in B°, whereas s' is adjacent to s, t', one vertex in B°, and one vertex outside $B^\circ \cup \{s, s', t, t'\}$. We claim that we can assign greedily s to the A-side and t to the B-side. To argue this, take any minimum-cost integral terminal separation (A^*, B^*) extending (A°, B°), and assume that $s \in B^*$ and $t \in A^*$. We can further assume that $s' \in B^*$ and $t' \in A^*$, because otherwise switching the sides of s and t produces an integral terminal separation of no larger cost where s and t are on the sides we aimed for. But then t' has two neighbors in B^* and one in A^*, which is a contradiction with the optimality of (A^*, B^*) — moving t' from A^* to B^* would decrease the cost. This justifies the correctness of the following step.

Reduction step 19 If $w = s'$, then recurse into branch (A_s, B_t).

From now on, we assume that $w \neq s'$.

Suppose now that $|E(w, A^\circ)| > 0$. Then in branch $(A_{tt' \to B}, B_{tt' \to B})$ one additional Boundary Reduction is triggered on edges incident to w. Since $w \neq s'$, the application of this Boundary Reduction cannot spoil the applicability of the Boundary Reduction on edges incident to s' that we counted in the same branch. This results in branching vector $[1, 1, 2; 1, 3, 1]$ or better, which is good, so we can do the following.

Branching step 20 If $|E(w, A^\circ)| > 0$, then recurse into branches $(A_{tt' \to B}, B_{tt' \to B})$ and $(A_{tt' \to A}, B_{tt' \to A})$.

Henceforth we assume that $E(w, A^\circ) = \emptyset$.

As in Case (c.ii.A), we argue that in branch $(A_{tt' \to A}, B_{tt' \to A})$ we can greedily assign w to the A-side. More precisely, instead of $(A_{tt' \to A}, B_{tt' \to A})$ we will from now on consider terminal separation $(A^{\mathrm{ext}}_{tt' \to A}, B^{\mathrm{ext}}_{tt' \to A})$ defined as the minimum-cost terminal separation extending $(A^\circ \cup \{t, t', w\}, B^\circ \cup \{s\})$. The argumentation for the correctness of this step is as before: Suppose there is some optimum terminal separation (A^*, B^*) extending (A°, B°) that conforms to this branch, i.e., it also extends $(A_{tt' \to A}, B_{tt' \to A})$.

Suppose that $w \in B^*$. Then this is clearly a contradiction with the optimality of (A^*, B^*), because t' has 2 neighbors in B^* and 1 in A^*, so moving it from A^* to B^* would decrease the cost of the separation.

As usual, $(A^{\text{ext}}_{tt'\to A}, B^{\text{ext}}_{tt'\to A})$ resolving an additional terminal pair would immediately lead to a good branching vector when branching into $(A_{tt'\to B}, B_{tt'\to B})$ and $(A^{\text{ext}}_{tt'\to A}, B^{\text{ext}}_{tt'\to A})$, so assume this is not the case.

In branch $(A^{\text{ext}}_{tt'\to A}, B^{\text{ext}}_{tt'\to A})$, by the definition of an antenna, we have that $\Delta(A^{\text{ext}}_{tt'\to A}) \geq 2$ and by Lemma 5.2 we have that $\Delta(B^{\text{ext}}_{tt'\to A}) \geq 1$. Also, a Boundary Reduction is triggered on the edge between t' and B° in this branch. Observe that if we in fact had that $\Delta(A^{\text{ext}}_{tt'\to A}) \geq 3$, then branching into $(A_{tt'\to B}, B_{tt'\to B})$ and $(A^{\text{ext}}_{tt'\to A}, B^{\text{ext}}_{tt'\to A})$ results in branching vector $[1, 1, 1; 1, 4, 1]$ or better, which is good. This justifies the execution of the following.

Branching step 21 If $\Delta(A^{\text{ext}}_{tt'\to A}) \geq 3$, then recurse into branches $(A_{tt'\to B}, B_{tt'\to B})$ and $(A^{\text{ext}}_{tt'\to A}, B^{\text{ext}}_{tt'\to A})$.

From now on we assume that $\Delta(A^{\text{ext}}_{tt'\to A}) = 2$. This means that $A_q = A^{\text{ext}}_{tt'\to A} \setminus \{t, t'\}$ is a terminal-free extension of A° of excess 2, and moreover $w \in A_q$. Hence, we can apply Lemma 4.7 to A_q and obtain a decomposition of the form $\{c_1, c_2, \}$ or $\{d, c_1, \ldots, c_r\}$ (we now drop the notation for the decomposition of B_s, and use it for the decomposition of A_q instead). By Lemma 4.7, each c_i is connected with A° via at least one edge, but we assumed that there is no edge between w and A°. Hence the decomposition has the form $\{d, c_1, \ldots, c_r\}$ and $d = w$. By Lemma 4.10, at least one Boundary Reduction is triggered within A_q in any branch where w is assigned to B.

