9/7-Approximation for Two-Edge-Connectivity and Two-Vertex-Connectivity

Ali Çivril*

September 26, 2024

Abstract

We provide algorithms for the minimum 2-edge-connected spanning subgraph problem and the minimum 2-vertex-connected spanning subgraph problem with approximation ratio $\frac{9}{7}$. This improves upon a recent algorithm with ratio slightly smaller than $\frac{4}{3}$ for 2-edge-connectivity, and another one with ratio $\frac{4}{3}$ for 2-vertex-connectivity.

1 Introduction

In the minimum 2-edge-connected spanning subgraph problem (2-ECSS), we are given an undirected simple graph $G = (V, E)$, and the goal is to compute a 2-edge-connected spanning subgraph (2-ECSS) of G with minimum number of edges. The problem is NP-hard and APX-hard even for subcubic graphs [4]. One of the first results breaking the barrier of approximation factor 2 for the problem is due to Khuller and Vishkin [12], which is a $\frac{3}{2}$-approximation algorithm. Cheriyan, Sebö and Szigeti [2] improved the factor to $\frac{17}{12}$. The problem is notoriously difficult to approximate, and there have been some incorrect and incomplete claims since this result. Vempala and Vetta [16], and Jothi, Raghavachari and Varadarajan [11] claimed to have $\frac{4}{3}$ and $\frac{5}{4}$ approximations, respectively. Krysta and Kumar [13] went on to give a $(\frac{4}{3} - \epsilon)$-approximation for some small $\epsilon > 0$ assuming the result of Vempala and Vetta [16]. Sebö and Vygen [15] provides a $\frac{4}{3}$-approximation algorithm by using ear decompositions, and mentions that the aforementioned claimed approximation ratio $\frac{5}{4}$ has not appeared with a complete proof in a fully refereed publication. The result of Vempala and Vetta [16] recently re-appeared in [10] with a correction. Given this, it is not clear if the ratio $(\frac{4}{3} - \epsilon)$ by Krysta and Kumar [13] still holds. Very recently, Garg, Grandoni, and Ameli [6] slightly improved upon the ratio $\frac{4}{3}$ by proving $\frac{118}{89} + \epsilon$ approximation, which stands as the best current result for approximating the problem.

In 2-VCSS, given an undirected simple graph $G = (V, E)$, one finds a 2-vertex-connected (or simply 2-connected) spanning subgraph (2-VCSS) of G with minimum number of edges. This problem is NP-hard via a reduction from the Hamiltonian cycle problem. Furthermore, by the result of Czumaj and Lingas [5], it does not admit a PTAS unless P = NP. The first result improving the factor 2 for 2-VCSS came from Khuller and Vishkin [12], which is a $\frac{5}{3}$-approximation algorithm. Garg, Vempala, and Singla [7] improved the approximation ratio to $\frac{3}{2}$. Cheriyan and Thurimella [3] also provides the same approximation ratio in a more general context, where they consider k-connectivity. Vempala and Vetta [16] claimed the ratio $\frac{4}{3}$, which is shown to be not valid by Heeger and Vygen [9]. Jothi, Raghavachari and Varadarajan [11] claimed the ratio $\frac{5}{4}$.

*Istanbul Atlas University, Computer Engineering Department, Kagithane, 34408 Istanbul, Turkey, e-mail: ali.civril@atlas.edu.tr

However, this claim has been later withdrawn (see [8]). Gubbala and Raghavachari [8] claimed to have a $\frac{9}{7}$-approximation algorithm. The only complete (and exceedingly long) proof of this claim is in Gubbala's thesis [14], which has not appeared anywhere else. To the best of our knowledge, the algorithm by Heeger and Vygen [9] with factor $\frac{10}{7}$ stands as the first refereed improvement over the factor $\frac{3}{2}$ after a long hiatus. This paper also contains a somewhat more detailed discussion about the aforementioned claimed results, implying that theirs is the first improvement. Very recently, Bosch-Calvo, Grandoni, and Ameli provided an algorithm with approximation ratio $\frac{4}{3}$ [1], the current best factor.

The purpose of this paper is to prove the following theorem via a common algorithm for both of the problems we consider:

Theorem 1. *There exists a polynomial-time $\frac{9}{7}$-approximation algorithm for the 2-edge-connected spanning subgraph problem, and a polynomial-time $\frac{9}{7}$-approximation algorithm for the 2-vertex-connected spanning subgraph problem.*

2 Preliminaries

We will use the lower bound derived from the dual of the natural LP relaxation for 2-ECSS. Here $\delta(S)$ denotes the set of edges with one end in the cut S and the other not in S.

$$\begin{aligned} \text{minimize} \quad & \sum_{e\in E} x_e & & \text{(EC)} \\ \text{subject to} \quad & \sum_{e\in\delta(S)} x_e \geq 2, & \forall\, \emptyset \subset S \subset V, & \\ & 1 \geq x_e \geq 0, & \forall\, e \in E. & \end{aligned}$$

The following is the dual of (EC).

$$\begin{aligned} \text{maximize} \quad & \sum_{\emptyset\subset S\subset V} 2y_S - \sum_{e\in E} z_e & & \text{(EC-D)} \\ \text{subject to} \quad & \sum_{S:e\in\delta(S)} y_S \leq 1 + z_e, & \forall e \in E, & \\ & y_S \geq 0, & \forall\, \emptyset \subset S \subset V, & \\ & z_e \geq 0, & \forall\, e \in E. & \end{aligned}$$

We assume that the input graph G is 2-connected. Otherwise, the algorithm of the next section can be executed on blocks (maximal 2-connected subgraphs) of G separately, and establishing the approximation ratio within a block suffices. This is due to the fact that the value of an optimal solution for 2-ECSS is the sum of those of blocks. Given a vertex $v \in V$ and a 2-ECSS or a 2-VCSS F, if the degree of v in the graph (V, F) is at least 3, it is called a *high-degree vertex* on F. For a path $P = v_1 v_2 \ldots v_{k-1} v_k$, v_1 and v_k are the *end vertices* of P, and all the other vertices are the *internal vertices* of P. In particular, v_2 and v_{k-1} are the *side vertices* of P. By definition, an internal vertex of a 2-segment or a 3-segment is also a side vertex. A path whose internal vertices are all degree-2 vertices on F is called a *plain path* on F. A maximal plain path is called a *segment*. The length of a segment is the number of edges on the segment. If the length of a segment is ℓ, it is called an *ℓ-segment*. A 1-segment is also called a *trivial segment*. An ℓ-segment with $\ell \geq 2$ is called a *short segment* if $\ell \leq 4$, otherwise a *long segment*. If a 2-ECSS remains feasible upon

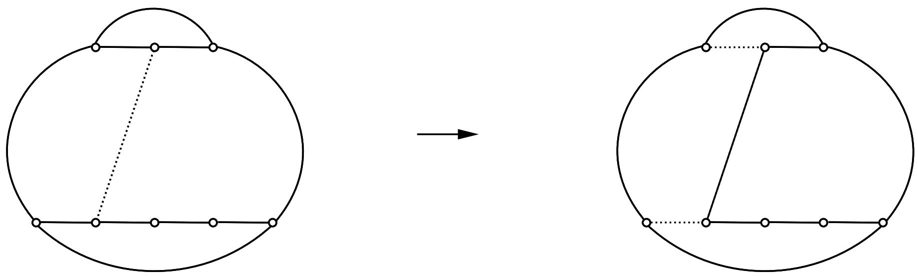


Figure 1: An example of an improvement operation on a strong 2-segment

removal of an edge, the edge is called *redundant*. If the removal of a segment from a 2-VCSS F violates 2-connectivity, it is called a *weak segment* on F, otherwise a *strong segment* on F.

