Beating the 2-approximation factor for Global Bicut*

Kristóf Bérczi[†‡] Karthekeyan Chandrasekaran[§] Tamás Király [†‡] Euiwoong Lee[¶]
Chao Xu [§‖]

Abstract

In the fixed-terminal bicut problem, the input is a directed graph with two specified nodes and the goal is to find a smallest subset of edges whose removal ensures that the two specified nodes cannot reach each other. In the global bicut problem, the input is a directed graph and the goal is to find a smallest subset of edges whose removal ensures that *there exist* two nodes that cannot reach each other. Fixed-terminal bicut and global bicut are natural extensions of $\{s,t\}$-min cut and global min-cut respectively, from undirected graphs to directed graphs. Fixed-terminal bicut is NP-hard, admits a simple 2-approximation, and does not admit a $(2-\epsilon)$-approximation for any constant $\epsilon > 0$ assuming the unique games conjecture. In this work, we show that global bicut admits a $(2-1/448)$-approximation, thus improving on the approximability of the global variant in comparison to the fixed-terminal variant.

1 Introduction

The global minimum cut problem is a classic interdiction problem that admits efficient algorithms in undirected graphs. In this work, we study the following generalization of the global minimum cut problem from undirected graphs to directed graphs:

BiCut: Given a directed graph, find a smallest subset of edges whose removal ensures that *there exist* two distinct nodes that cannot reach each other.

A natural approach to solving BiCut is by iterating over all pairs of distinct nodes s and t in the input graph and solving the following fixed-terminal bicut problem:

$\{s,t\}$**-BiCut:** Given a directed graph with two specified terminal nodes s,t, find a smallest subset of edges whose removal ensures that s and t cannot reach each other.

Clearly, $\{s,t\}$-BiCut is equivalent to 2-terminal multiway-cut in directed graphs (the goal in k-terminal multiway cut is to remove a smallest subset of edges to ensure that the given k terminals cannot reach each other). A classic result by Garg, Vazirani and Yannakakis shows that $\{s,t\}$-BiCut is NP-hard [8]. A simple 2-approximation algorithm is to return the union of a minimum $s \to t$ cut and a minimum $t \to s$ cut in the input directed graph. The approximability of $\{s,t\}$-BiCut has seen renewed interest in the last few months culminating in inapproximability results matching the best-known approximability factor [3, 14]: $\{s,t\}$-BiCut has no efficient $(2-\epsilon)$-approximation for any constant $\epsilon > 0$ assuming the Unique Games Conjecture [13]. These results suggest that we have a very good understanding of the complexity and the approximability of the fixed-terminal variant, i.e., $\{s,t\}$-BiCut. In contrast, even the complexity of the global variant, i.e., BiCut, is still an open problem.

The motivations for studying BiCut are multifold. In several network defense/attack applications, global cuts and connectivity are much more important than connectivity between fixed pairs of terminals. On the one hand, BiCut is a fundamental global cut problem with interdiction applications involving directed graphs. On the other hand, there is no known complexity theoretic result for BiCut. The fundamental nature of the problem coupled with the lack of basic tractability results are compelling reasons to investigate this problem.

*A preliminary version of this work appeared in the 20th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems (APPROX 2017).

†MTA-ELTE Egerváry Research Group, Department of Operations Research,Eötvös Loránd University, Budapest {berkri,tkiraly}@cs.elte.hu

‡Kristóf and Tamás are supported by the Hungarian National Research, Development and Innovation Office – NKFIH grants K109240 and K120254 and by the ÚNKP-17-4 New National Excellence Program of the Ministry of Human Capacities.

§University of Illinois at Urbana-Champaign {karthe,chaoxu3}@illinois.edu

¶Carnegie Mellon University, Pittsburgh euiwoonl@cs.cmu.edu

‖Chao is supported in part by NSF grant CCF-1526799.

Furthermore, BiCut is an ideal candidate problem to study towards understanding whether cut problems exhibit a dichotomous behaviour between global and fixed-terminal variants in directed graphs. For concreteness, we recall the 3-Cut problem and the 3-way-Cut problem in undirected graphs. In 3-Cut, the input is an undirected graph and the goal is to find a smallest subset of edges whose removal ensures that there exist 3 nodes that cannot reach each other. In 3-way-Cut, the input is an undirected graph with 3 specified nodes and the goal is to find a smallest subset of edges whose removal ensures that the 3 specified nodes cannot reach each other. While the global variant, namely 3-Cut, admits an efficient algorithm [9, 12], the fixed-terminal variant, namely 3-way-Cut, is NP-hard [6]. Such a dichotomy in complexity/approximability between global and fixed-terminal variants is hardly understood in directed graphs. In this work, we exhibit such a dichotomy for directed graphs by focusing on BiCut.

1.1 Results

In spite of an extensive body of literature on cut problems in directed graphs, the complexity of BiCut is still an open problem. In this work, we exhibit a dichotomy in the approximability between BiCut and $\{s,t\}$-BiCut. While $\{s,t\}$-BiCut is inapproximable to a constant factor better than 2 assuming UGC, we show that BiCut is approximable to a constant factor that is strictly better than 2. The following is our main result:

Theorem 1.1 *There exists an efficient* $(2-1/448)$*-approximation algorithm for* BiCut.

We emphasize that the complexity of BiCut is still an open problem.

Additional Results on Sub-problems. As a sub-problem in the algorithm for Theorem 1.1, we consider the following problem:

$(s,*,t)$-Lin-3-Cut (abbreviating linear 3-cut)**:** Given a directed graph $D=(V,E)$ and two specified nodes $s,t\in V$, find a smallest subset of edges to remove so that there exists a node r with the property that s cannot reach r and t, and r cannot reach t in the resulting graph.

$(s,*,t)$-Lin-3-Cut is a global variant of (s,r,t)-Lin-3-Cut, introduced in [7], where the input specifies three terminals s,r,t and the goal is to find a smallest subset of edges whose removal achieves the property above. A simple reduction from 3-way-Cut shows that (s,r,t)-Lin-3-Cut is NP-hard. The approximability of (s,r,t)-Lin-3-Cut was studied by Chekuri and Madan [3]. They showed that the inapproximability factor coincides with the flow-cut gap of an associated *path-blocking linear program* assuming the Unique Games Conjecture. However, the exact approximability factor is still unknown. On the positive side, there exists a simple combinatorial 2-approximation algorithm for (s,r,t)-Lin-3-Cut.

A 2-approximation for $(s,*,t)$-Lin-3-Cut can be obtained by iterating over all choices for the terminal r and using the above-mentioned 2-approximation for (s,r,t)-Lin-3-Cut. However, for the purposes of getting a strictly better than 2-approximation for BiCut, we need a strictly better than 2-approximation for $(s,*,t)$-Lin-3-Cut. We obtain the following improved approximation factor:

Theorem 1.2 *There exists an efficient* $3/2$*-approximation algorithm for* $(s,*,t)$-Lin-3-Cut.

We emphasize that, similar to BiCut, we do not know if $(s,*,t)$-Lin-3-Cut is NP-hard. Upon encountering cut problems in directed graphs whose complexity is difficult to determine, it is often insightful to consider the complexity of the analogous problem in undirected graphs. Our next result shows that the undirected counterpart of $(s,*,t)$-Lin-3-Cut is in fact solvable in polynomial time. We observe that reachability in undirected graphs is a symmetric property: if a node s can reach another node t, then the node t can also reach the node s. Hence, the analogous problem in undirected graphs is the following: given an undirected graph with two specified nodes s,t, remove a smallest subset of edges so that the resulting graph has at least 3 connected components with s and t being in different components. More generally, we consider the following:

$\{s,t\}$-Sep-k-Cut: Given an *undirected graph* $G=(V,E)$ with two specified nodes $s,t\in V$, find a smallest subset of edges to remove so that the resulting graph has at least k connected components with s and t being in different components.

The complexity of $\{s,t\}$-Sep-k-Cut for constant k was posed as an open problem by Queyranne [16]. In this work, we resolve this open problem by showing that $\{s,t\}$-Sep-k-Cut is solvable in polynomial-time for every constant k.

Theorem 1.3 *For every constant k, there is an efficient algorithm to solve* $\{s,t\}$-Sep-k-Cut.

Organization. We set the notation and discuss another cut problem which is useful as a subproblem in our algorithm in Section 1.3. We prove Theorems 1.2 and 1.3 in Section 2 and Theorem 1.1 in Section 3.

1.2 Related Work

In spite of an extensive literature on cut problems, we are unaware of any work on BICUT. We mention some work related to the other two problems mentioned in the previous section. (s,r,t)-LIN-3-CUT was introduced by Erbacher et al. in [7]. They showed that the problem is fixed-parameter tractable when parameterized by the size of the solution.

k-CUT is a well-known partitioning problem in undirected graphs with a rich history. In k-CUT, the input is an undirected graph and the goal is to find a smallest subset of edges to remove so that the resulting graph has at least k connected components. When k is part of the input, this is NP-hard [9] and admits a 2-approximation [17]. When k is a constant, this is solvable in polynomial time [9, 12, 19].

The fixed-terminal variant of k-CUT is known as k-WAY-CUT. In k-WAY-CUT, the input is an undirected graph with k specified terminals $s_1,\ldots,s_k$ and the goal is to find a smallest subset of edges to remove so that no two terminals can reach each other in the resulting graph. It is well-known that k-WAY-CUT is NP-hard [6]. For $k=3$, a 12/11-approximation is known [4, 10], while for constant k, the current-best approximation factor is 1.2975 due to Sharma and Vondrák [18]. These results are based on an LP-relaxation proposed by Călinescu, Karloff and Rabani [5], known as the CKR relaxation. Manokaran, Naor, Raghavendra and Shwartz [15] showed that the inapproximability factor coincides with the integrality gap of the CKR relaxation. Recently, Angelidakis, Makarychev and Manurangsi [1] exhibited instances with integrality gap at least $6/(5+(1/k-1))-\epsilon$ for every $k\geq 3$ and every $\epsilon>0$ for the CKR relaxation.

1.3 Preliminaries

We recall another cut problem in digraphs that is used as a subproblem in our algorithm. Given a directed graph $D=(V,E)$, we call a node to be a *source* if it can reach every other node in D. The following subproblem is used in our algorithm:

DOUBLECUT: Given a directed graph, find a smallest subset of edges to remove so that the resulting graph has no source node.

DOUBLECUT is also an extension of global minimum cut from undirected graphs to directed graphs. The tractability of DOUBLECUT is folklore (e.g., see [2]). We will need the specific structure of an optimal solution to DOUBLECUT. The following characterization of directed graphs with no source node shows the needed structure:

Theorem 1.4 *(E.g., see [2]) Let $D=(V,E)$ be a directed graph. The following are equivalent:*

1. *D has no source node.*
2. *There exist two disjoint non-empty sets $S,T\subseteq V$ with $\delta^{in}(S)\cup\delta^{in}(T)=\emptyset$.*

From the above theorem, we conclude that every optimal solution to DOUBLECUT is given by the incoming edges of two disjoint non-empty subsets of nodes. The efficient algorithms for solving DOUBLECUT can be used to obtain such a pair of sets.

Notations. Let $D=(V,E)$ be a directed graph. For two disjoint sets $X,Y\subseteq V$, we denote $\delta_D(X,Y)$ to be the set of edges (u,v) with $u\in X$ and $v\in Y$ and $d(X,Y)$ to be the cut value $|\delta_D(X,Y)|$. We use $\delta^{in}_D(X):=\delta_D(V\setminus X,X)$, $\delta^{out}_D(X):=\delta(X,V\setminus X)$, $d^{in}_D(X):=|\delta^{in}_D(X)|$ and $d^{out}_D(X):=|\delta^{out}(X)|$. We drop the subscripts when the graph D is clear from context. We use a similar notation for undirected graphs by dropping the superscripts *in* and *out*. For two nodes $s,t\in V$, a subset $X\subseteq V$ is an $\bar{s}t$*-set* if $t\in X\subseteq V-s$. The *cut value* of an $\bar{s}t$-set X is $d^{in}(X)$. For two sets $A,B\subseteq V$, let

$$\beta(A,B):=|\delta^{in}(A)\cup\delta^{in}(B)|,\text{ and}$$
$$\sigma(A,B):=|\delta^{in}(A)|+|\delta^{in}(B)|.$$

2 LIN3CUT problems

In this section, we prove Theorems 1.2 and 1.3. Theorem 1.2 gives a 3/2-approximation for $(s,*,t)$-LIN-3-CUT and is a necessary component of our proof of Theorem 1.1. Theorem 1.3 is an investigation of $(s,*,t)$-LIN-3-CUT in undirected graphs and answers an open problem posed by Queyranne [16].