We will pursue branching on vertex w. More precisely, we consider recursing into branches $(A_{w\to A}, B_{w\to A})$ and $(A_{w\to B}, B_{w\to B})$, defined as minimum-cost terminal separations extending $(A^\circ \cup \{w\}, B^\circ)$ and $(A^\circ, B^\circ \cup \{w\})$, respectively.

Branching step 22 Pursue branching on w, that is, recurse into branches $(A_{w\to A}, B_{w\to A})$ and $(A_{w\to B}, B_{w\to B})$.

The remainder of this case is devoted to arguing that this branching step leads to a good branching vector.

Consider branch $(A_{w\to A}, B_{w\to A})$. Then a Boundary Reduction is triggered on edges incident to t', and consequently the pair $\{s, t\}$ either gets resolved in this branch, or is removed by an application of the Lonely Terminal Reduction (possibly preceded by the Pendant Reduction that removes t'). Hence, at least the terminal pair $\{s, t\}$ gets resolved or removed in this branch.

In branch $(A_{w\to B}, B_{w\to B})$ we have that $t' \in B_{w\to B}$ due to the optimality of $(A_{w\to B}, B_{w\to B})$. Moreover, we can assume that $t \in B_{w\to B}$ and $s \in A_{w\to B}$ for the following reason. If this terminal pair was not resolved in $(A_{w\to B}, B_{w\to B})$, then assigning t to the B-side and s to the A-side would not increase the cost of the separation while extending $(A_{w\to B}, B_{w\to B})$, a contradiction with the maximality of $(A_{w\to B}, B_{w\to B})$. However, if $t \in A_{w\to B}$ and $s \in B_{w\to B}$, then we can modify separation $(A_{w\to B}, B_{w\to B})$ by switching the sides of s and t, and because $t' \in B_{w\to B}$, then this modification does not increase the cost.

Hence, in both branches, the terminal pair $\{s, t\}$ gets eventually removed. Suppose then that neither $(A_{w\to A}, B_{w\to A})$ nor $(A_{w\to B}, B_{w\to B})$ resolves another terminal pair, because otherwise we would immediately have a good branching vector. We continue the calculation of the obtained branching vector with this assumption. First, we need an analogue of Claim 5.10, whose proof is very similar.

Claim 5.11 $\Delta(A_{w\to A}) \geq 2$.

Proof If $A_{w\to A}$ is a terminal-free extension of A°, then this follows from Lemma 4.9. We have two cases left to investigate: either $t \in A_{w\to A}$ or $s \in A_{w\to A}$.

In the first case, since $w \in A_{w\to A}$ by the optimality of $(A_{w\to A}, B_{w\to A})$ it follows that also $t' \in A_{w\to A}$. But then if $A_{w\to A}$ was an extension of excess at most 1, then $A_{w\to A}\backslash\{t, t'\}$ would be a terminal-free extension of excess at most 1, a contradiction with Lemma 4.9.

In the second case, assume for the sake of contradiction that $\Delta(A_{w\to A}) = 1$ (it cannot happen that $\Delta(A_{w\to A}) = 0$, because then $(A_{w\to A}, B^\circ \cup \{t, t'\},)$ would be an extension of (A°, B°) of the same cost, a contradiction with the maximality of (A°, B°)). Then $A_{w\to A}\backslash\{s\}$ is a terminal-free set of excess 2 that contains w. Since $w \in A_{w\to A}$, by the optimality of $(A_{w\to A}, B_{w\to A})$ we obtain that $c_i \in A_{w\to A}$ for each $i \in \{1, 2, \dots, r\}$, so $A_q \subseteq A_{w\to A}\backslash\{s\}$.

On the other hand, $s' \in A_{w\to A}$ because $G[A_{w\to A}\backslash A^\circ]$ is connected by the same reasoning as in the proof of Lemma 5.1. However, observe that $s' \notin A_q$. Indeed, we are working with the assumption that $s' \neq w$, and moreover $s' \neq c_i$ for each $i = 1, 2, \dots, r$ because otherwise the Boundary Reduction would apply to s'. Hence A_q is a strict subset of $A_{w\to A}\backslash\{s\}$.