3 The Algorithm for 2-ECSS and 2-VCSS

The first step of the algorithm is to compute an inclusion-wise minimal 2-VCSS on G (Recall our assumption from the previous section). This can be computed by taking all the edges in E, and then deleting an element of this set one by one as long as the feasibility is not violated. Let F be such a solution. *The second step* of the algorithm modifies the running solution F via *improvement processes* to eliminate specific sets of edges from $E \setminus F$. Given a strong short segment S on F and a side vertex u of S, let $N(u) \subseteq E \setminus F$ denote the set of edges incident to u, which are not in F. In each iteration of a loop, the algorithm checks for a selected S and u if including k edges from $N(u)$, which we call a *critical edge set*, and excluding $k+1$ edges from F maintains feasibility, where $k \in \{1, 2\}$. If so, it switches to this cheaper feasible solution, which we call an *improvement operation*. Note that this improves the cost of the solution by 1, and the critical edge sets can be examined in polynomial time, as their sizes are constant. We list all 4 types of improvement operations and the corresponding critical edge sets in Figure 1-Figure 4.

If there are no $k+1$ edges in F whose removal maintains feasibility, the algorithm resorts to

---

Algorithm 1: 2-ECSS-2-VCSS($G(V, E)$)

// First step: Initialization
Let F be an inclusion-wise minimal 2-VCSS of G

// Second step: Improvement processes
while *there is a strong short segment S on F and a side vertex u of S on which an improvement process has not been called* **do**
  | Improvement-Process(F, S, u)

// Third step: Final improvements
for *all $e \in E \setminus F$* **do**
  | If including e and excluding at least two edges from F maintains feasibility, switch to that cheaper solution
  | Update F and $E \setminus F$

// Fourth step: Clean-up
Delete redundant edges from F to obtain $\overline{F}$

return $(F, \overline{F})$

Algorithm 2: IMPROVEMENT-PROCESS(F, S, u)

```
if there is an improvement operation that can be performed on u then
    Apply the improvement operation on F
    return
for each set of critical edges H on u do
    Let 𝒮 be the set of strong short segments on F ∪ H that do not exist on F
    for each strong short segment T in 𝒮 and a side vertex v of T do
        if no improvement process has been called on (T, v) then
            IMPROVEMENT-PROCESS(F ∪ H, T, v)
            if there is an improvement operation performed in
             IMPROVEMENT-PROCESS(F ∪ H, T, v) then
                Perform deletion operation on F ∪ H in the order F, H
                return
if there is no improvement operation performed in any of the recursive calls above then
    Restore F to the original set considered before the function call
```