2.1 A 3/2-approximation for $(s,*,t)$-LIN-3-CUT

One of our main tools used in the approximation algorithm for BICUT is a 3/2-approximation algorithm for $(s,*,t)$-LIN-3-CUT. We present this algorithm now. We recall the problem $(s,*,t)$-LIN-3-CUT: Given a directed graph with specified nodes s,t, find a smallest subset of edges whose removal ensures that the graph contains a node r with the property that s cannot reach r and t, and r cannot reach t.

Notations. Let V be the node set of a graph. A family $\mathcal{C}$ of subsets of V is a *chain* if for every pair of sets $A,B \in \mathcal{C}$, we have $A \subseteq B$ or $B \subseteq A$. We observe that a chain family can have at most $|V|$ non-empty sets. Two sets A and B are *uncomparable* if $A \setminus B$ and $B \setminus A$ are non-empty. A set A is *compatible* with a chain $\mathcal{C}$ if $\mathcal{C} \cup \{A\}$ is a chain, and it is *not compatible* otherwise.

We first rephrase the problem in a convenient way.

Lemma 2.1 $(s,*,t)$-LIN-3-CUT *in a directed graph* $D=(V,E)$ *is equivalent to*

$$\min\{\beta(A,B) : t \in A \subsetneq B \subseteq V - \{s\}\}.$$

Proof: Let $F \subseteq E$ be an optimal solution for $(s,*,t)$-LIN-3-CUT in D and let

$$(A,B) := \operatorname{argmin}\{\beta(A,B) : t \in A \subsetneq B \subseteq V - s\}.$$

Let us fix an arbitrary node $r \in B - A$. Since the deletion of $\delta^{in}(A) \cup \delta^{in}(B)$ results in a graph with no directed path from s to r, from r to t and from s to t, the edge set $\delta^{in}(A) \cup \delta^{in}(B)$ is a feasible solution to (s,r,t)-LIN-3-CUT in D, thus implying that $|F| \le \beta(A,B)$.

On the other hand, F is a feasible solution for (s,r',t)-LIN-3-CUT in D for some $r' \in V - \{s,t\}$. Let A' be the set of nodes that can reach t in $D-F$, and R' be the set of nodes that can reach r' in $D-F$. Then, $F \supseteq \delta^{in}(A')$. Moreover, $F \supseteq \delta^{in}(R' \cup A')$ since $R' \cup A'$ has in-degree 0 in $D-F$, and s is not in $R' \cup A'$ because it cannot reach r' and t in $D-F$. Therefore, taking $B' = R' \cup A'$ we get $F \supseteq \delta^{in}(A') \cup \delta^{in}(B')$. □

The above reformulation shows that the optimal solution is given by a chain consisting of two $\bar{s}t$-sets. The following lemma shows that we can obtain a 3/2 approximation to the required chain.

Lemma 2.2 *There exists an efficient algorithm that given a directed graph* $D=(V,E)$ *with nodes* $s,t \in V$ *returns a pair of* $\bar{s}t$*-sets* $A \subsetneq B \subseteq V$ *such that*

$$\beta(A,B) \le \frac{3}{2}\min\{\beta(A,B) : t \in A \subsetneq B \subseteq V - \{s\}\}.$$

Proof: The objective is to find a chain of two $\bar{s}t$-sets A, B with minimum $\beta(A,B)$. To obtain an approximation, we build a chain $\mathcal{C}$ of $\bar{s}t$-sets with the property that, for some value $k \in \mathbb{Z}_+$,

(i) every set $C \in \mathcal{C}$ is a $\bar{s}t$-set with $d^{in}(C) \le k$, and

(ii) every $\bar{s}t$-set T with $d^{in}(T)$ strictly less than k is in $\mathcal{C}$.

We use the following procedure to obtain such a chain: We initialize with k being the minimum $\bar{s}t$-cut value and $\mathcal{C}$ consisting of a single minimum $\bar{s}t$-cut. In a general step, we find two $\bar{s}t$-sets: a $\bar{s}t$-set Y compatible with the current chain $\mathcal{C}$, i.e. $\mathcal{C} \cup \{Y\}$ forming a chain, with minimum $d^{in}(Y)$ and a $\bar{s}t$-set Z *not* compatible with the current chain $\mathcal{C}$, i.e. crossing at least one member of $\mathcal{C}$, with minimum $d^{in}(Z)$. We will later see that such sets Y and Z can be found in polynomial time. If $d^{in}(Y) \le d^{in}(Z)$, then we add Y to $\mathcal{C}$, and set k to $d^{in}(Y)$; otherwise we set k to $d^{in}(Z)$, and stop.

Proposition 2.3 *Let* $\mathcal{C}$ *denote the chain before any general step of the above-mentioned procedure. Then, for every* $C \in \mathcal{C}$ *and for every* $\bar{s}t$*-set* A *that is not in* $\mathcal{C}$*, we have*

$$d^{in}(A) \ge d^{in}(C).$$

Proof: Let A be a $\bar{s}t$-set that is not in $\mathcal{C}$. Suppose for the sake of contradiction that $d^{in}(A) < d^{in}(C)$ for some $C \in \mathcal{C}$. Let $\mathcal{C}'$ denote the chain consisting of those members of $\mathcal{C}$ that were added before C. Since $A \notin \mathcal{C}$ and C is a set of minimum cut value compatible with $\mathcal{C}'$, we have that A should cross at least one member of $\mathcal{C}'$. Hence, by $d^{in}(A) < d^{in}(C)$, the procedure stops before adding C to the chain $\mathcal{C}'$, a contradiction. □

Proposition 2.4 *The chain $\mathcal{C}$ and the value k obtained at the end of the above-mentioned procedure satisfy* (i) *and* (ii).

Proof: The construction immediately guarantees that every set $C \in \mathcal{C}$ is a $\bar{s}t$-set. By Proposition 2.3 and by construction of $\mathcal{C}$ and k, we have that $d^{in}(C) \le k$ for every $C \in \mathcal{C}$ and hence, we have (i).

By construction, $\mathcal{C}$ contains all $\bar{s}t$-sets T that are compatible with $\mathcal{C}$ with $d^{in}(T) < k$. Suppose for the sake of contradiction, we have an $\bar{s}t$-set T with $d^{in}(T) < k$ that is not in $\mathcal{C}$. Then, the set T should be incompatible with $\mathcal{C}$. We note that the procedure terminates by setting $k = d^{in}(Z)$ for some Z that is incompatible with $\mathcal{C}$. However, the set T is a contradiction to the choice of Z in the procedure. Therefore, there does not exist a $\bar{s}t$-set T with $d^{in}(T) < k$ that is not in $\mathcal{C}$ and hence, we have (ii). $\square$

By the above, the procedure stops with a chain $\mathcal{C}$ containing all $\bar{s}t$-sets of cut value less than k, and an $\bar{s}t$-set Z of cut value exactly k which crosses some member X of $\mathcal{C}$. If the optimum value of our problem is less than k, then both members of the optimal pair (A, B) belong to the chain $\mathcal{C}$, and we can find them by taking the minimum of $\beta(A', B')$ where $A' \subseteq B'$ with $A', B' \in \mathcal{C}$.

We can thus assume that the optimum is at least k. Since $d^{in}(Z) = k$ and $d^{in}(X) \le k$, the submodularity of the in-degree function implies

$$d^{in}(X \cap Z) + d^{in}(X \cup Z) \le d^{in}(Z) + d^{in}(X) \le 2k.$$

Hence either $d^{in}(X \cap Z) \le k$ or $d^{in}(X \cup Z) \le k$. Since

$$d(X \setminus Z, X \cap Z) + d(Z \setminus X, X \cap Z) \le d^{in}(X \cap Z) \text{ and}$$
$$d(V \setminus (X \cup Z), X \setminus Z) + d(V \setminus (X \cup Z), Z \setminus X) \le d^{in}(X \cup Z),$$

at least one of the following four possibilities holds:

1. $d^{in}(X \cap Z) \le k$ and $d(X \setminus Z, X \cap Z) \le \frac{1}{2}k$. Choose $A = X \cap Z$, $B = X$. Then $\beta(A, B) = d(X \setminus Z, X \cap Z) + d^{in}(X) \le \frac{1}{2}k + k = \frac{3}{2}k$.

2. $d^{in}(X \cap Z) \le k$ and $d(Z \setminus X, X \cap Z) \le \frac{1}{2}k$. Choose $A = X \cap Z$, $B = Z$. Then $\beta(A, B) = d(Z \setminus X, X \cap Z) + d^{in}(Z) \le \frac{1}{2}k + k = \frac{3}{2}k$.

3. $d^{in}(X \cup Z) \le k$ and $d(V \setminus (X \cup Z), X \setminus Z) \le \frac{1}{2}k$. Choose $A = Z$, $B = X \cup Z$. Then $\beta(A, B) = d^{in}(Z) + d(V \setminus (X \cup Z), X \setminus Z) \le k + \frac{1}{2}k = \frac{3}{2}k$.

4. $d^{in}(X \cup Z) \le k$ and $d(V \setminus (X \cup Z), Z \setminus X) \le \frac{1}{2}k$. Choose $A = X$, $B = X \cup Z$. Then $\beta(A, B) = d^{in}(X) + d(V \setminus (X \cup Z), Z \setminus X) \le k + \frac{1}{2}k = \frac{3}{2}k$.

Thus a pair (A, B) can be obtained by taking the minimum among the four possibilities above and $\beta(A', B')$ where $A' \subseteq B'$ with $A', B' \in \mathcal{C}$, concluding the proof of the approximation factor. It remains to ensure that the algorithm can be implemented to run in polynomial-time.

The algorithm is summarized below. Step 2(a) to obtain Y can be implemented to run in polynomial-time as follows: let $t \in C_1 \subseteq \ldots, \subseteq C_q \subseteq V - s$ denote the members of $\mathcal{C}$. Find a minimum cut Y_i with $C_i \subseteq Y_i \subseteq V \setminus C_{i+1}$ for $i = 1, \ldots, q$, and choose Y to be a minimum one among these cuts. Step 2(b) to obtain Z can be implemented to run in polynomial-time as follows: for each pair x, y of nodes with $y \in C_i \subseteq V - x$ for some $i \in \{1, \ldots, q\}$, find a minimum cut Z_{xy} with $\{t, x\} \subseteq Z_{xy} \subseteq V - \{s, y\}$, and choose Z to be a minimum one among these cuts. Since $\mathcal{C}$ is a chain, we have that $q \le |V|$ and hence both steps can be implemented to run in polynomial-time.

Theorem 1.2 is a consequence of Lemmas 2.1 and 2.2. The approximation algorithm is summarized below.

2.2 An exact algorithm for $\{s, t\}$-SEP-k-CUT

In this section, we show that $\{s, t\}$-SEP-k-CUT is solvable in polynomial time if k is a fixed constant. We recall the problem $\{s, t\}$-SEP-k-CUT: Given an *undirected* graph with specified nodes s, t, find a smallest subset of edges whose removal ensures that the resulting graph has at least k connected components with s and t being in different components.

Approximation Algorithm for $(s, *, t)$-LIN-3-CUT

Input: Directed graph $D = (V, E)$ with $s, t \in V$

1. Let S denote the sink-side of a minimum $s \to t$ cut and α denote its value. Initialize $\mathcal{C} \leftarrow \{S\}$ and $k \leftarrow \alpha$.

2. Repeat:

   (a) $Y \leftarrow \arg\min\{d^{in}(Y) : Y$ is a $\bar{s}t$-set compatible with $\mathcal{C}\}$

   (b) $Z \leftarrow \arg\min\{d^{in}(Z) : Z$ is a $\bar{s}t$-set not compatible with $\mathcal{C}\}$

   (c) If $d^{in}(Y) \le d^{in}(Z)$, then update $\mathcal{C} \leftarrow \mathcal{C} \cup \{Y\}$ and $k \leftarrow d^{in}(Y)$.