Thus, we obtain a contradiction with Lemma 4.8. Indeed, from this lemma it follows that A_q consists of two vertices that are adjacent to A°, but we know that $w \in A_q$ and $E(w, A^\circ) = \emptyset$. □

Observe that in branch $(A_{w\to A}, B_{w\to A})$ we also have one Boundary Reduction triggered on the edges incident to t'.

On the other hand, in branch $(A_{w\to B}, B_{w\to B})$ we have $\Delta(A_{w\to B}) \geq 1$ by Lemma 5.2 because $s \in A_{w\to B}$. Also, $\Delta(B_{w\to B}) \geq 1$ by the definition of an antenna and the fact that $\{w, t, t'\} \subseteq B_{w\to B}$. Finally, one Boundary Reduction is triggered on edges incident to s' and one Boundary Reduction is triggered within A_q. Since $s' \neq w$, the application of one of these reductions cannot spoil the applicability of the other.

Thus we obtain branching vector $[1, 2, 1; 1, 2, 2]$ or better, which is a good branching vector.

Subsubsubcase (c.ii.B.2) $r = 1$. Recall that s' has degree 3 and has one edge to c_1, one edge to s, and one edge to $V(G)\backslash(B^\circ \cup \{s, c_1\})$, whereas c_1 has x edges to B° and x edges to $V(G)\backslash(B^\circ \cup \{s, c_1\})$, for some $x \geq 1$. Let v be the neighbor of s' other than c_1 and s.

First, note that v is not a terminal, as we have already excluded the case when s' is adjacent to a second terminal different than s.

We claim that there is an optimum integral terminal separation (A^*, B^*) extending (A°, B°) where vertices s' and v are assigned to the same side. Take any optimum integral separation (A^*, B^*) and suppose that s' and v are assigned to different sides.

Assume first that $s' \in A^*$ and $v \in B^*$. Then $s \in A^*$ and $c_1 \in A^*$ because otherwise moving s' from A^* to B^* would decrease the cost of the separation. Hence $t \in B^*$, and again by the optimality of (A^*, B^*) we have $t' \in B^*$. Consider a modified integral terminal separation (A^*_m, B^*_m) obtained from (A^*, B^*) by moving $\{c_1, s'\}$ from the A-side to the B-side. Recall that c_1 has x edges to B° and x edges going outside of $B^\circ \cup \{c_1, s'\}$. Then it is easy to see that the cost of (A^*_m, B^*_m) is not larger than the cost of (A^*, B^*), whereas s' and v are both assigned to the B-side.

Assume second that $s' \in B^*$ and $v \in A^*$. Similarly as before, by the optimality of (A^*, B^*) we have that $s \in B^*$ and $c_1 \in B^*$. Consequently, $t \in A^*$. If we had that $t' \in B^*$, then moving t from A^* to B^* and $\{s, s'\}$ from B^* to A^* would strictly decrease the cost of the separation, a contradiction with the optimality of (A^*, B^*). Hence we have that $t' \in A^*$. Consider a modified integral terminal separation (A^*_m, B^*_m) obtained from (A^*, B^*) by moving $\{s, s'\}$ from the B-side to the A-side, and moving $\{t, t'\}$ from the A-side to the B-side. Recall that t' has y edges going to B° and y edges going outside of $B^\circ \cup \{t, t'\}$. Hence it is easy to see that the cost of (A^*_m, B^*_m) is not larger than the cost of (A^*, B^*), whereas s' and v are both assigned to the A-side.

This justifies the execution of the following reduction in this case.

Reduction step 23 Merge s' and v.

As the case study is exhaustive, this finishes the description of the branching algorithm. We hope that the reader shares the joy of the writer after getting to this line.

6 Conclusions

In this work we have developed an algorithm for Edge Bipartization that has running time $\mathcal{O}(1.977^k \cdot nm)$, which is the first one to achieve a dependence on the parameter better than 2^k. Our result shows that in the case of Edge Bipartization the constant 2 in the base of the exponent is not the ultimate answer, as is conjectured for CNF- SAT. Also, it improves some recent works where the FPT algorithm for Edge Bipartization is used as a black-box [20]. However, our work leaves a number of open questions that we would like to highlight.