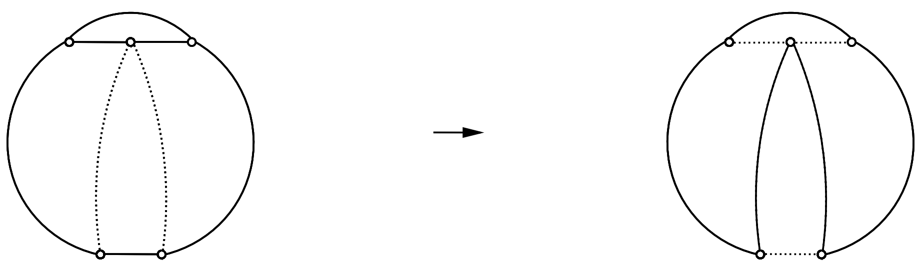


Figure 2: An example of an improvement operation on a strong 2-segment

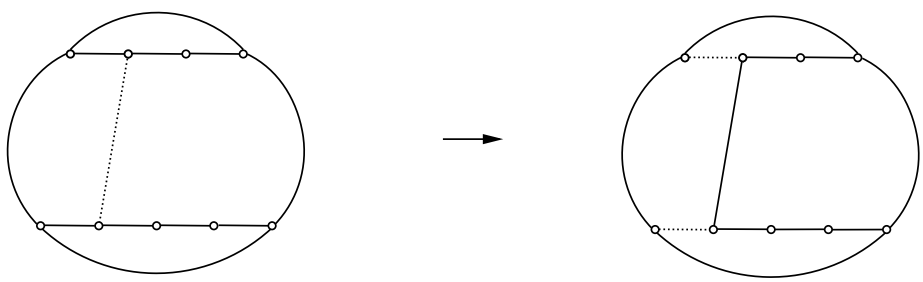


Figure 3: An example of an improvement operation on a strong 3-segment

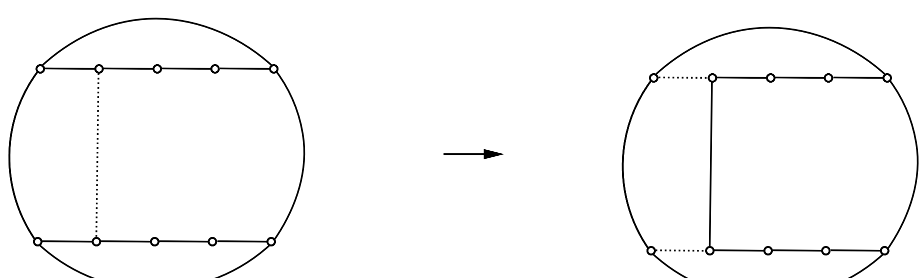


Figure 4: An example of an improvement operation on a strong 4-segment

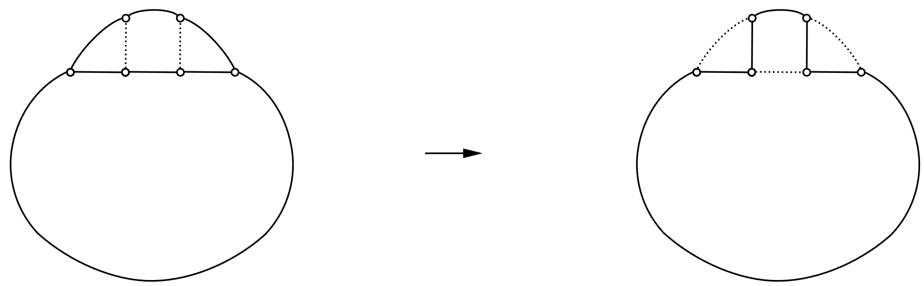


Figure 5: An example of an improvement process of recursion depth 2

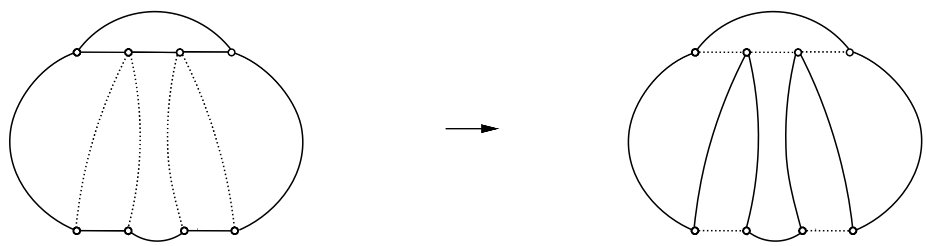


Figure 6: An example of an improvement process of recursion depth 2

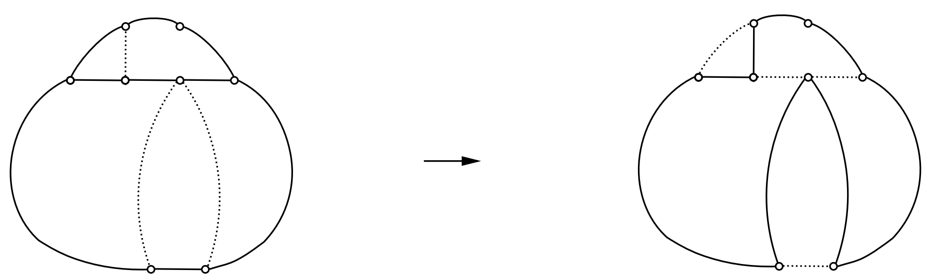


Figure 7: An example of an improvement process of recursion depth 2

recursion. In particular, given a set H of edges in $N(u)$ to include, the algorithm checks if $F \cup H$ contains new strong short segments that do not exist on F. If it does, it calls the improvement process procedure given in Algorithm 2 for S and u *recursively* on the side vertices of the newly appearing strong short segments provided that no improvement process has been previously called on a given segment and a side vertex. These calls are performed for all H on u and for each side vertex $u \in S$. If there is an improvement operation in one of the recursive calls, the called function returns and the caller performs a specific *deletion operation* as follows. It attempts to delete the edges from $F \cup H$ in the order F, H, where the order within the sets F and H are immaterial. Specifically, it deletes an edge provided that the residual graph remains feasible. This enforces to keep the edges in H in the solution. Examples of these operations are given in Figure 5-Figure 13, where the depth of the recursion tree is either 2 or 3. After the reverse-delete operation, the current function call returns. If after all the recursive calls from u there is no improvement operation performed, the solution F is restored to the original one before the function call on u. The main iterations continue until there is no S and u on which we can perform an improvement process.

The algorithm finishes as follows.

Third step: Iterate over all the edges $e \in E \setminus F$, and check if including e and excluding at least two edges from F maintains a 2-VCSS. If so, switch to that solution. Update F and $E \setminus F$. This step

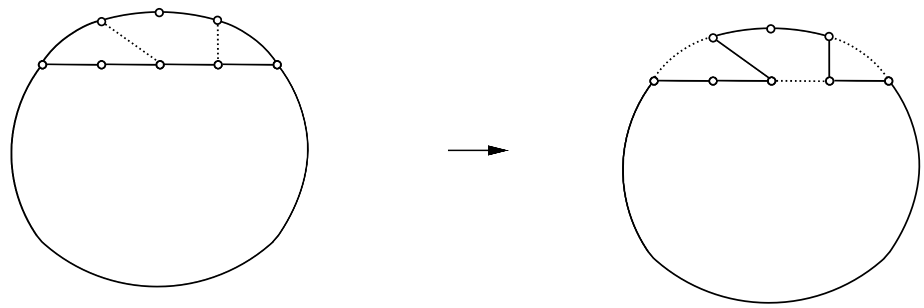


Figure 8: An example of an improvement process of recursion depth 2

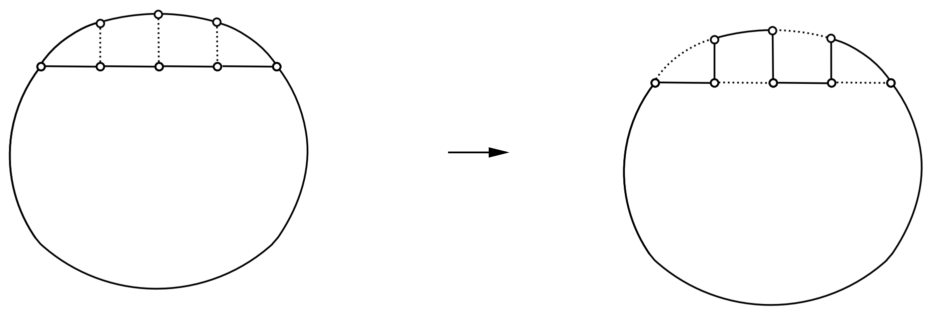


Figure 9: An example of an improvement process of recursion depth 3

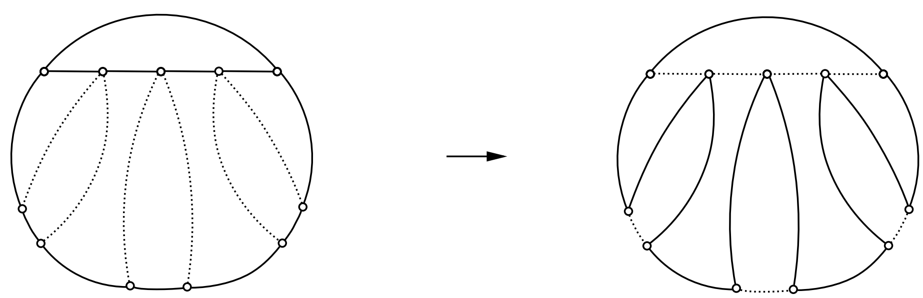


Figure 10: An example of an improvement process of recursion depth 3

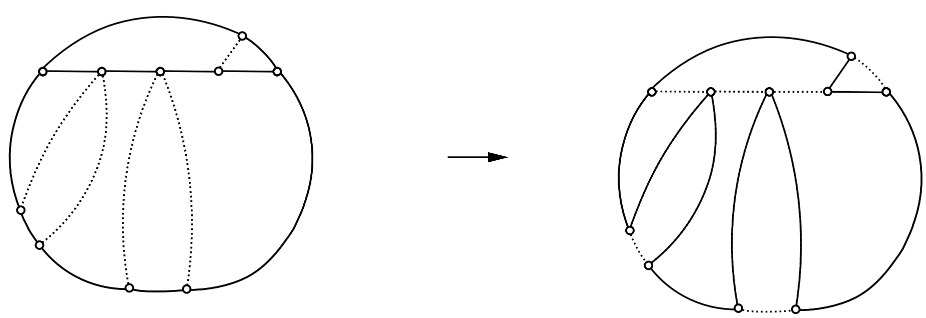


Figure 11: An example of an improvement process of recursion depth 3

might be considered as a final check for improvements, this time not specific to the configurations considered in improvement operations of the second step, but a general one.