   (d) Else, update $k \leftarrow d^{in}(Z)$, set X to be a set in $\mathcal{C}$ that crosses Z and go to Step 3.

3. Let $(A, B) \leftarrow \arg\min\{\beta(A, B) : A, B \in \mathcal{C}, A \neq B\}$.

4. Let $(S, T) \leftarrow \arg\min\{\beta(X \cap Z, X), \beta(X \cap Z, Z), \beta(Z, X \cup Z), \beta(X, X \cup Z)\}$

5. Return $\arg\min\{\beta(A, B), \beta(S, T)\}$.

Notations. Let $G = (V, E)$ be an undirected graph. Let the minimum size of an $\{s, t\}$-cut in G be denoted by $\lambda_G(s, t)$. For two subsets of nodes X, Y, we recall that $d(X, Y)$ denotes the number of edges between X and Y and that $d(X) = d(X, V \setminus X)$. The *cut value* of a partition $\{V_1, \ldots, V_q\}$ of V is defined to be the total number of crossing edges, that is, $(1/2)\sum_{i=1}^{q} d(V_i)$, and is denoted by $\gamma(V_1, \ldots, V_q)$. Let $\gamma^q(G)$ denote the value of an optimum q-CUT in G, i.e.,

$$\gamma^q(G) := \min\{\gamma(V_1, \ldots, V_q) : V_i \neq \emptyset \ \forall\ i \in [q], \quad V_i \cap V_j = \emptyset \ \forall\ i, j \in [q], \cup_{i=1}^{q} V_i = V\}.$$

Proof (Proof of Theorem 1.3): Let γ^* denote the optimum value of $\{s, t\}$-SEP-k-CUT in $G = (V, E)$ and let H denote the graph obtained from G by adding an edge of infinite capacity between s and t. The algorithm is based on the following observation (we recommend the reader to consider $k = 3$ for ease of understanding):

Proposition 2.5 *Let $\{V_1, \ldots, V_k\}$ be a partition of V corresponding to an optimal solution of $\{s, t\}$-SEP-k-CUT, where s is in V_{k-1} and t is in V_k. Then $\gamma(V_1, \ldots, V_{k-2}, V_{k-1} \cup V_k) \le 2\gamma^{k-1}(H)$.*

Proof: Let $W_1, \ldots, W_{k-1}$ be a minimum $(k-1)$-cut in H. Clearly, s and t are in the same part, so we may assume that they are in W_{k-1}. Let U_1, U_2 be a minimum $\{s, t\}$-cut in $G[W_{k-1}]$. Then $\{W_1, \ldots, W_{k-2}, U_1, U_2\}$ gives an $\{s, t\}$-separating k-cut, showing that

$$\gamma^* \le \gamma(W_1, \ldots, W_{k-2}, U_1, U_2) = \gamma^{k-1}(H) + \lambda_{G[W_{k-1}]}(s, t). \tag{1}$$

By Menger's theorem, we have $\lambda_G(s, t)$ pairwise edge-disjoint paths $P_1, \ldots, P_{\lambda_G(s,t)}$ between s and t in G. Consider one of these paths, say P_i. If all nodes of P_i are from $V_{k-1} \cup V_k$, then P_i has to use at least one edge from $\delta(V_{k-1}, V_k)$. Otherwise, P_i uses at least two edges from $\delta(V_1 \cup \cdots \cup V_{k-2}) \cup \bigcup_{\substack{i,j \le k-2 \\ i \neq j}} \delta(V_i, V_j)$. Hence the maximum number of pairwise edge-disjoint paths between s and t is

$$\lambda_G(s, t) \le d(V_{k-1}, V_k) + \frac{1}{2}\left(d(V_1 \cup \cdots \cup V_{k-2}) + \sum_{\substack{i,j \le k-2 \\ i \neq j}} d(V_i, V_j) \right).$$

Thus, we have

$$\begin{aligned}
\gamma^* &= d(V_{k-1}, V_k) + d(V_1 \cup \cdots \cup V_{k-2}) + \sum_{\substack{i,j \le k-2 \\ i \ne j}} d(V_i, V_j) \\
&\ge \lambda_G(s,t) + \frac{1}{2}\left(d(V_1 \cup \cdots \cup V_{k-2}) + \sum_{\substack{i,j \le k-2 \\ i \ne j}} d(V_i, V_j) \right) \\
&= \lambda_G(s,t) + \frac{1}{2}\gamma(V_1, \ldots, V_{k-2}, V_{k-1} \cup V_k) \\
&\ge \lambda_{G[W_{k-1}]}(s,t) + \frac{1}{2}\gamma(V_1, \ldots, V_{k-2}, V_{k-1} \cup V_k)
\end{aligned}$$

that is,

$$\gamma^* \ge \lambda_{G[W_{k-1}]}(s,t) + \frac{1}{2}\gamma(V_1, \ldots, V_{k-2}, V_{k-1} \cup V_k). \tag{2}$$

By combining (1) and (2), we get $\gamma(V_1, \ldots, V_{k-2}, V_{k-1} \cup V_k) \le 2\gamma^{k-1}(H)$, proving the proposition. □

Karger and Stein [12] showed that the number of feasible solutions to k-CUT in an undirected graph G with value at most $2\gamma^k(G)$ is $O(n^{4k})$. All these solutions can be enumerated in polynomial-time for fixed k [11, 12, 20]. This observation together with Proposition 2.5 gives the algorithm for finding an optimal solution to $\{s,t\}$-SEPEDGEkCUT. The algorithm is summarized below.

Algorithm for $\{s,t\}$-SEP-k-CUT

Input: Undirected graph $G = (V, E)$ with $s, t \in V$

1. Let H be the graph obtained from G by adding an edge of infinite capacity between s and t. In H, enumerate all feasible solutions to EDGE-$(k-1)$-CUT—namely the vertex partitions $\{W_1, \ldots, W_{k-1}\}$—whose cut value $\gamma_H(W_1, \ldots, W_{k-1})$ is at most $2\gamma^{k-1}(H)$. Without loss of generality, assume $s, t \in W_{k-1}$.
2. For each feasible solution to EDGE-$(k-1)$-CUT in H listed in Step 1, find a minimum $\{s,t\}$-cut in $G[W_{k-1}]$, say U_1, U_2.
3. Among all feasible solutions $\{W_1, \ldots, W_{k-1}\}$ to EDGE-$(k-1)$-CUT listed in Step 1 and the corresponding U_1, U_2 found in Step 2, return the k-cut $\{W_1, \ldots, W_{k-2}, U_1, U_2\}$ with minimum $\gamma(W_1, \ldots, W_{k-2}, U_1, U_2)$.

The correctness of the algorithm follows from Proposition 2.5: one of the choices enumerated in Step 1 will correspond to the partition $(V_1, \ldots, V_{k-2}, V_{k-1} \cup V_k)$, where $(V_1, \ldots, V_k)$ is the partition corresponding to the optimal solution.

3 BICUT

In this section, we present our approximation algorithm (Theorem 1.1) for BICUT. We begin with the high-level ideas of the approximation algorithm in Section 3.1. The full algorithm and the proof of its approximation ratio are presented in Section 3.2.

We recall the problem BICUT: Given a directed graph, find a smallest number of edges in whose removal ensures that there exist two distinct nodes s and t such that s cannot reach t and t cannot reach s. We begin with a reformulation of BICUT that is helpful for the purposes of designing an algorithm. We recall that for two sets of nodes A, B, the quantity $\beta(A,B) = |\delta^{in}(A) \cup \delta^{in}(B)|$.

Definition We define two sets A and B to be *uncomparable* if $A \setminus B \ne \emptyset$ and $B \setminus A \ne \emptyset$. For a directed graph $D = (V, E)$, let

$$\beta := \min\{\beta(A,B) : A \text{ and } B \text{ are uncomparable}\}.$$

The following lemma shows that bicut is equivalent to finding an uncomparable pair of subsets of nodes A, B with minimum $\beta(A,B)$.

Lemma 3.1 BiCut *in a given directed graph $D=(V,E)$ is equal to β.*

Proof: If A and B are uncomparable and we remove $\delta^{in}(A)\cup\delta^{in}(B)$ from the directed graph, then nodes in $A\setminus B$ cannot reach nodes in $B\setminus A$ and vice versa. On the other hand, if s cannot reach t and t cannot reach s, then the set of nodes that can reach s and the set of nodes that can reach t are uncomparable, and have in-degree 0. □

Using the above formulation, and by recalling that $\sigma(A,B)=|\delta^{in}(A)|+|\delta^{in}(B)|$, we have the following natural relaxation of bicut:

Definition For a directed graph $D=(V,E)$, let

$$\sigma := \min\{\sigma(A,B):\ A \text{ and } B \text{ are uncomparable}\}.$$

A pair where the latter value is attained is called a *minimum uncomparable cut-pair*.

3.1 Overview of the Approximation Algorithm

In this section, we sketch the argument for a $(2-\varepsilon)$-approximation for some small enough ε. We observe that for every pair of subsets of nodes (A,B), we have

$$\beta(A,B)=\sigma(A,B)-d(V\setminus(A\cup B),A\cap B). \tag{3}$$

Therefore, $\beta(A,B)\le\sigma(A,B)\le 2\beta(A,B)$ for every pair of subsets of nodes (A,B) and hence $\beta\le\sigma\le 2\beta$. Furthermore, σ can be computed efficiently (see Lemma 3.2). Hence, we immediately have a $(2-\varepsilon)$-approximation if $\sigma\le(2-\varepsilon)\beta$. On the other hand, if $\sigma>(2-\varepsilon)\beta$, then $d(V\setminus(A\cup B),A\cap B)>(1-\epsilon)\beta$ for every minimizer (A,B) of $\beta(A,B)$, thus providing a structural handle on optimal solutions. Our algorithm proceeds by making several attempts at finding pairs (A',B') that could give a $(2-\varepsilon)$-approximation. Each attempt that is unsuccessful at giving a $(2-\varepsilon)$-approximation implies some structural property of the optimal solution. These structural properties are together exploited by the last attempt to succeed.

Our next attempt is to solve a constrained variant of BiCut: For fixed $Z\subseteq V$, we would like to find an uncomparable pair (A,B) satisfying $A\cap B=Z$ that minimizes $\beta(A,B)$ among pairs with this property. This problem is solvable efficiently by reducing to DoubleCut (see Lemma 3.3). The same holds when $V\setminus(A\cup B)$ is fixed. In particular, if there is a pair (A,B) that minimizes $\beta(A,B)$ and $|A\cap B|\le 2$ or $|V\setminus(A\cup B)|\le 2$, then we can find the minimizer efficiently. Therefore we assume that every minimizer (A,B) for $\beta(A,B)$ satisfies $|A\cap B|\ge 3$ and $|V\setminus(A\cup B)|\ge 3$. Let us fix one such minimizer (A,B).

In the algorithm, we guess nodes $x\in A\setminus B$, $y\in B\setminus A$, $w_1,w_2\in V\setminus(A\cup B)$, and $z_1,z_2\in A\cap B$. The reason for guessing *two* nodes as opposed to just one node in the intersection and in the complement of the union is highly technical (it certifies a detailed structural property of the minimizer (A,B)), and is not relevant to this overview. We use the notation $X:=A\setminus B$, $Y:=B\setminus A$, $W:=V\setminus(A\cup B)$, and $Z:=A\cap B$ (see Figure 3.1).

We now observe that A is the sink-side of a $\{w_1,w_2,y\}\to\{x,z_1,z_2\}$-cut while B is the sink-side of a $\{w_1,w_2,x\}\to\{y,z_1,z_2\}$-cut. Our next attempt in the algorithm is to find (X',Y'), where X' is the sink-side of a minimum $\{w_1,w_2,y\}\to\{x,z_1,z_2\}$-cut, and Y' is the sink-side of the minimum $\{w_1,w_2,x\}\to\{y,z_1,z_2\}$-cut. The hope behind this attempt is that X' could be A and Y' could be B as these are feasible solutions to the respective problems and thus, they would together help us recover the optimal solution. Unfortunately, this favorable best-case scenario may not happen. Yet, owing to the feasibility of A and B for the respective problems, we may conclude that $\sigma(X',Y')\le\sigma(A,B)\le 2\beta(A,B)=2\beta$.