- Reducing the dependence on the parameter from 2^k to 1.977^k can be only considered a "proof of concept" that such an improvement is possible. Even though we believe that it is an important step in understanding the optimal parameterized complexity of graph separation problems, we put forward the question of designing a reasonably simple algorithm with the running time dependence on the parameter substantially better than 2^k. Last but not least, it is also interesting whether the dependence of the running time on the input size can be improved from $\mathcal{O}(nm)$ to linear.
- There is a simple reduction that reduces back Terminal Separation to Edge Bipartization without changing the size of the cutset.[4] Given a Terminal Separation instance: $(G, \mathcal{T}, (A^\circ, B^\circ), k)$:
  1. subdivide once every edge of G;
  2. add a new terminal pair (a_0, b_0), connect a_0 to every vertex of A° with $k+1$ parallel edges, and connect b_0 to every vertex of B° with $k+1$ parallel edges;

[4] We thank an anonymous reviewer for suggesting this reduction.

3. for every terminal pair (s, t), including (a_0, b_0), connect s and t with $k + 1$ parallel edges.

Recall that our approach can be summarized as follows: having observed that Terminal Separation admits a simple $\mathcal{O}^\star(2^{|\mathcal{T}|})$-time algorithm and an $\mathcal{O}^\star(4^k)$-time algorithm using the CSP-guided technique of Wahlström [30], we develop an algorithm for a joint parameterization $(|\mathcal{T}|, k)$ that for $|\mathcal{T}| = k + 1$ achieves running time $\mathcal{O}^\star(1.977^k)$. The aforementioned reduction yields an $\mathcal{O}^\star(1.977^k)$-time algorithm for Terminal Separation. The remaining question is: can Terminal Separation be solved in time $\mathcal{O}^\star(c^{|\mathcal{T}|})$ for some $c < 2$? Maybe one can prove matching lower bounds under the Strong Exponential Time Hypothesis (SETH), or under the Set Cover Conjecture (SeCoCo) [3]?

- Finally, we would like to reiterate two related open questions.
  - The currently fastest algorithm for OCT runs in time $\mathcal{O}^\star(2.3146^k)$ [25]. It is reasonable to expect that this base of the exponent is an artifact of the technique. Is it possible to design an algorithm with running time simply $\mathcal{O}^\star(2^k)$?
  - In their work, Cygan et al. [9] presented an algorithm for Node Multiway Cut with running time $\mathcal{O}^\star(2^k)$, based on the idea of LP-guided branching. Is it possible to obtain an algorithm with running time $\mathcal{O}^\star(c^k)$ for some $c < 2$? As shown by Cao et al. [1], this is indeed the case for the edge deletion variant.

Acknowledgements Marcin Pilipczuk: Partially supported by the Centre for Discrete Mathematics and its Applications (DIMAP) at the University of Warwick and by Warwick-QMUL Alliance in Advances in Discrete Mathematics and its Applications. Michał Pilipczuk: Supported by the Polish National Science Centre Grant DEC-2013/11/D/ST6/03073 and Foundation for Polish Science via the START stipend program. During the work on these results, Michał Pilipczuk has been holding a post-doc position of Warsaw Centre of Mathematics and Computer Science. Marcin Wrochna: Supported by the Polish National Science Centre Grant DEC-2013/11/D/ST6/03073.

References

1. Cao, Y., Chen, J., Fan, J.: An $\mathcal{O}^\star(1.84^k)$ parameterized algorithm for the multiterminal cut problem. Inf. Process. Lett. **114**(4), 167–173 (2014)
2. Cunnigham, W.H., Edmonds, J.: A combinatorial decomposition theory. Can. J. Math. **32**, 734–765 (1980)
3. Cygan, M., Dell, H., Lokshtanov, D., Marx, D., Nederlof, J., Okamoto, Y., Paturi, R., Saurabh, S., Wahlström, M.: On problems as hard as CNF-SAT. ACM Trans. Algorithms **12**(3), 41 (2016)
4. Cygan, M., Fomin, F.V., Jansen, B.M.P., Kowalik, L., Lokshtanov, D., Marx, D., Pilipczuk, M., Pilipczuk, M., Saurabh, S.: Open problems from the school on parameterized algorithms and complexity, Bedlewo (2014). http://fptschool.mimuw.edu.pl/opl.pdf
5. Cygan, M., Fomin, F.V., Kowalik, L., Loksthanov, D., Marx, D., Pilipczuk, M., Pilipczuk, M., Saurabh, S.: Parameterized Algorithms. Springer, Berlin (2015)
6. Cygan, M., Kowalik, L., Pilipczuk, M.: Open problems from the update meeting on graph separation problems. Workshop on Kernels, Warsaw (2013). http://worker2013.mimuw.edu.pl/slides/update-opl.pdf