Fourth step: Remove all the redundant edges, i.e., the edges whose removal does not violate 2-edge-connectivity. The result of this operation is denoted by $\overline{F}$, distinguished from F, which is

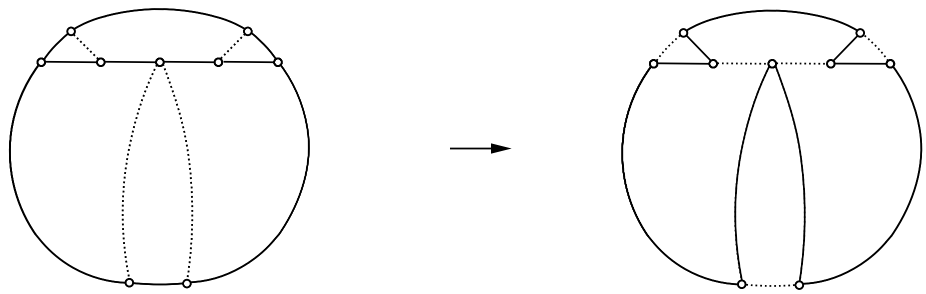


Figure 12: An example of an improvement process of recursion depth 3

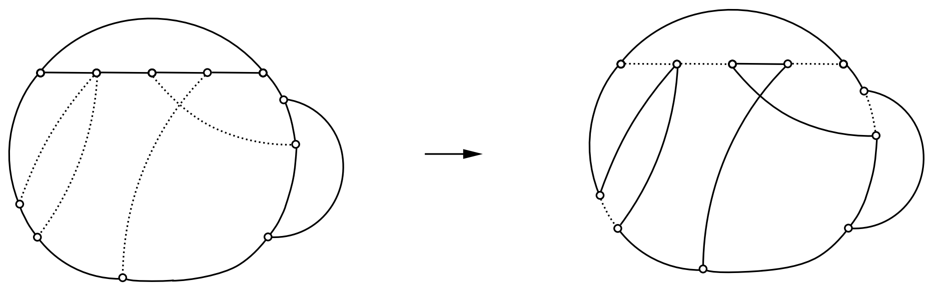


Figure 13: An example of an improvement process of recursion depth 3

the result of the previous step.

We do not discuss an efficient implementation of the algorithm.

Proposition 2. *Algorithm 1 can be implemented in polynomial time.*

Proof. It is clear that the first step and the fourth step can be implemented in polynomial time. The third step terminates in $O(|E|)$ iterations, since each iteration either improves the cost of the solution or skips an edge. As for the second step, it suffices to argue that the main loop of Improvement-Process terminates in polynomial number of operations. As noted, there are polynomially many sets H, since $|H|$ is constant. Starting from a side vertex u of a strong short segment S, consider the recursion tree in which each node represents a recursive function call. By definition, each node of this tree is associated to a side vertex of a strong short segment. A vertex can be a side vertex of constant number of strong segments. This implies that the number of nodes in the tree is polynomially bounded. So the main loop of Improvement-Process terminates in polynomial number of operations. □

4 Proof of Theorem 1

Let $(F, \overline{F})$ be a solution returned by Algorithm 1, and $opt(G)$ denote the value of an optimal 2-ECSS on G.

Lemma 3. *We can modify $(F, \overline{F})$ such that the following hold:*

1. *$|F|$ and $|\overline{F}|$ does not change.*

2. *Let $e_1 = (u_1, v_1)$, $e_2 = (u_2, v_2)$, and $f_1 = (u_1, w_1)$ be the edges of three distinct trivial segments on F. Let S be a cut such that $\delta(S) \cap F = \{e_1, e_2\}$. Then there is no edge $g \in E \setminus F$ such that $g \in \delta(S)$.*

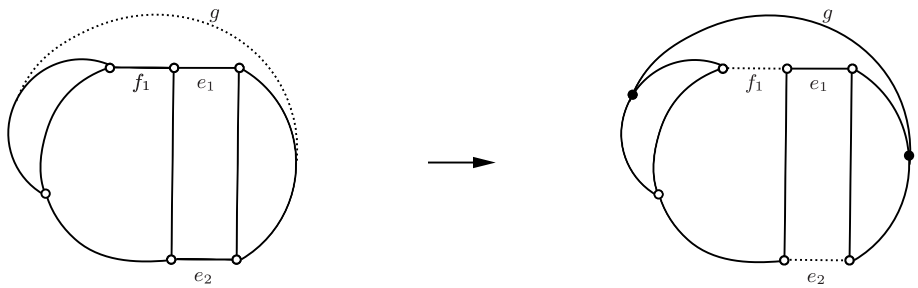


Figure 14: Contradiction to the existence of $g \in E \setminus F$ such that $g \in \delta(S)$ in the proof of Lemma 3

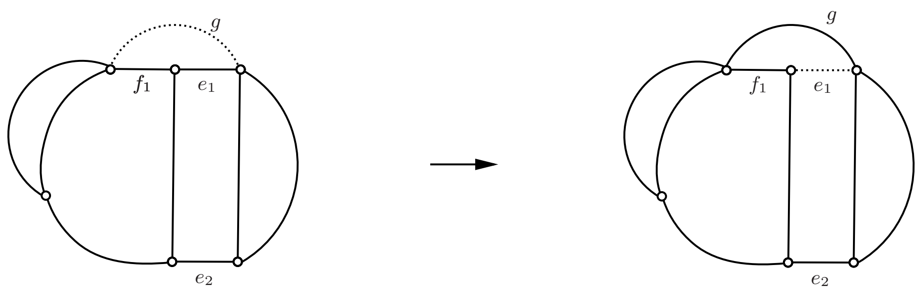


Figure 15: Modification of F in the proof of Lemma 3

Proof. If $F \cup \{g\} \setminus \{e_2, f_1\}$ is a 2-VCSS, then the existence of an edge $g \in \delta(S)$ contradicts the third step of the algorithm. In this case the algorithm includes g into the solution by excluding e_2 and f_1, which is feasible (See Figure 14 for an illustration). If $F \cup \{g\} \setminus \{e_2, f_1\}$ is not a 2-VCSS (hence a 2-ECSS), then we modify F (and $\overline{F}$) by switching to $F \cup \{g\} \setminus \{e_1\}$. This breaks the configuration of the edges e_1, e_2, f_1, and does not change the size of F and $\overline{F}$ (See Figure 15 for an illustration). □

Lemma 4. *There exists G_1, a 2-VCSS $F_1 \subseteq E(G_1)$, and a 2-ECSS $\overline{F_1} \subseteq E(G_1)$ such that the following hold:*