Our subsequent attempts are more complex and proceed by refining X' and Y'. For our next attempt, we observe that Z is the sink-side of a $\{w_1,w_2,x,y\}\to\{z_1,z_2\}$-cut. So, our next attempt in the algorithm would be to find Z' as the sink-side of a minimum $\{w_1,w_2,x,y\}\to\{z_1,z_2\}$-cut and expand X' and Y' by Z' to obtain an uncomparable pair $(A'=X'\cup Z', B'=Y'\cup Z')$. Our hope is to find a Z' so that the resulting $\beta(A',B')$ is small. While finding Z', we prefer not to have many edges of $E[X']\cup E[Y']$ in the new bicut (A',B'). This is because, such edges enter only one among the two sets A' and B'. We recall that if we have an uncomparable pair (A',B') with lot of edges from $V\setminus(A'\cup B')$ to $A'\cap B'$, then the value of $\beta(A',B')$ is going to be much less than $\sigma(A',B')$ (e.g., see (3)), thus leading to a $(2-\epsilon)$-approximation. So, in order to avoid the edges of $E[X']\cup E[Y']$ in the new bicut (A',B'), we make such edges more expensive by duplicating them before finding Z'. Let D_1 be the digraph obtained

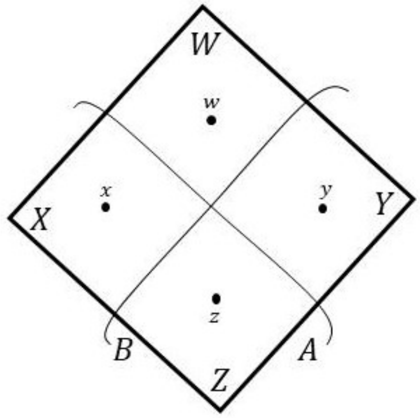

Figure 3.1. The partitioning of the node set in the graph D. Here, (A,B) denotes the optimum bicut that is fixed.

by duplicating the edges in $E[X'] \cup E[Y']$, and let Z' be the sink-side of the minimum $\{w_1, w_2, x, y\} \to \{z_1, z_2\}$-cut in D_1. We then show that the pair $(X' \cup Z', Y' \cup Z')$ is a $(2-\varepsilon)$-approximation unless $|\delta^{in}_{D_1}(Z)| > (2-3\varepsilon)\beta$, thus giving us more structural handle on the optimum solution.

We next make an analogous attempt by shrinking X' and Y' instead of expanding. Let D_2 be the digraph obtained by duplicating the edges in $E[V \setminus X'] \cup E[V \setminus Y']$, and let W' be the source-side of the minimum $\{w_1, w_2\} \to \{x, y, z_1, z_2\}$-cut in D_2. We obtain that the pair $(X' \setminus W', Y' \setminus W')$ is a $(2-\varepsilon)$-approximation unless $|\delta^{out}_{D_2}(W)| > (2-3\varepsilon)\beta$.

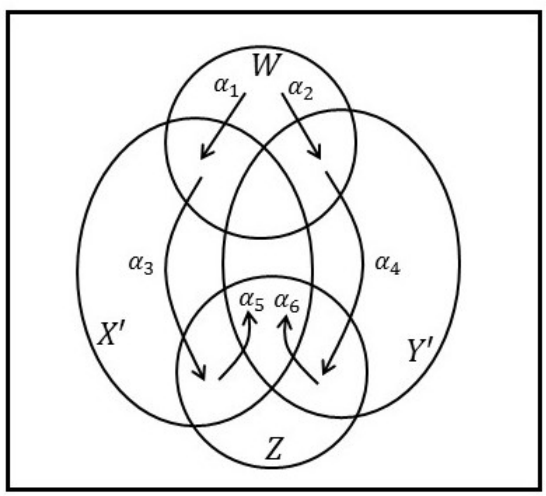

Figure 3.2. The quantities $\alpha_1, \ldots, \alpha_6$.

Let $\alpha_1, \ldots, \alpha_6$ be the number of edges in each position indicated in Figure 3.2. If the attempts so far are unsuccessful, then we use the structural properties derived so far to arrive at the following:

1. All but $O(\varepsilon\beta)$ edges in $\delta^{in}(X') \cup \delta^{in}(Y') \cup \delta^{out}(W) \cup \delta^{in}(Z)$ are as positioned in Figure 3.2.

2. The quantities $\alpha_1, \alpha_3, \alpha_5$ are within $O(\epsilon\beta)$ of each other (see (29), (30), (31)) and so are $\alpha_2, \alpha_4, \alpha_6$.

3. Furthermore, $(1-O(\epsilon))\beta = \alpha_3 + \alpha_4 \le \beta$ (see Proposition 3.9).

Without loss of generality, we may assume $\alpha_3 \ge \alpha_4$. Hence, by conclusion (3) from above, we have that $\alpha_3 \ge \beta/2 - O(\epsilon)\beta$.

Our final attempt in the algorithm to obtain a $(2-\epsilon)$-approximate bicut is to expand Y' by including some nodes from $X' \setminus Y'$ and to shrink X' by excluding some nodes from $X' \setminus Y'$. We now explain the motivation behind this choice of expanding and shrinking. Consider $S := Y' \cup (X' \cap Z)$, which is obtained by expanding Y' by including some nodes from $X' \setminus Y'$ and $T := X' \setminus (X' \cap (W \setminus Y'))$, which is obtained by shrinking X' by excluding some nodes from $X' \setminus Y'$ (see figure 3.3). By definition, (S, T) is an uncomparable pair. We will now see that the bicut value of (S, T) is much smaller than 2β. Using conclusions (1) and (2) from above, we obtain that

$$\begin{aligned}
\beta(S,T) &= |\delta^{in}(Y' \cup (X' \cap Z)) \cup \delta^{in}(X' \setminus (X' \cap (W \setminus Y')))| \\
&= |\delta^{in}(Y')| - \alpha_5 + \alpha_3 + |\delta^{in}(X')| - \alpha_1 + O(\epsilon)\beta && (4)\\
&= \sigma(X', Y') - \alpha_1 - \alpha_5 + \alpha_3 + O(\epsilon)\beta && (5)\\
&\le 2\beta - \alpha_3 + O(\epsilon)\beta && (6)\\
&\le \frac{3}{2}\beta + O(\epsilon)\beta. && (7)
\end{aligned}$$

In the above, equation (4) is by using conclusion (1), equation (5) is by definition of σ, inequality (6) is by using conclusion (2) and $\sigma(X', Y') \le \sigma(A, B) \le 2\beta$, and inequality (7) is because $\alpha_3 \ge \beta/2 - O(\epsilon)\beta$.

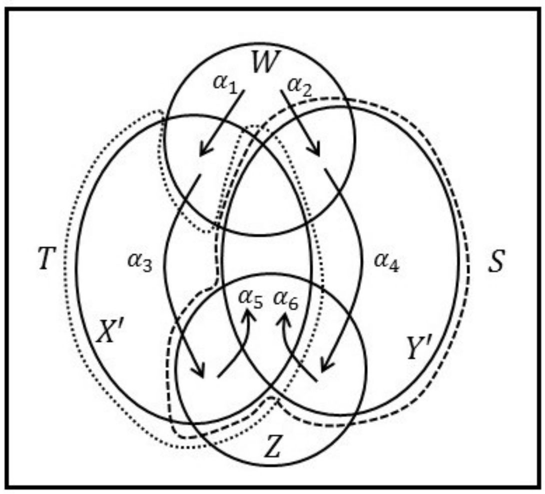

Figure 3.3. The motivation behind the last attempt.

Although (S, T) is a good approximation to the optimal bicut, we cannot obtain the sets S and T without the knowledge of W and Z (which, in turn, depend on the optimal bicut (A, B)). Instead, our algorithmic attempt is to expand Y' by including some nodes from $X' \setminus Y'$ and to shrink X' by excluding some nodes from $X' \setminus Y'$. In other words, our candidate is a pair $(B', Y' \cup A')$ for some $X' \cap Y' \subseteq A' \subsetneq B' \subseteq X'$ (we need the condition $A' \subsetneq B'$ because B' and $Y' \cup A'$ should be uncomparable). When choosing A' and B', we ignore the edges whose contribution do not depend on A' and B'. Let H be the digraph obtained by removing the edges in $E[Y' \cup (V \setminus X')]$. Our aim is to minimize $|\delta_H^{in}(B') \cup \delta_H^{in}(Y' \cup A')|$. However, this quantity differs from $|\delta_H^{in}(A') \cup \delta_H^{in}(B')|$ by $O(\varepsilon\beta)$, so we may instead aim to minimize the latter.

The crucial observation now is that this latter minimization problem is an instance of $(s, *, t)$-Lin-3-Cut. While we do not know how to solve $(s, *, t)$-Lin-3-Cut optimally, we can efficiently obtain a 3/2-approximation by Theorem 1.2. By the reformulation of $(s, *, t)$-Lin-3-Cut in Lemma 2.1, we get a pair of subsets (A', B') for which $X' \cap Y' \subseteq A' \subsetneq B' \subseteq X'$ and which is a 3/2-approximation. In particular, $|\delta_H^{in}(A') \cup \delta_H^{in}(B')| \le (3/2)|\delta_H^{in}((X' \cap (Z \cup Y')) \cup \delta_H^{in}(X' \setminus (W \setminus Y'))| \le 3(\alpha_3 + O(\epsilon)\beta)/2$. Using this and proceeding similar to the calculations shown above to obtain the bound on $\beta(S, T)$ (i.e., 4, 5, 6, and 7), we derive that $\beta(B', Y' \cup A') \le (7/4 + O(\epsilon))\beta$, concluding the proof.

3.2 Approximation Algorithm and Analysis

In this section we prove Theorem 1.1 by giving an efficient $(2-\varepsilon)$-approximation algorithm for BiCut for a constant $\varepsilon > 0$. We will describe the algorithm, analyze its approximation factor to show that it is $(2-\varepsilon)$ for some constant ε and compute the value of ε at the end of the analysis.

We begin by showing that certain relaxations of β can be solved. We first show that σ can be computed efficiently.

Lemma 3.2 *For a directed graph $D = (V, E)$, there exists a polynomial time algorithm to find a minimum uncomparable cut-pair.*

Proof: For fixed vertices a and b, there is an efficient algorithm to find A and B such that $a \in A \setminus B$ and $b \in B \setminus A$ and $\sigma(A, B)$ is minimized. Indeed, this is precisely finding the sink side of a min $a \to b$ cut and that of a min $b \to a$ cut. Trying all distinct pairs of nodes a and b and taking the minimum gives the desired result. □

We next show that we can minimize $\beta(A, B)$ among uncomparable pairs (A, B) whose intersection is fixed.

Lemma 3.3 *Given a directed graph $D = (V, E)$ and $Z \subseteq V$, there exists a polynomial time algorithm to find an uncomparable pair A, B satisfying $A \cap B = Z$ that minimizes $\beta(A, B)$ among pairs with this property.*

Proof: Let $D' = D[V \setminus Z]$ be the directed graph induced on $V \setminus Z$. We recall that DOUBLECUT can be solved in polynomial time in D' [2]; let X' and Y' be the disjoint sets whose incoming edges give the optimal double cut. We claim that the pair $X' \cup Z, Y' \cup Z$ forms a minimum bicut among all bicuts with intersection Z. Indeed, assume the optimal solution is $\beta(A, B)$. Let $X = A \setminus B$, $Y = B \setminus A$ and $W = V - (A \cup B)$. Then

$$\begin{aligned}
\beta(X' \cup Z, Y' \cup Z) &= d^{in}_{D'}(X') + d^{in}_{D'}(Y') + d^{in}(Z) \\
&\leq d^{in}_{D'}(X) + d^{in}_{D'}(Y) + d^{in}(Z) \\
&= d^{in}(Z) + d(W, X) + d(W, Y) + d(X, Y) + d(Y, X) \\
&= \beta(A, B).
\end{aligned}$$

□

A similar argument shows that we can minimize $\beta(A, B)$ among uncomparable pairs (A, B) for which the complement of the union is fixed.

Lemma 3.4 *Given a directed graph $D = (V, E)$ and $W \subseteq V$, there exists a polynomial time algorithm to find an uncomparable pair A, B satisfying $V \setminus (A \cup B) = W$ that minimizes $\beta(A, B)$ among pairs with this property.*

We need the following definition.