7. Cygan, M., Kratsch, S., Nederlof, J.: Fast hamiltonicity checking via bases of perfect matchings. In: Proceedings of STOC'13, pp. 301–310. ACM (2013)
8. Cygan, M., Nederlof, J., Pilipczuk, M., Pilipczuk, M., van Rooij, J.M.M., Wojtaszczyk, J.O.: Solving connectivity problems parameterized by treewidth in single exponential time. In: Proceedings of FOCS'11, pp. 150–159. IEEE (2011)
9. Cygan, M., Pilipczuk, M., Pilipczuk, M., Wojtaszczyk, J.O.: On multiway cut parameterized above lower bounds. TOCT **5**(1), 3 (2013)
10. Diestel, R.: Graph Theory. Springer, Berlin (2005)
11. Downey, R.G., Fellows, M.R.: Fundamentals of Parameterized Complexity. Texts in Computer Science. Springer, Berlin (2013)
12. Fomin, F.V., Kratsch, D.: Exact Exponential Algorithms. Texts in Theoretical Computer Science. An EATCS Series. Springer, Berlin (2010)
13. Guillemot, S.: FPT algorithms for path-transversal and cycle-transversal problems. Discrete Optim. **8**(1), 61–71 (2011)
14. Guo, J., Gramm, J., Hüffner, F., Niedermeier, R., Wernicke, S.: Compression-based fixed-parameter algorithms for feedback vertex set and edge bipartization. J. Comput. Syst. Sci. **72**(8), 1386–1396 (2006)
15. Hopcroft, J.E., Tarjan, R.E.: Dividing a graph into triconnected components. SIAM J. Comput. **2**(3), 135–158 (1973)
16. Hüffner, F.: Algorithm engineering for optimal graph bipartization. J. Graph Algorithms Appl. **13**(2), 77–98 (2009)
17. Iwata, Y., Wahlström, M., Yoshida, Y.: Half-integrality, lp-branching, and FPT algorithms. SIAM J. Comput. **45**(4), 1377–1411 (2016)
18. Kawarabayashi, K., Reed, B.A.: An (almost) linear time algorithm for odd cyles transversal. In: Proceedings of SODA'10, pp. 365–378. SIAM (2010)
19. Kolay, S., Misra, P., Ramanujan, M.S., Saurabh, S.: Parameterized approximations via d-skew-symmetric multicut. In: Proceedings of MFCS'14. Lecture notes in computer science, vol. 8635, pp. 457–468. Springer (2014)
20. Kolay, S., Panolan, F., Raman, V., Saurabh, S.: Parameterized algorithms on perfect graphs for deletion to (r, l)-graphs. In: Proceedings of MFCS'14, LIPIcs, vol. 58, pp. 75:1–75:13. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik (2016)
21. Kolmogorov, V., Thapper, J., Živný, S.: The power of linear programming for general-valued CSPs. SIAM J. Comput. **44**(1), 1–36 (2015)
22. Kratsch, S., Wahlström, M.: Compression via matroids: a randomized polynomial kernel for odd cycle transversal. ACM Trans. Algorithms **10**(4), 20:1–20:15 (2014)
23. Lokshtanov, D., Marx, D., Saurabh, S.: Known algorithms on graphs on bounded treewidth are probably optimal. In: Proceedings of SODA'11, pp. 777–789. SIAM (2011)
24. Lokshtanov, D., Marx, D., Saurabh, S.: Lower bounds based on the exponential time hypothesis. Bull. EATCS **105**, 41–72 (2011)
25. Lokshtanov, D., Narayanaswamy, N.S., Raman, V., Ramanujan, M.S., Saurabh, S.: Faster parameterized algorithms using linear programming. ACM Trans. Algorithms **11**(2), 15:1–15:31 (2014)
26. Lokshtanov, D., Saurabh, S., Sikdar, S.: Simpler parameterized algorithm for OCT. In: Proceedings of IWOCA'09, pp. 380–384. Springer (2009)
27. Marx, D.: What's next? Future directions in parameterized complexity. The multivariate algorithmic revolution and beyond. Lecture Notes in Computer Science, vol. 7370, pp. 469–496. Springer, Berlin (2012)
28. Reed, B.A., Smith, K., Vetta, A.: Finding odd cycle transversals. Oper. Res. Lett. **32**(4), 299–301 (2004)
29. Tutte, W.: Connectivity in graphs, Mathematical expositions, vol. 15. University of Toronto Press, Toronto (1966)
30. Wahlström, M.: Half-integrality, LP-branching and FPT algorithms. In: Proceedings of SODA'14, pp. 1762–1781. SIAM (2014)