1. *$\overline{F_1}$ is obtained from F_1 by deleting redundant edges.*
2. *For any side vertex s of a strong short segment S on F_1, there is no edge $e \in E(G_1) \setminus F_1$ incident to s.*
3. $\frac{|F_1|}{opt(G_1)} \leq \frac{9}{7} \Rightarrow \frac{|F|}{opt(G)} \leq \frac{9}{7}$. $\frac{|\overline{F_1}|}{opt(G_1)} \leq \frac{9}{7} \Rightarrow \frac{|\overline{F}|}{opt(G)} \leq \frac{9}{7}$.
4. *F_1 is minimal with respect to inclusion.*

Proof. We reduce G to G_1 and F to F_1 by performing a series of operations. Let S be a strong short segment on F, and s be a side vertex of S. Let O be an optimal 2-ECSS on G. Clearly, O contains two edges incident to s, say e_1 and e_2. Assume it contains a third edge e_3 incident to s. Let the other end vertices of these edges be w_1, w_2, and w_3, respectively. By the improvement operations performed by the algorithm, none of these vertices is a side vertex of a strong short segment. If O contains all the edges incident to w_i that are in F, we call w_i a *special vertex*, for $i = 1, 2, 3$.

Claim 5. *There is at most one special vertex in the set* $\{w_1, w_2, w_3\}$.

Proof. Assume without loss of generality that w_1 and w_2 are special vertices. Then by the structure of a 2-ECSS, we can discard e_1 or e_2 from O without violating feasibility, which contradicts its optimality. □

Claim 6. *There exists an optimal 2-ECSS* O' *on* G *such that* O' *contains* 2 *edges incident to* s.

Proof. By Claim 5, there are at least two vertices in the set $\{w_1, w_2, w_3\}$ that are not special. Let two of them be without loss of generality w_2 and w_3. By the structure of a 2-ECSS, one of these vertices, say w_2, satisfies the following. There is a neighbor w_2' of w_2 such that $f = (w_2, w_2') \in F \setminus O$, and $O' = O \cup \{f\} \setminus \{e_2\}$ is another optimal solution. In this case the degree of s on O' is 2, which completes the proof. □

Let $O'(S)$ be the set of edges in this solution incident to the side vertices of S. Let $F' = F \cup O'(S) \setminus P$ be a minimal 2-VCSS on G, where $P \subseteq F \setminus O'(S)$. Examples of this operation are illustrated in Figure 16 and Figure 17. Let $E(S)$ denote the set of edges incident to the side vertices of S on $E(G)$, which excludes the edges in $O'(S)$. Note that this set contains P. Delete the edges in $E(S)$ from G to obtain G'. Perform these operations, including the switch to an optimal solution implied by Claim 6, recursively on the new strong short segments that appear on F', which we call *emerging segments*. Note that since none of the aforementioned vertices w_1, w_2, and w_3 can be a side vertex of a strong short segment as noted, the switch from O to O' cannot be reversed. After the recursion starting from S terminates, continue performing the described operations on the strong short segments on the residual solution and the graph. Let the results be F_1 and G_1. Given these, the second claim of the lemma holds, since there is no edge in $E(G_1) \setminus F_1$ incident to the side vertices of a strong short segment on F_1 by construction. Let $\overline{F_1}$ be a 2-ECSS obtained by deleting redundant edges from F_1, thus satisfying the first claim of the lemma. The fourth claim of the lemma holds by Claim 6. We now show that the third claim holds.

Claim 7. $|\overline{F_1}| \geq |\overline{F}|$.

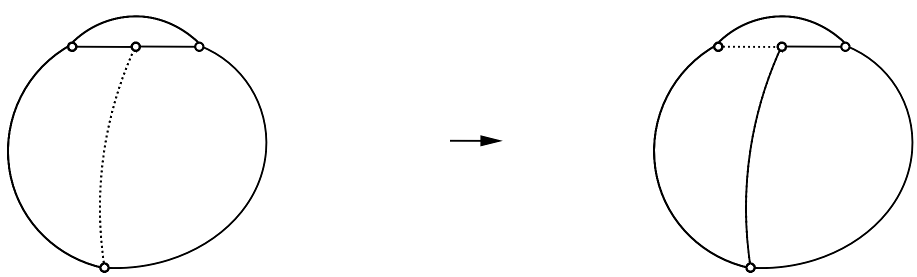


Figure 16: A transition from F to F_1 via a 2-segment with $|P| = 1$

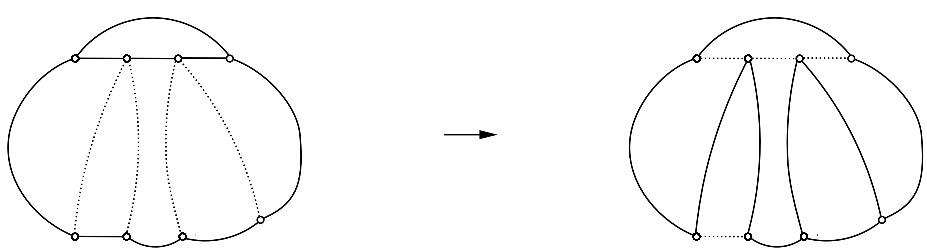


Figure 17: A transition from F to F_1 via a 3-segment with $|P| = 4$

Proof. Let S be a strong short segment on which we start the recursive operations above or an emerging segment. The inequality $|P| \leq |O'(S)|$ implies $|F_1| \geq |F|$. Indeed, by the algorithm and the construction of F_1, there is no improvement process performed on S that has improved the cost of the solution. In this case $|P| > |O'(S)|$ derives a contradiction. In particular, by all the listed improvement operations we cannot have the configurations on the left hand sides of Figure 1-Figure 13. Note next that if $e \in F_1 \setminus \overline{F_1}$, then we also have $e \in F \setminus \overline{F}$ by construction. This implies $|F| - |\overline{F}| \geq |F_1| - |\overline{F_1}|$, which is equivalent to $|\overline{F_1}| - |\overline{F}| \geq |F_1| - |F| \geq 0$. □

We next note that $opt(G_1) \leq opt(G)$. This follows from our construction ensuring that there is an optimal solution O such that for any strong short segment S on F_1, $E(S)$ does not contain any edge from O. Combining this with Claim 7, we obtain $\frac{|\overline{F}|}{opt(G)} \leq \frac{|\overline{F_1}|}{opt(G_1)}$, which establishes the third claim of the lemma, and completes the proof. □

Let G_1, F_1, and $\overline{F_1}$ be as implied by Lemma 4.

Lemma 8. *Let $e_1 = (u_1, v_1)$, $e_2 = (u_2, v_2)$, and $f_1 = (u_1, w_1)$ be the edges of three distinct trivial segments on F_1. Let S be a cut such that $\delta(S) \cap F_1 = \{e_1, e_2\}$. Then there is no edge $g \in E(G_1) \setminus F_1$ such that $g \in \delta(S)$.*

Proof. Follows from Lemma 3 and the fact that the operations described in the proof of Lemma 4, which define G_1 and F_1, do not introduce a trivial segment. □

Lemma 9. $\frac{|F_1|}{opt(G_1)} \leq \frac{9}{7} \cdot \frac{|\overline{F_1}|}{opt(G_1)} \leq \frac{9}{7}$.