Definition If c is a capacity function on a directed graph D, then $d^{in}_c(U) = \sum_{e \in \delta^{in}(U)} c(e)$ is the sum of the capacities of incoming edges of U. Similarly, $d^{out}_c(U) = \sum_{e \in \delta^{out}(U)} c(e)$.

We now present the approximation algorithm and the analysis.

Proof (Proof of Theorem 1.1): The algorithm is summarized below. We first note that the algorithm indeed returns the bicut value of an uncomparable pair. The run-time of the algorithm being polynomial follows from Lemmas 2.2, 3.2, 3.3 and 3.4. In the rest of the proof, we analyze the approximation factor. We will show that the algorithm achieves a $(2 - \epsilon)$-approximation factor and compute ϵ at the end.

Approximation Algorithm for BICUT

Input: Directed graph $D = (V, E)$

1. Compute $(S, T) \leftarrow \arg\min\{\sigma(S, T) : S \text{ and } T \text{ are uncomparable}\}$ using Lemma 3.2 and set $\mu_1 \leftarrow \beta(S, T)$
2. Compute $\mu_2 \leftarrow \min\{\beta(A, B) : A \text{ and } B \text{ are uncomparable}, |A \cap B| \leq 2\}$ using Lemma 3.3
3. Compute $\mu_3 \leftarrow \min\{\beta(A, B) : A \text{ and } B \text{ are uncomparable}, |V \setminus (A \cup B)| \leq 2\}$ using Lemma 3.4
4. Initialize $\mu_4 \leftarrow \infty$
5. For each tuple of nodes $(x, y, z_1, z_2, w_1, w_2)$

(i) $X' \leftarrow$ sink-side of the minimum $\{w_1, w_2, y\} \to \{x, z_1, z_2\}$-cut

(ii) $Y' \leftarrow$ sink-side of the minimum $\{w_1, w_2, x\} \to \{y, z_1, z_2\}$-cut

(iii) $E_1 \leftarrow E[X'] \cup E[Y']$

(iv) $E_2 \leftarrow E[V \setminus X'] \cup E[V \setminus Y']$

(v) $D_1 \leftarrow D$ with the arcs in E_1 duplicated

(vi) $D_2 \leftarrow D$ with the arcs in E_2 duplicated

(vii) $Z' \leftarrow$ sink-side of minimum $\{w_1, w_2, x, y\} \to \{z_1, z_2\}$-cut in D_1

(viii) $W' \leftarrow$ source-side of minimum $\{w_1, w_2\} \to \{x, y, z_1, z_2\}$-cut in D_2

(ix) $H \leftarrow$ contract $X' \cap Y'$ to z', contract $V \setminus X'$ to w', remove all $w'z'$ arcs

(x) In H, find $\overline{w'}z'$-sets $A' \subsetneq B'$ such that $\beta(A', B')$ is at most $(3/2)\min\{\beta(A, B) : z' \in A \subsetneq B \subseteq V - \{w'\}\}$ using Lemma 2.2

(xi) $A_1 \leftarrow (A' \setminus \{z'\}) \cup (X' \cap Y')$ and $B_1 \leftarrow (B' \setminus \{z'\}) \cup (X' \cap Y')$

(xii) Find all bicuts that can be generated using set operations on X', Y', Z', W', A_1, B_1 and let μ_4' denote the minimum bicut value among these.

(xiii) If $\mu_4' < \mu_4$, update $\mu_4 \leftarrow \mu_4'$

6. Return $\mu \leftarrow \min\{\mu_1, \mu_2, \mu_3, \mu_4\}$.

To analyze the approximation factor, let us fix a minimizer (A, B) for BiCut in the input graph $D = (V, E)$, i.e. fix an uncomparable pair (A, B) such that $\beta(A, B) = \beta$. Let $X := A \setminus B$, $Y := B \setminus A$, $Z := A \cap B$, and $W := V \setminus (A \cup B)$ (see Figure 3.1). With this notation, we have

$$\beta = d(W \cup Y, X) + d(W \cup X, Y) + d^{in}(Z) = d(Y, X \cup Z) + d(X, Y \cup Z) + d^{out}(W). \qquad (8)$$

We may assume that both Z and W are of size at least 3, otherwise the algorithm finds the optimum since it returns a value $\mu \leq \mu_2, \mu_3$. Let $\varepsilon > 0$ be a constant whose value will be determined later.

Lemma 3.5 *If one of the following is true, then $\sigma \leq (2 - \varepsilon)\beta$:*

(i) $d(W, Z) \leq (1 - \varepsilon)\beta$,

(ii) For every $z_1, z_2 \in Z$, there exists a subset U of nodes containing z_1, z_2 but not Z with $d^{in}(U) < (1 - \varepsilon)\beta$.

(iii) For every $w_1, w_2 \in W$, there exists a subset U of nodes not containing w_1, w_2 but intersecting W with $d^{in}(U) < (1 - \varepsilon)\beta$.

Proof:

(i) If $d(W, Z) \leq (1 - \varepsilon)\beta$, then $\sigma(A, B) = \beta(A, B) + d(W, Z) \leq (2 - \varepsilon)\beta$. The pair (A, B) is uncomparable, and hence $\sigma \leq \sigma(A, B)$. Therefore, we have $\mu_1 = \beta(S, T) \leq \sigma(S, T) = \sigma \leq \sigma(A, B) \leq (2 - \varepsilon)\beta$.

(ii) Suppose condition *(ii)* holds. Among the sets with in-degree less than $(1 - \varepsilon)\beta$ which do not contain every node of Z, let T be the one with inclusionwise maximal intersection with Z. Such a set T exists since condition *(ii)* holds. Let $z_1 \in Z \setminus T$ and $z_2 \in Z \cap T$. There exists a set U containing z_1, z_2 but not Z with $d^{in}(U) < (1 - \varepsilon)\beta$ and $z_1, z_2 \in U$. Because of the maximal intersection of T with Z, we have that $T \nsubseteq U$. Hence T and U are uncomparable and therefore $\sigma \leq \sigma(T, U) \leq (2 - 2\varepsilon)\beta$. Therefore, we have $\mu_1 = \beta(S, T) \leq \sigma(S, T) = \sigma \leq \sigma(A, B) \leq (2 - 2\varepsilon)\beta$.

(iii) Argument similar to the proof of (ii) shows that the minimum uncomparable cut-pair is a $(2—2\varepsilon)$-approximation if condition *(iii)* holds.

$\square$

For the rest of the proof, we may assume that

$$\sigma \geq (2-\varepsilon)\beta \tag{9}$$

since otherwise, the algorithm returns $\mu \leq \mu_1 = \sigma \leq (2-\varepsilon)\beta$. By Lemma 3.5, we have

$$d(W,Z) \geq (1-\varepsilon)\beta. \tag{10}$$

We also have vertices $z_1, z_2 \in Z$ and $w_1, w_2 \in W$ violating conditions *(ii)* and *(iii)* of Lemma 3.5 respectively. Let us fix such vertices, i.e.,

(a) fix $z_1, z_2 \in Z$ such that $d^{in}(U) \geq (1-\varepsilon)\beta$ for all subsets U of nodes containing z_1, z_2 but not Z, and

(b) fix $w_1, w_2 \in W$ such that $d^{in}(U) \geq (1-\varepsilon)\beta$ for all subsets U of nodes not containing w_1, w_2 but intersecting W.

Also let us fix an arbitrary choice of $x \in X, y \in Y$ (since A and B are uncomparable, we have that X and Y are non-empty and hence such an x and y can be chosen). Henceforth, we will consider the iteration of Step 5 in the algorithm for this choice of x, y, z_1, z_2, w_1, w_2.

We note that (X', Y') form an uncomparable pair. If $\beta(X', Y') \leq (2-\varepsilon)\beta$, then the algorithm returns $\mu \leq \mu_4 \leq (2-\varepsilon)\beta$. Therefore, we may assume that

$$\beta(X', Y') \geq (2-\varepsilon)\beta. \tag{11}$$

Also, we have $d^{in}(X') \leq d^{in}(X \cup Z)$ because X' is the sink-side of a min $\{w_1, w_2, y\} \to \{x, z_1, z_2\}$ cut. Since $d^{in}(X \cup Z) \leq d^{in}(A) \leq \beta$, we have that

$$d^{in}(X') \leq \beta. \tag{12}$$

Similarly,

$$d^{in}(Y') \leq d^{in}(Y \cup Z) \leq \beta. \tag{13}$$

Consequently,

$$\sigma(X', Y') \leq d^{in}(X') + d^{in}(Y') \leq 2\beta. \tag{14}$$

We consider four cases depending on the relations between W and $X' \cup Y'$, and between Z and $X' \cap Y'$.

Case 0. Suppose $W \cap (X' \cup Y') = \emptyset$, $Z \subseteq X' \cap Y'$ (see figure 3.4). In this case $\delta^{in}(X')$ and $\delta^{in}(Y')$ both contain all edges counted in $d(W,Z)$. Hence $\beta(X', Y') \leq \sigma(X', Y') - d(W,Z) \leq (1+\varepsilon)\beta$. The second inequality here is because $\sigma(X', Y') \leq 2\beta$ by (14) and $d(W,Z) \geq (1-\epsilon)\beta$ by 10. This shows that (X', Y') is a $(1+\varepsilon)$-approximation.

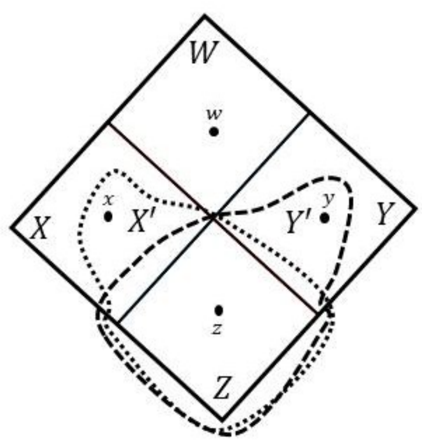

Figure 3.4. The case where $W \cap (X' \cup Y') = \emptyset$, $Z \subseteq X' \cap Y'$.

Let c be the capacity function obtained by increasing the capacity of each edge in E_1 to 2, and let $\bar{c}$ be the capacity function obtained by increasing the capacity of each edge in E_2 to 2. For the remaining three cases, we will use the following proposition.

Proposition 3.6 *If $d^{in}(X' \cap Z') \geq (1-\varepsilon)\beta$ and $d^{in}(Y' \cap Z') \geq (1-\varepsilon)\beta$, then $\beta(X' \cup Z', Y' \cup Z') \leq 2\varepsilon\beta + d_c^{in}(Z)$.*

Proof: If $d^{in}(X' \cap Z') \geq (1-\varepsilon)\beta$, then $d^{in}(X') - d^{in}(X' \cap Z') \leq \varepsilon\beta$. So

$$\begin{aligned} d^{in}(X' \cup Z') &= d^{in}(Z') + d^{in}(X') - d^{in}(X' \cap Z') \\ &\quad - d(X' \setminus Z', Z' \setminus X') - d(Z' \setminus X', X' \setminus Z') \\ &\leq d^{in}(Z') + \varepsilon\beta - d(X' \setminus Z', Z' \setminus X') - d(Z' \setminus X', X' \setminus Z'). \end{aligned} \tag{15}$$

Hence, we have

$$d^{in}(X' \cup Z') \leq d^{in}(Z') + \varepsilon\beta - d(X' \setminus Z', Z' \setminus X'). \tag{16}$$

Similarly,

$$d^{in}(Y' \cup Z') \leq d^{in}(Z') + \varepsilon\beta - d(Y' \setminus Z', Z' \setminus Y'). \tag{17}$$

We need the following proposition.

Proposition 3.7

$$\begin{aligned} \beta(X' \cup Z', Y' \cup Z') \leq \sigma(X' \cup Z', Y' \cup Z') + d_c^{in}(Z') - 2d^{in}(Z') \\ + d(X' \setminus Z', Z' \setminus X') + d(Y' \setminus Z', Z' \setminus Y'). \end{aligned} \tag{18}$$

Proof: By counting the edges entering Z', we have

1. $d_c^{in}(Z') = d^{in}(Z') + |\delta^{in}(Z') \cap E_1|$.
2. $d^{in}(Z') = d(V \setminus (X' \cup Y' \cup Z'), Z') + |\delta^{in}(Z') \cap E_1| + d(X' \setminus Z', Z' \setminus X') + d(Y' \setminus Z', Z' \setminus Y') - d((X' \cap Y') \setminus Z', Z' \setminus (X' \cup Y'))$.