Proof. We first construct a feasible dual solution in (EC-D) with total value at least $\frac{7}{9}|\overline{F_1}|$. Given a strong short segment S on F_1 and a side vertex s of S, we assign $y_{\{s\}} = 1$. If the segment is a 4-segment, then the middle vertex t of the segment is assigned the dual value $y_{\{t\}} = 0$. If the segment is a 3-segment, we assign the dual value $z_e = 1$ for the middle edge e of the segment to maintain feasibility. Let $y_{\{w\}} = 1/2$ for the internal vertices w of weak segments and strong long segments on F_1. Note that the overall assignment thus far is feasible by the second claim of Lemma 4. Let u be an end vertex of a weak segment. If u is not shared by any strong short segment, we assign $y_{\{u\}} = 1/2$. Otherwise, let S be a set satisfying the following:

1. $u \in S$,
2. S consists of vertices of strong short segments,
3. S induces a connected subgraph,
4. S is maximal.

We call S an *augmented set for u*. Assign $y_S = 1/2$. Note that this is also feasible by the stated properties of S and the second claim of Lemma 4. We call the dual variables $y_{\{u\}}$ and y_S *augmented dual variables.*

We will establish the lemma by a cost sharing argument in which the cost of a set of segments is paid by a unique set of dual values with ratio at least $\frac{7}{9}$. We call this the *cover ratio.* If the cover ratio of a set of segments or edges is 1, we say that they are *optimally covered.* Note that there is a distinction between a dual value in (EC-D), and twice its value in the objective function of (EC-D), which we compare with the cost. The latter is called the *dual contribution*, which we refer to from this point on. The strong short segments are paid by the dual contributions defined by their internal vertices and edges. They are optimally covered, since for a 2-segment the total

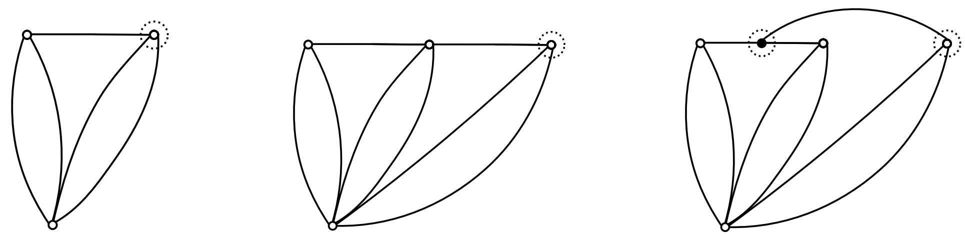


Figure 18

contribution is 2, for a 3-segment the total contribution is $2+2-1=3$, and for a 4-segment the total dual contribution is $2+2=4$. The cost of the strong long segments are paid by the dual contributions defined by their internal vertices, so that their cover ratio is at least $\frac{4}{5}$, since the cost of a long segment is at least 5. Given a weak ℓ-segment, the cost $\ell-1$ of the segment is paid by the dual contributions of its internal vertices. There remains the cost 1 to be covered associated to each weak segment, which we analyze below.

Let k be the number of weak segments on F_1, excluding the trivial segments defined by redundant edges. We argue by induction on k. The case $k=1$ is depicted in Figure 18a, where the weak segment is not a trivial segment. Let u be an end vertex of this weak segment. If u is not shared by any strong short segment, define $y_{\{u\}}=1/2$, which optimally covers the weak segment. Otherwise, we have the following cases.

1) There are at least 2 *short segments among the strong segments with an end vertex u:* Let L be the total cost of the strong long segments. Then the total cost of the strong segments and the weak segment is $p+L+1$, and the total dual contribution is at least $p+\frac{4L}{5}$, where $p\geq 4$. Thus, the cover ratio satisfies

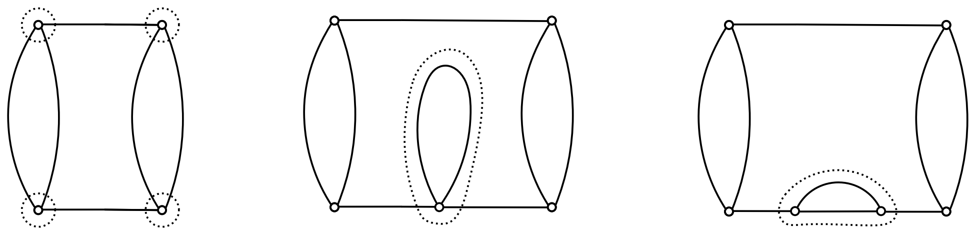
$$\frac{p+\frac{4L}{5}}{p+L+1}\geq \min\left\{\frac{p}{p+1},\frac{4}{5}\right\}\geq \min\left\{\frac{4}{5},\frac{4}{5}\right\}=\frac{4}{5}.$$

2) There is at most one short segment among the strong segments with an end vertex u: In this case define $y_{\{u\}}=1/2$, and $z_e=1/2$ for the edge e between u and the neighboring side vertex of the short segment. Let L be the total cost of the strong long segments. Then the total cost of all the strong segments and the weak segment is $p+L+1$, and the total dual contribution is $p+\frac{4L}{5}+\frac{1}{2}$, where $p\geq 2$. The cover ratio then satisfies

$$\frac{p+\frac{4L}{5}+\frac{1}{2}}{p+L+1}\geq \min\left\{\frac{p+\frac{1}{2}}{p+1},\frac{4}{5}\right\}\geq \min\left\{\frac{5}{6},\frac{4}{5}\right\}=\frac{4}{5}.$$

The inductive step might introduce one or two new weak segments as depicted in Figure 18b and Figure 18c. The analysis of the configuration in Figure 18b is identical to that of the base case. Figure 18c additionally contains a vertex incident to only weak segments on which we define a dual of value 1/2, so that the two new weak segments are optimally covered.

For $k=2$, we have that an end vertex of a weak segment is only shared by strong segments. This is depicted in Figure 19a. It is clear for $k=2$ that the number of augmented dual variables is at least 2, which optimally covers the two weak segments. Figure 19a depicts this case, where the end vertices of the weak segments are not shared by strong short segments. This establishes the base case $k=2$.