The first equation can be rewritten as

$$d_c^{in}(Z') - 2d^{in}(Z') = -d^{in}(Z') + |\delta^{in}(Z') \cap E_1|.$$

Using this and the second equation, we get

$$\begin{aligned} & d_c^{in}(Z') - 2d^{in}(Z') + d(X' \setminus Z', Z' \setminus X') + d(Y' \setminus Z', Z' \setminus Y') \\ &= -d(V \setminus (X' \cup Y' \cup Z'), Z') + d((X' \cap Y') \setminus Z', Z' \setminus (X' \cup Y')). \end{aligned}$$

Thus the desired inequality (18) simplifies to

$$\begin{aligned} \beta(X' \cup Z', Y' \cup Z') \leq \sigma(X' \cup Z', Y' \cup Z') - d(V \setminus (X' \cup Y' \cup Z'), Z') \\ + d((X' \cap Y') \setminus Z', Z' \setminus (X' \cup Y')). \end{aligned}$$

To prove this inequality, we observe that the edges counted by $d(V \setminus (X' \cup Y' \cup Z'), Z')$ are counted twice in $\sigma(X' \cup Z', Y' \cup Z')$. Hence we have the desired relation (18). □

Using (18), (17) and (16) we get

$$\begin{aligned} \beta(X' \cup Z', Y' \cup Z') &\leq d^{in}(X' \cup Z') + d^{in}(Y' \cup Z') + d_c^{in}(Z') - 2d^{in}(Z') \\ &\quad + d(X' \setminus Z', Z' \setminus X') + d(Y' \setminus Z', Z' \setminus Y') \\ &\leq d^{in}(Z') + \varepsilon\beta + d^{in}(Z') + \varepsilon\beta + d_c^{in}(Z') - 2d^{in}(Z') \\ &= 2\varepsilon\beta + d_c^{in}(Z') \\ &\leq 2\varepsilon\beta + d_c^{in}(Z). \end{aligned}$$

The last inequality above is because Z is a feasible solution for the minimization problem that obtains Z' and hence $d_c^{in}(Z') \leq d_c^{in}(Z)$. This completes the proof of the proposition.

Case 1. Suppose $W \cap (X' \cup Y') = \emptyset$ and $Z \nsubseteq X' \cap Y'$. Without loss of generality, let $Z \nsubseteq X'$. The set $X' \cap Z'$ contains z_1, z_2 but not the whole Z, hence $d^{in}(X' \cap Z') \geq (1-\varepsilon)\beta$ by (a).

We first consider the subcase where $d^{in}(Y' \cap Z') < (1-\varepsilon)\beta$. By the choice of z_1, z_2, this means that $Z \subseteq Y' \cap Z'$. In this case $Y' \cap Z'$ crosses X', because X' does not contain all vertices in Z, and $Y' \cap Z'$ does not contain x. Thus $(X', Y' \cap Z')$ is an uncomparable pair. Now we observe that $\sigma(X', Y' \cap Z') = d^{in}(X') + d^{in}(Y' \cap Z') \leq (2-\varepsilon)\beta$. Thus, $\sigma \leq (2-\varepsilon)\beta$, a contradiction to (9).

Next we consider the other subcase where $d^{in}(Y' \cap Z') \geq (1-\varepsilon)\beta$. Then, by Proposition 3.6, we get

$$\beta(X' \cup Z', Y' \cup Z') \leq 2\varepsilon\beta + d_c^{in}(Z).$$

We are in the case where $(X' \cup Y') \cap W = \emptyset$, so $d_c^{in}(Z) \leq d^{in}(Z) + d(X,Z) + d(Y,Z)$. We now note that $d^{in}(Z) + d(X,Z) + d(Y,Z) = 2d^{in}(Z) - d(W,Z) \leq 2\beta - (1-\varepsilon)\beta = (1+\varepsilon)\beta$ since $d(W,Z) \geq (1-\varepsilon)\beta$ and $d^{in}(Z) \leq \beta$. Hence we have $\beta(X' \cup Z', Y' \cup Z') \leq (1+3\varepsilon)\beta$. Since $(X' \cup Z', Y' \cup Z')$ is an uncomparable pair, we have that $\mu_4 \leq (1+3\varepsilon)\beta$.

Case 2. Suppose $W \cap (X' \cup Y') \neq \emptyset$ and $Z \subseteq X' \cap Y'$. This is similar to Case 1 by symmetry.

Case 3. Suppose $W \cap (X' \cup Y') \neq \emptyset$ and $Z \nsubseteq X' \cap Y'$.

Without loss of generality, suppose $Z \nsubseteq X'$. The set $X' \cap Z'$ contains z_1, z_2 but not the whole Z, hence $d^{in}(X' \cap Z') \geq (1-\varepsilon)\beta$. By the same argument as in the first subcase of Case 1 (first paragraph), we may assume that $d^{in}(Y' \cap Z') \geq (1-\varepsilon)\beta$ (otherwise, $\sigma \leq (2-\varepsilon)\beta$, a contradiction to (9)). The inequality $\beta(X' \cup Z', Y' \cup Z') \leq 2\varepsilon\beta + d_c^{in}(Z)$ holds using Proposition 3.6. If $d_c^{in}(Z) \leq (2-3\varepsilon)\beta$, then these imply $\beta(X' \cup Z', Y' \cup Z') \leq (2-\varepsilon)\beta$. Since $(X' \cup Z', Y' \cup Z')$ is an uncomparable pair, we would thus have $\mu_4 \leq (2-\varepsilon)\beta$. Similarly, if $d_{\bar{c}}^{out}(W) \leq (2-3\varepsilon)\beta$, then we obtain $\mu_4 \leq \beta(X' \setminus W', Y' \setminus W') \leq (2-\varepsilon)\beta$. Thus, we may assume that both

$$d_c^{in}(Z) \geq (2-3\varepsilon)\beta, \text{and} \tag{19}$$

$$d_{\bar{c}}^{out}(W) \geq (2-3\varepsilon)\beta. \tag{20}$$

Let us define the following quantities (see Figure 3.2):

1. $\alpha_1 := d(W \setminus (X' \cup Y'), W \cap (X' \setminus Y'))$,
2. $\alpha_2 := d(W \setminus (X' \cup Y'), W \cap (Y' \setminus X'))$,
3. $\alpha_3 := d(W \cap (X' \setminus Y'), Z \cap (X' \setminus Y'))$,
4. $\alpha_4 := d(W \cap (Y' \setminus X'), Z \cap (Y' \setminus X'))$,
5. $\alpha_5 := d(Z \cap (X' \setminus Y'), X' \cap Y' \cap Z)$, and
6. $\alpha_6 := d(Z \cap (Y' \setminus X'), X' \cap Y' \cap Z)$.

In propositions 3.8, 3.9, 3.10, 3.11, 3.12 and 3.13, we show a sequence of inequalities involving these quantities.

Proposition 3.8

$$(1-\varepsilon)\beta \leq d^{in}(X' \cap Y'), d^{in}(X' \cup Y'), d^{in}(X' \cap Z), d^{in}(X' \cup Z) \leq (1+\varepsilon)\beta.$$

Proof: By submodularity,

$$d^{in}(X' \cap Y') + d^{in}(X' \cup Y') \leq d^{in}(X') + d^{in}(Y') \leq 2\beta.$$

We note that $d^{in}(X' \cap Y') \geq (1-\varepsilon)\beta$ by the choice of z_1, z_2. This shows $d^{in}(X' \cup Y') \leq (1+\varepsilon)\beta$. Similarly, $d^{in}(X' \cup Y') \geq (1-\varepsilon)\beta$ by the choice of w_1, w_2, and hence $d^{in}(X' \cap Y') \leq (1+\varepsilon)\beta$.

By the assumption of Case 3 and $Z \subsetneq X'$, we have that the sets X' and Z are uncomparable. Hence $X' \cap Z$ contains both z_1, z_2 but not all of Z. By the choice of z_1, z_2, we have $d^{in}(X' \cap Z) \geq (1-\varepsilon)\beta$. By submodularity,

$$d^{in}(X' \cup Z) \leq d^{in}(X') + d^{in}(Z) - d^{in}(X' \cap Z) \leq 2\beta - (1-\varepsilon)\beta = (1+\varepsilon)\beta.$$

For the remaining inequalities, we notice that $X' \cup Z$ and Y' are uncomparable, so $\sigma(X' \cup Z, Y') \geq (2-\varepsilon)\beta$ by (9). However, we have

$$\sigma(X' \cup Z, Y') = d^{in}(X' \cup Z) + d^{in}(Y') \leq d^{in}(X' \cup Z) + \beta.$$

Hence, $d^{in}(X' \cup Z) \geq (1-\varepsilon)\beta$. Using submodularity, we obtain $d^{in}(X' \cap Z) \leq (1+\varepsilon)\beta$. ◻

Proposition 3.9 *$(1-6\varepsilon)\beta \le \alpha_3+\alpha_4 \le \beta$.*

Proof: The upper bound follows immediately by definition of α_3, α_4, β and using (8). We show the lower bound. From (19), we recall that $(2-3\varepsilon)\beta \le d_c^{in}(Z) = d^{in}(Z) + |\delta^{in}(Z) \cap E_1|$ and from (20), we recall that $(2-3\varepsilon)\beta \le d_{\bar{c}}^{out}(W) = d^{out}(W) + |\delta^{out}(W) \cap E_2|$. Moreover, we have $d^{in}(Z) \le \beta$ and $d^{out}(W) \le \beta$ by (8). Let C be the set of edges from W to Z, i.e. those counted by $d(W,Z)$. Let $a = |\delta^{in}(Z) \setminus C|$ and $b = |\delta^{out}(W) \setminus C|$. We note that $\alpha_3+\alpha_4 = |C \cap E_1 \cap E_2|$ and $|C|+a+b \le \beta$. We have $|C \cap E_1| \ge |\delta^{in}(Z) \cap E_1| - a$ and $|C \cap E_2| \ge |\delta^{out}(W) \cap E_2| - b$.

From all the above, we get the following sequence of inequalities that show the lower bound:

$$\begin{aligned}
|C \cap E_1 \cap E_2| \ge& |C| - |C \setminus E_1| - |C \setminus E_2| \\
=& |C| - (|C| - |C \cap E_1|) - (|C| - |C \cap E_2|) \\
=& |C \cap E_1| + |C \cap E_2| - |C| \\
\ge& |\delta^{in}(Z) \cap E_1| - a + |\delta^{out}(W) \cap E_2| - b - |C| \\
\ge& (2-3\varepsilon)\beta - d^{in}(Z) + (2-3\varepsilon)\beta - d^{out}(W) - (a+b+|C|) \\
\ge& (4-6\varepsilon)\beta - 3\beta \\
=& (1-6\varepsilon)\beta.
\end{aligned}$$

□

Proposition 3.10 *$(1-8\varepsilon)\beta \le \alpha_1+\alpha_2 \le (1+\varepsilon)\beta$ and $(1-8\varepsilon)\beta \le \alpha_5+\alpha_6 \le (1+\varepsilon)\beta$.*

Proof: We first show the upper bounds. We have $\alpha_1+\alpha_2 \le d^{in}(X' \cup Y')$ which is at most $(1+\varepsilon)\beta$ by Proposition 3.8. Similarly, we have $\alpha_5+\alpha_6 \le d^{in}(X' \cap Y') \le (1+\varepsilon)\beta$. We next show the lower bounds.

We first note that

$$\begin{aligned}
\alpha_5+\alpha_6 \ge d^{in}(X' \cap Y' \cap Z) &- |\delta^{in}(Z) \cap \delta^{in}(X' \cap Y' \cap Z)| \\
&- d(V \setminus (X' \cup Y'), X' \cap Y' \cap Z).
\end{aligned} \tag{21}$$

We bound each of the terms in the RHS now. We observe that $X' \cap Y' \cap Z$ contains z_1, z_2 but not all nodes in Z, hence

$$d^{in}(X' \cap Y' \cap Z) \ge (1-\varepsilon)\beta. \tag{22}$$

Moreover, we have

$$\begin{aligned}
|\delta^{in}(X') \cap \delta^{in}(Y')| &= |\delta^{in}(X')| + |\delta^{in}(Y')| - |\delta^{in}(X') \cup \delta^{in}(Y')| \\
&= \sigma(X', Y') - \beta(X', Y') \\
&\le 2\beta - (2-\varepsilon)\beta \qquad \text{(Using (9) and (11))} \\
&= \varepsilon\beta.
\end{aligned}$$

Here, $|\delta^{in}(X') \cap \delta^{in}(Y')| \le \varepsilon\beta$ implies that we have at most $\varepsilon\beta$ edges entering $X' \cap Y' \cap Z$ from $V \setminus (X' \cup Y')$. Thus, we have

$$d(V \setminus (X' \cup Y'), X' \cap Y' \cap Z) \le \varepsilon\beta. \tag{23}$$

We further have $|\delta^{in}(Z) \cap \delta^{in}(X' \cap Y' \cap Z)| \le \delta^{in}(Z) - \alpha_3 - \alpha_4$. Using Proposition 3.9, we obtain that

$$|\delta^{in}(Z) \cap \delta^{in}(X' \cap Y' \cap Z)| \le \delta^{in}(Z) - \alpha_3 - \alpha_4 \le 6\varepsilon\beta. \tag{24}$$

Substituting the bounds from (22), (23), and (24) in (21), we obtain that $\alpha_5+\alpha_6 \ge (1-\varepsilon)\beta - 6\varepsilon\beta - \varepsilon\beta = (1-8\varepsilon)\beta$. A similar argument shows the lower bound for $\alpha_1+\alpha_2$. □

Proposition 3.11 *$(1-16\varepsilon)\beta \le \alpha_1+\alpha_6 \le \beta$ and $(1-16\varepsilon)\beta \le \alpha_2+\alpha_5 \le \beta$.*

Proof: The upper bounds follow by $\alpha_1+\alpha_6 \le d^{in}(X') \le \beta$ and $\alpha_2+\alpha_5 \le d^{in}(Y') \le \beta$. On the other hand, combining the two inequalities in Proposition 3.10 gives $(2-16\varepsilon)\beta \le \alpha_1+\alpha_2+\alpha_5+\alpha_6$. Now using the upper bound $\alpha_2+\alpha_5 \le \beta$ gives $(1-16\varepsilon)\beta \le \alpha_1+\alpha_6$. Similarly, we obtain $(1-16\varepsilon)\beta \le \alpha_2+\alpha_5$. □

Proposition 3.12 $(1-23\varepsilon)\beta \le \alpha_3+\alpha_6 \le (1+\varepsilon)\beta$.

Proof: Consider the set $M := X' \cap Z$. We note that $\alpha_3+\alpha_6 \le d^{in}(X'\cap Z)$. By Proposition 3.8, we have $d^{in}(M) \le (1+\varepsilon)\beta$, which gives the upper bound. We now show the lower bound.

By Proposition 3.8, we have

$$(1-\varepsilon)\beta \le d^{in}(M). \tag{25}$$

Next we have

$$d^{in}(M) = \alpha_6 + d((Z\setminus X')\cap Y', M\setminus Y') + d(Z\setminus(X'\cup Y'), M) + d(V\setminus Z, M). \tag{26}$$

Also,

$$\alpha_1+\alpha_6+d((Z\setminus X')\cap Y', M\setminus Y') + d(Z\setminus(X'\cup Y'), M) \le d^{in}(X') \le \beta.$$

Using Proposition 3.11, we thus obtain

$$d((Z\setminus X')\cap Y', M\setminus Y') + d(Z\setminus(X'\cup Y'), M) \le 16\varepsilon\beta. \tag{27}$$

We next note that $\alpha_3+\alpha_4+d(V\setminus Z, M) \le d^{in}(Z) \le \beta$. Since $\alpha_3+\alpha_4 \ge (1-6\varepsilon)\beta$ using Proposition 3.9, we obtain

$$d(V\setminus Z, M) \le 6\varepsilon\beta. \tag{28}$$

Using (25), (26), (27), and (28), we obtain

$$\begin{aligned}(1-\varepsilon)\beta &\le d^{in}(M)\\ &= \alpha_6 + d((Z\setminus X')\cap Y', M\setminus Y')\\ &\qquad + d(Z\setminus(X'\cup Y'), M) + d(V\setminus Z, M)\\ &\le \alpha_6 + 16\varepsilon\beta + \alpha_3 + 6\varepsilon\beta\\ &\le \alpha_3+\alpha_6+22\varepsilon\beta.\end{aligned}$$

Rewriting the final inequality gives $(1-23\varepsilon)\beta \le \alpha_3+\alpha_6$. □

Proposition 3.13 $\alpha_1+\alpha_5 \ge 2\alpha_3 - 51\varepsilon\beta$.

Proof: The above propositions give us a chain of relations:

$$\begin{aligned}(1-16\varepsilon)\beta-\alpha_6 &\le \alpha_1 \le \beta-\alpha_6,\\ (1-8\varepsilon)\beta-\alpha_1 &\le \alpha_2 \le (1+\varepsilon)\beta-\alpha_1,\\ (1-16\varepsilon)\beta-\alpha_2 &\le \alpha_5 \le \beta-\alpha_2,\\ (1-23\varepsilon)\beta-\alpha_3 &\le \alpha_6 \le (1+\varepsilon)\beta-\alpha_3.\end{aligned}$$

By substitution, we get

$$\alpha_3-17\varepsilon\beta \le \alpha_1 \le \alpha_3+23\varepsilon\beta, \tag{29}$$

$$\alpha_1-17\varepsilon\beta \le \alpha_5 \le \alpha_1+8\varepsilon\beta. \tag{30}$$

By substituting again, we get

$$\alpha_3-34\varepsilon\beta \le \alpha_5 \le \alpha_3+31\varepsilon\beta. \tag{31}$$

Using (29) and (31), we obtain $\alpha_1+\alpha_5 \ge 2\alpha_3-51\varepsilon\beta$. □

Without loss of generality, let $\alpha_3 \ge (\alpha_3+\alpha_4)/2$, since if not, there is another iteration of the algorithm where x and y are switched. Therefore, by Proposition 3.9, we have

$$\alpha_3 \ge (1/2-3\varepsilon)\beta. \tag{32}$$

Let H be the directed graph obtained in Step 5(ix) of the algorithm, i.e., by contracting $X'\cap Y'$ to a node z', contracting $V\setminus X'$ to a node w', and removing all $w'z'$ arcs. Let

$$A_0 := (X' \cap Z) \cup \{z'\} \text{ and}$$
$$B_0 := (X' \setminus W) \cup \{z'\}.$$

We note that (A_0, B_0) is a feasible solution for Step 5(x) of the algorithm. The following proposition shows an upper bound on the value of $\beta(A_0, B_0)$ in H:

Proposition 3.14

$$|\delta_H^{in}(A_0) \cup \delta_H^{in}(B_0)| \leq \alpha_3 + 39\varepsilon\beta. \tag{33}$$

Proof: We have that

$$\begin{aligned} |\delta_H^{in}(A_0)| &= |\delta_H(V \setminus A_0, (X' \cap Z) \setminus Y')| + |\delta_H(X' \setminus (Y' \cup Z), z')| \\ &= d(V \setminus A_0, (X' \cap Z) \setminus Y') + d(X' \setminus (Y' \cup Z), X' \cap Y'), \end{aligned} \tag{34}$$

and

$$\begin{aligned} |\delta_H^{in}(B_0) \setminus \delta_H^{in}(A_0)| = d(V \setminus X', X' \setminus (Y' \cup W \cup Z)) \\ + d((X' \cap W) \setminus Y', X' \setminus (Y' \cup W \cup Z)). \end{aligned} \tag{35}$$

We would like to bound the sum of the above four terms. The term $d(V \setminus A_0, (X' \cap Z) \setminus Y')$ counts a subset of the edges entering Z. Since we have $d(W, Z) \geq (1-\varepsilon)\beta$, while $d^{in}(Z) \leq \beta$, it follows that all but $\varepsilon\beta$ edges entering Z are from W. Hence,

$$d(V \setminus A_0, (X' \cap Z) \setminus Y') \leq d((V \setminus A_0) \cap W, (X' \cap Z) \setminus Y') + \varepsilon\beta. \tag{36}$$

Next, we observe that

$$d((V \setminus A_0) \cap W, (X' \cap Z) \setminus Y') = \alpha_3 + d(W \setminus X', (X' \cap Z) \setminus Y'). \tag{37}$$

Using (34), (35), (36), and (37), we obtain that $|\delta_H^{in}(A_0) \cup \delta_H^{in}(B_0)| - \alpha_3 - \varepsilon\beta$ is at most

$$\begin{aligned} &d(W \setminus X', (X' \cap Z) \setminus Y') + d(X' \setminus (Y' \cup Z), X' \cap Y') \\ &\quad + d(V \setminus X', X' \setminus (Y' \cup W \cup Z)) + d((X' \cap W) \setminus Y', X' \setminus (Y' \cup W \cup Z)). \end{aligned}$$

We now bound the terms in the above sum to show that the total is at most $38\varepsilon\beta$.

1. In order to bound the sum of the first and the fourth terms, we observe that

   $$\begin{aligned} &d(W \setminus X', (X' \cap Z) \setminus Y') + d((X' \cap W) \setminus Y', X' \setminus (Y' \cup W \cup Z)) \\ &\qquad + \alpha_3 + \alpha_4 \leq d^{out}(W) \leq \beta. \end{aligned}$$

   Using $\alpha_3 + \alpha_4 \geq (1-6\varepsilon)\beta$ from Proposition 3.9, we obtain

   $$d(W \setminus X', (X' \cap Z) \setminus Y') + d((X' \cap W) \setminus Y', X' \setminus (Y' \cup W \cup Z)) \leq 6\varepsilon\beta.$$

2. The second term $d(X' \setminus (Y' \cup Z), X' \cap Y')$ counts a subset of the edges entering Y'. We have

   $$d(X' \setminus (Y' \cup Z), X' \cap Y') + \alpha_2 + \alpha_5 \leq d^{in}(Y') \leq \beta.$$

   Using $\alpha_2 + \alpha_5 \geq (1-16\varepsilon)\beta$ from Proposition 3.11, we obtain

   $$d(X' \setminus (Y' \cup Z), X' \cap Y') \leq 16\varepsilon\beta.$$

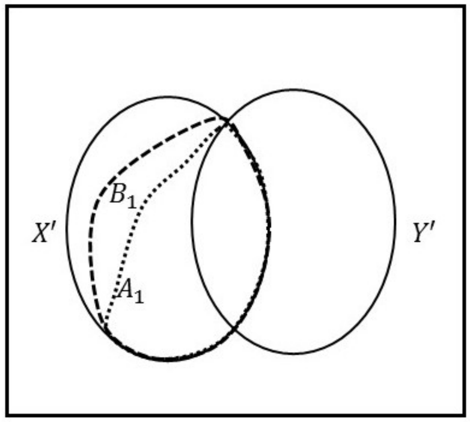

Figure 3.5. The sets A_1 and B_1 are completely contained in X'.

3. The third term $d(V \setminus X', X' \setminus (Y' \cup W \cup Z))$ counts a subset of the edges entering X'. We have

$$d(V \setminus X', X' \setminus (Y' \cup W \cup Z)) + \alpha_1 + \alpha_6 \leq d^{in}(X') \leq \beta.$$

Using $\alpha_1 + \alpha_6 \geq (1 - 16\varepsilon)\beta$ from Proposition 3.11, we obtain

$$d(V \setminus X', X' \setminus (Y' \cup W \cup Z)) \leq 16\varepsilon\beta.$$

Thus, the total contribution is at most $38\varepsilon\beta$. □

Using Proposition 3.14, Step 5(x) of the algorithm finds $\overline{w'}z'$-sets $A' \subsetneq B'$ such that

$$\begin{aligned} |\delta_H^{in}(A') \cup \delta_H^{in}(B')| &\leq \frac{3}{2}|\delta_H^{in}(A_0) \cup \delta_H^{in}(B_0)| \\ &\leq \frac{3}{2}(\alpha_3 + 39\varepsilon\beta) = \frac{3}{2}\alpha_3 + \frac{117}{2}\varepsilon\beta. \end{aligned} \tag{38}$$

Let $A_1 := (A' \setminus \{z'\}) \cup (X' \cap Y')$ and $B_1 := (B' \setminus \{z'\}) \cup (X' \cap Y')$, i.e., A_1 and B_1 are the corresponding sets in V obtained by replacing z' by $X' \cap Y'$ (see Figure 3.5). Now we consider the pair $(X' \cap B_1, Y' \cup A_1)$ and observe that it is an uncomparable pair. We next compute the bicut value $\beta(X' \cap B_1, Y' \cup A_1)$ of this pair in the original directed graph. The next proposition will help in bounding the bicut value.