(a) **(b)** **(c)**

Figure 19

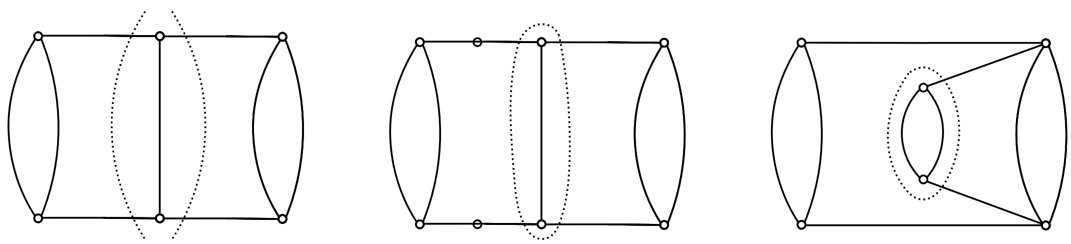
In the inductive step one may introduce one, two, three, or four new weak segments by extending the graph in the induction hypothesis. These are depicted in Figure 19, Figure 20, and Figure 21, where we show the extending subgraphs in their simplest form, thus without depicting all possible cases, which would lead to the same analysis. In Figure 19b and Figure 19c one new weak segment is introduced. Let u be a newly introduced high-degree vertex. If there is no strong short segment with an end vertex u, we assign $y_{\{u\}} = 1/2$. Otherwise, we assign $y_S = 1/2$, where S is an augmented set for u. This dual is shown via a dotted oval in the figures. Note that this dual assignment is feasible by the second claim of Lemma 4, a fact we will also be tacitly using in the following cases. The total dual contribution in both configurations is thus 1, which optimally covers the new weak segment.

In Figure 20a two new weak segments are introduced, where out of the four weak segments in the configuration at least three of them are trivial segments. In this case Lemma 8 implies that there is no $g \in E(G_1) \setminus F_1$ such that the cuts shown via dotted curves in the figure cross g. We thus assign the duals representing these cuts the value 1/2, which maintains feasibility. Their total dual contribution is 2, which optimally covers the newly introduced two weak segments. The generalization of this case, analogous to the one from Figure 19b to Figure 19c, is not depicted, as it does not change the analysis.

Figure 20b depicts an example of the introduction of two new weak segments, where there are at most two trivial segments after the extension. In this case we cannot use Lemma 8. Instead, we assign $y_S = 1/2$ if there is an augmented set S for one of the newly introduced high-degree vertices. This dual is shown via a dotted oval in the figure. Observe now that the induction hypothesis holds even if the two weak segments before the extension are trivial segments. Then we have that the total dual contribution in the induction hypothesis is at least $\frac{7\ell}{9}$, where ℓ is the cost assuming that there are two trivial segments before the extension. By the assumption of this case however, the cost is $\ell + \ell'$ and the total dual contribution is $\frac{7\ell}{9} + \ell'$, where $\ell' \geq 2$. The set S contains at least one optimally covered strong short segment of length at least 2, and y_S contributes 1, so that there is a total contribution of $p - 1$ from the duals associated to S, whereas the cost of S together with the new weak segments is p, with $p \geq 4$. We thus have total cost $\ell + p + \ell'$, and the total dual contribution $\frac{7\ell}{9} + p + \ell' - 1$, yielding

$$\frac{\frac{7\ell}{9} + p + \ell' - 1}{\ell + p + \ell'} \geq \min\left\{\frac{7}{9}, \frac{p + \ell' - 1}{p + \ell'}\right\} \geq \min\left\{\frac{7}{9}, \frac{5}{6}\right\} = \frac{7}{9}.$$

If there is no such augmented set S, then there is a long segment containing one of the newly introduced high-degree vertices. By definition, there are $r \geq 6$ new vertices on this long segment, where the total cost together with the weak segments is $(r-1)+2 = r+1$. We define the dual value 1/2 on the end vertices of the long segment, so that the cover ratio is at least $\frac{6}{7}$. The generalization



(a) (b) (c)

Figure 20

of Figure 20b, analogous to the one from Figure 19b to Figure 19c, is not depicted, as it does not change the analysis.

Figure 20c shows another case for the introduction of two new weak segments. There are at least two new strong segments in this case. We have the following sub-cases for these:

1) All of the new strong segments are short: We assign $y_S = 1/2$ for the set S containing the vertices of these segments. This dual is shown via a dotted oval in the figure. In this case the total dual contribution introduced in the inductive step is $p+1$, and the total cost is $p+2$, where the cost $p \geq 4$ of the short segments are optimally covered as noted in the base case. The cover ratio of all the new segments is thus at least $\frac{5}{6}$.

2) Some of the new strong segments are short, and some are long: We keep $y_S = 1/2$. Let R be a set consisting of all the long segments among the new strong segments except one long segment L. The cover ratio of R is at least $\frac{4}{5}$ as noted previously. Among the remaining new segments, let p_2 be the number of 2-segments, p_3 be the number of 3-segments, p_4 be the number of 4-segments, and ℓ be the length of L. The total cost of the new segments excluding R is then $2p_2 + 3p_3 + 4p_4 + \ell + 2$. Given the dual assignment for the strong segments described in the base case, the total dual contribution is $2p_2 + 3p_3 + 4p_4 + \ell$. We then obtain

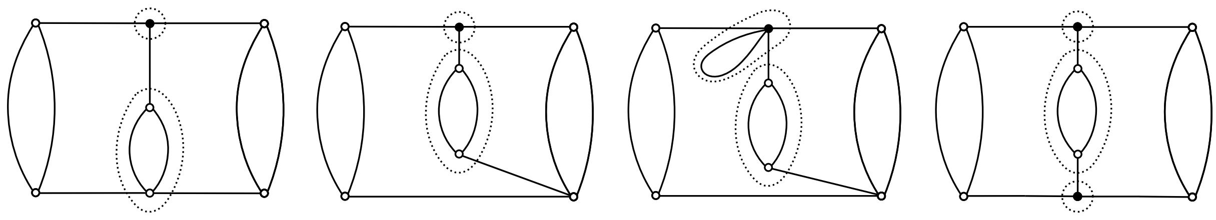
$$\frac{2p_2 + 3p_3 + 4p_4 + \ell}{2p_2 + 3p_3 + 4p_4 + \ell + 2} \geq \frac{2+\ell}{\ell+4} \geq \frac{7}{9},$$

which follows from the fact that $p_2 + p_3 + p_4 \geq 1$ and $\ell \geq 5$.

3) All of the strong segments are long: In this case there exists at least two new long segments. There are $p \geq 10$ vertices belonging to these segments, and the total cost together with the weak segments is $p+2$. This leads to a cover ratio of at least $\frac{5}{6}$.