Proposition 3.15

$$\beta(X' \cap B_1, Y' \cup A_1) + \alpha_5 + \alpha_1 \leq \sigma(X', Y') + |\delta_H^{in}(A') \cup \delta_H^{in}(B')|.$$

Proof: The proposition follows by counting the edges on the left hand side. We use a figure to easily visualize the counting argument. We recall that $X' \cap Y' \subseteq A_1 \subseteq B_1 \subseteq X'$.

We use Figure 3.6. Each arrow represents that all edges from the set of nodes in the rectangle containing its tail to the set of nodes in the rectangle containing its head are counted in the left hand side of Proposition 3.15. In particular, edges corresponding to $\delta^{in}(X' \cap B_1)$ are marked as thin continuous arrows and $\delta^{in}(Y' \cup A_1) \setminus \delta^{in}(X' \cap B_1)$ are marked as thin dotted arrows. Edges corresponding to $\delta(W \setminus (X' \cup Y'), W \cap (X' \cap Y')$ are marked as thick $\rightarrow W$ arrows to indicate that the head v of the edges are in $W \cap S$ where S is the set of nodes in the rectangle containing the head. Edges corresponding to $\delta(Z \cap (X' \setminus Y'), X' \cap Y' \cap Z)$ are marked as thick dotted $Z \rightarrow Z$ arrows to indicate that the tail u and the head v of the edges are in $Z \cap S_1$ and $Z \cap S_2$ respectively where S_1 and S_2 are the set of nodes in the rectangles containing the tail and head respectively.

In order to argue that every edge in the LHS is also counted in the RHS, we mark the tail of the arrows as follows: $\square$ indicates that the edge is counted in $\delta^{in}(X')$, $\diamond$ indicates that the edge is counted in $\delta^{in}(Y')$ and $\circ$ indicates that the edge is counted in $\delta_H^{in}(A') \cup \delta_H^{in}(B')$.

□

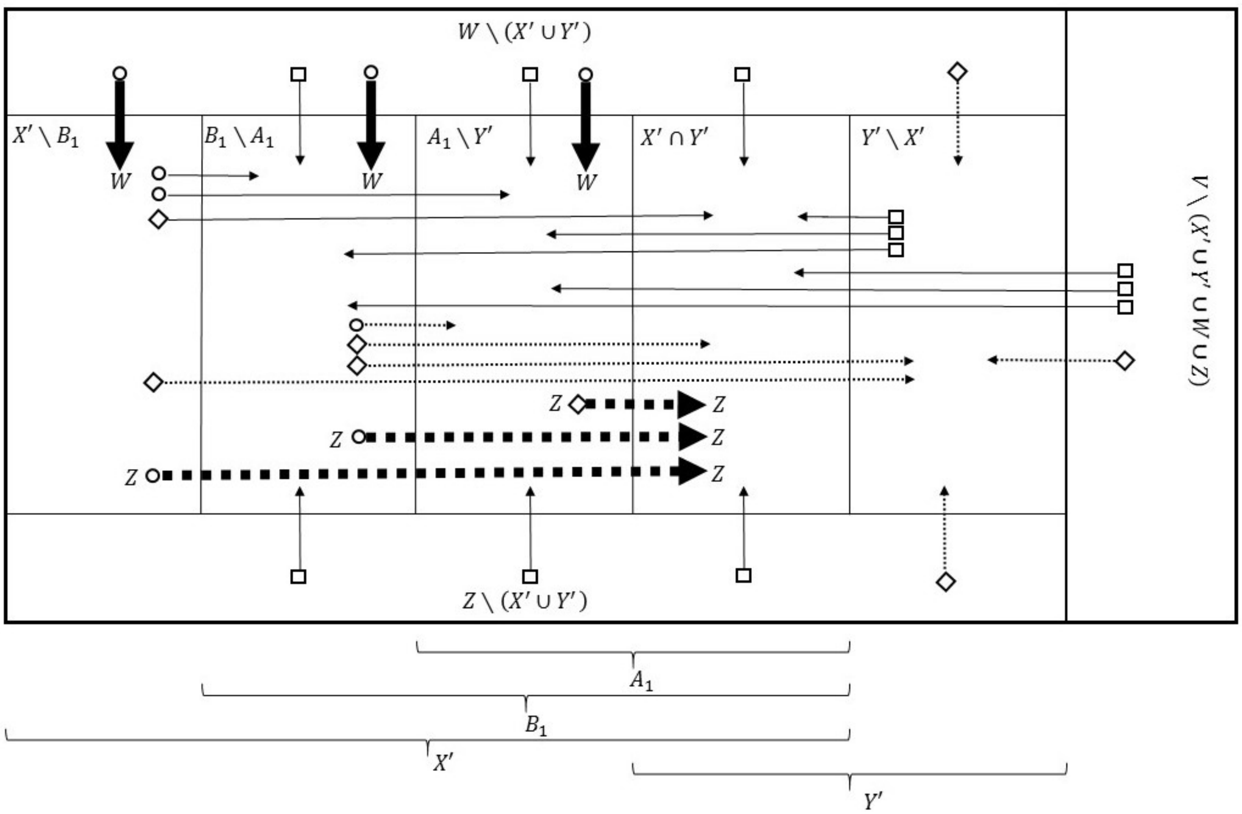

Figure 3.6. Proof of Proposition 3.15.

Using Proposition 3.15 and inequality (38), we get

$$\begin{aligned}\beta(X' \cap B_1, Y' \cup A_1) &\leq \sigma(X', Y') + |\delta_H^{in}(A') \cup \delta_H^{in}(B')| - \alpha_5 - \alpha_1 \\ &\leq 2\beta + \frac{3}{2}\alpha_3 + \frac{117}{2}\varepsilon\beta - \alpha_5 - \alpha_1.\end{aligned}$$

Next, using Proposition 3.13, we get

$$\beta(X' \cap B_1, Y' \cup A_1) \leq 2\beta + \frac{3}{2}\alpha_3 + \frac{117}{2}\varepsilon\beta - (2\alpha_3 - 51\varepsilon\beta) = 2\beta - \frac{1}{2}\alpha_3 + \frac{219}{2}\varepsilon\beta.$$

Finally, we recall that $\alpha_3 \geq (1/2 - 3\varepsilon)\beta$ from (32) and hence,

$$\beta(X' \cap B_1, Y' \cup A_1) \leq \left(2 + \frac{219}{2}\varepsilon\right)\beta - \frac{1}{2}\left(\frac{1}{2} - 3\varepsilon\right)\beta = \left(\frac{7}{4} + 111\varepsilon\right)\beta.$$

Based on all the cases analyzed above, the approximation factor is at most

$$\max\left\{1 + \varepsilon, 1 + 3\varepsilon, 2 - \varepsilon, \frac{7}{4} + 111\varepsilon\right\} = \max\left\{2 - \varepsilon, \frac{7}{4} + 111\varepsilon\right\}.$$

In order to minimize the factor, we set $\varepsilon = 1/448$ to get the desired approximation factor.

4 Conclusion and Open Problems

In this work, we considered BiCut which is a natural extension of the global minimum cut problem from undirected graphs to directed graphs. While its fixed-terminal variant is well-understood both in terms of complexity and approximability, BiCut has hardly been investigated in the literature. In this work, we gave a $(2 - 1/448)$-approximation for BiCut thus exhibiting a dichotomous behaviour in the approximability between the global and the fixed-terminal variants. Intriguingly, the complexity of BiCut remains elusive and is an open problem that merits thorough investigation.

Our approximation algorithm for BiCut needs to solve $(s, *, t)$-Lin-3-Cut as an intermediate subproblem. In this work, we gave a 3/2-approximation for $(s, *, t)$-Lin-3-Cut and use this factor in the analysis of the approximation factor for BiCut. If $(s, *, t)$-Lin-3-Cut is solvable efficiently or if its approximability is better than 3/2, then the approximability of BiCut would also improve using our techniques. Hence, it would be interesting to resolve the complexity of $(s, *, t)$-Lin-3-Cut.

References

[1] H. Angelidakis, Y. Makarychev, and P. Manurangsi. An Improved Integrality Gap for the Călinescu-Karloff-Rabani Relaxation for Multiway Cut. In *Proceedings of the 19th International Conference on Integer Programming and Combinatorial Optimization*, IPCO '17, pages 40–50, 2017.

[2] A. Bernáth and G. Pap. Blocking optimal arborescences. *Mathematical Programming*, 161(1):583–601, Jan 2017.

[3] C. Chekuri and V. Madan. Approximating multicut and the demand graph. In *Proceedings of the 28th Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '17, pages 855–874, 2017.

[4] K. Cheung, W. Cunningham, and L. Tang. Optimal 3-terminal cuts and linear programming. *Mathematical Programming*, 106(1):1–23, 2006.

[5] G. Călinescu, H. Karloff, and Y. Rabani. An improved approximation algorithm for multiway cut. *Journal of Computer and System Sciences*, 60(3):564–574, 2000.

[6] E. Dahlhaus, D. Johnson, C. Papadimitriou, P. Seymour, and M. Yannakakis. The complexity of multiterminal cuts. *SIAM Journal on Computing*, 23(4):864–894, 1994.

[7] R. Erbacher, T. Jaeger, N. Talele, and J. Teutsch. Directed multicut with linearly ordered terminals. Preprint: https://arxiv.org/abs/1407.7498, 2014.

[8] N. Garg, V. Vazirani, and M. Yannakakis. Multiway cuts in directed and node weighted graphs. In *Proceedings of the 20th International Colloquium on Automata, Languages and Programming*, ICALP '94, pages 487–498, 1994.

[9] O. Goldschmidt and D. Hochbaum. A polynomial algorithm for the k-cut problem for fixed k. *Math. Oper. Res.*, 19(1):24–37, Feb 1994.

[10] D. Karger, P. Klein, C. Stein, M. Thorup, and N. Young. Rounding algorithms for a geometric embedding of minimum multiway cut. *Mathematics of Operations Research*, 29(3):436–461, 2004.

[11] D. Karger and R. Motwani. Derandomization through approximation. In *Proceedings of the 26th annual ACM symposium on Theory of computing*, STOC '94, pages 497–506, 1994.

[12] D. Karger and C. Stein. A new approach to the minimum cut problem. *Journal of ACM*, 43(4):601–640, July 1996.

[13] S. Khot. On the power of unique 2-prover 1-round games. In *Proceedings of the 34th annual ACM Symposium on Theory of Computing*, STOC '02, pages 767–775, 2002.

[14] E. Lee. Improved Hardness for Cut, Interdiction, and Firefighter Problems. In *Proceedings of the 44th International Colloquium on Automata, Languages, and Programming*, ICALP, pages 92:1–92:14, 2017.

[15] R. Manokaran, J. Naor, P. Raghavendra, and R. Schwartz. SDP Gaps and UGC Hardness for Multiway Cut, 0-extension, and Metric Labeling. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing*, STOC '08, pages 11–20, 2008.

[16] M. Queyranne. On Optimum *k*-way Partitions with Submodular Costs and Minimum Part-Size Constraints. Talk Slides, URL: https://smartech.gatech.edu/bitstream/handle/1853/43309/Queyranne.pdf, 2012.

[17] H. Saran and V. Vazirani. Finding k Cuts within Twice the Optimal. *SIAM Journal on Computing*, 24(1):101–108, 1995.

[18] A. Sharma and J. Vondrák. Multiway cut, pairwise realizable distributions, and descending thresholds. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing*, STOC '14, pages 724–733, 2014.

[19] M. Thorup. Minimum *k*-way Cuts via Deterministic Greedy Tree Packing. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing*, STOC '08, pages 159–166, 2008.

[20] V. Vazirani and M. Yannakakis. Suboptimal cuts: Their enumeration, weight and number (extended abstract). In *Proceedings of the 19th International Colloquium on Automata, Languages and Programming*, ICALP '92, pages 366–377, 1992.