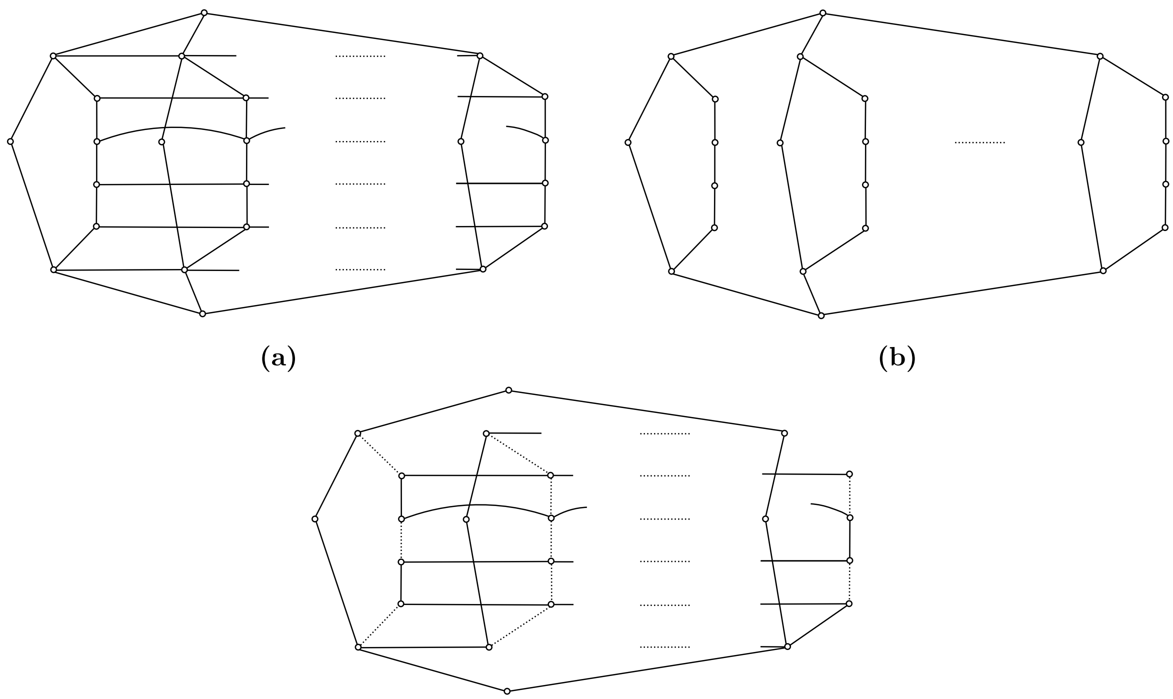
In Figure 21a and Figure 21b three new weak segments are introduced. The vertex incident to three weak segments is assigned the dual value 1/2. Similar to the previous case, there are at least two new strong segments here. The analysis essentially reduces to that of the previous case as follows. Both the cost and the dual contribution of a specific sub-case of the previous case is incremented by 1 due to the introduction of one new weak segment and the assignment of the new vertex. The ratios proved in the previous case are thus retained. Figure 21c is a generalization of Figure 21b, where the vertex incident to three weak segments is also incident to another subgraph. In this case we apply the assignment in Figure 19b, which reduces the analysis to that of the previous case.

In Figure 21d four new weak segments are introduced. There are two vertices, each incident to distinct sets of three weak segments. These vertices are assigned the dual value 1/2. Similar to the case depicted in Figure 21b, there are at least two new strong segments here. The analysis thus reduces to that of Figure 20c for the same reason as stated for the case in Figure 21b. The



(a) (b) (c) (d)

Figure 21



(a) (b)

(c)

Figure 22: A tight example for the algorithm: (a) Input graph; (b) A solution returned by the algorithm; (c) An optimal solution.

generalizations of Figure 21d, analogous to the one from Figure 21b to Figure 21c, are not depicted, as it does not change the analysis.

We now show that $\frac{|F_1|}{opt(G_1)} \leq \frac{9}{7}$. Note that both the base case and the inductive step for a 2-ECSS are subsets of those for a 2-VCSS except the one depicted in Figure 19b. The set of configurations specific to a 2-VCSS are those depicted in Figure 18a, Figure 18b, and Figure 18c, where the weak segments are trivial segments. The analysis for these figures have already been given, which completes the proof. □

The following completes the proof of Theorem 1.

Theorem 10. $\frac{|F|}{opt(G)} \leq \frac{9}{7}$. $\frac{|\overline{F}|}{opt(G)} \leq \frac{9}{7}$.

Proof. Follows from Lemma 9 and the third claim of Lemma 4. □

5 A Tight Example

A tight example for the algorithm is given in Figure 22, which is derived from the second case of the analysis of the configuration in Figure 20c. The solution returned by the algorithm has cost $9k$, where k is an even integer. Note that there is no improvement performed on this solution. The optimal solution has cost $7k+3$, where approximately $4k$ of it covers the 5-segments of the solution by the algorithm, and $3k$ of it covers the 2-segments of that solution.

Acknowledgment

This work was supported by Scientific and Technological Research Council of Turkey (TUBITAK) under the Grant Number 123E530. The author thanks TUBITAK for their supports.

References

[1] M. Bosch-Calvo, F. Grandoni, and A. J. Ameli. A 4/3 approximation for 2-vertex-connectivity. In *50th International Colloquium on Automata, Languages, and Programming, ICALP*, pages 29:1–29:13, 2023.

[2] J. Cheriyan, A. Sebö, and Z. Szigeti. Improving on the 1.5-approximation of a smallest 2-edge connected spanning subgraph. *SIAM J. Discrete Math.*, 14(2):170–180, 2001.

[3] J. Cheriyan and R. Thurimella. Approximating minimum-size k-connected spanning subgraphs via matching. *SIAM J. Comput.*, 30(2):528–560, 2000.

[4] B. Csaba, M. Karpinski, and P. Krysta. Approximability of dense and sparse instances of minimum 2-connectivity, TSP and path problems. In *Proceedings of the Thirteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 74–83, 2002.

[5] A. Czumaj and A. Lingas. On approximability of the minimum-cost k-connected spanning subgraph problem. In *Proceedings of the Tenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 281–290, 1999.

[6] M. Garg, F. Grandoni, and A. J. Ameli. Improved approximation for two-edge-connectivity. In *Proceedings of the 2023 ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 2368–2410, 2023.

[7] N. Garg, S. Vempala, and A. Singla. Improved approximation algorithms for biconnected subgraphs via better lower bounding techniques. In *Proceedings of the Fourth Annual ACM/SIGACT-SIAM Symposium on Discrete Algorithms (SODA)*, pages 103–111, 1993.

[8] P. Gubbala and B. Raghavachari. Approximation algorithms for the minimum cardinality two-connected spanning subgraph problem. In *Integer Programming and Combinatorial Optimization, 11th International IPCO Conference*, pages 422–436, 2005.

[9] K. Heeger and J. Vygen. Two-connected spanning subgraphs with at most $\frac{10}{7}$OPT edges. *SIAM J. Discrete Math.*, 31(3):1820–1835, 2017.

[10] C. Hunkenschröder, S. Vempala, and A. Vetta. A 4/3-approximation algorithm for the minimum 2-edge connected subgraph problem. *ACM Trans. Algorithms*, 15(4):55:1–55:28, 2019.

[11] R. Jothi, B. Raghavachari, and S. Varadarajan. A 5/4-approximation algorithm for minimum 2-edge-connectivity. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 725–734, 2003.

[12] S. Khuller and U. Vishkin. Biconnectivity approximations and graph carvings. *J. ACM*, 41(2):214–235, 1994.

[13] P. Krysta and V. S. A. Kumar. Approximation algorithms for minimum size 2-connectivity problems. In *18th Annual Symposium on Theoretical Aspects of Computer Science (STACS)*, pages 431–442, 2001.

[14] P.Gubbala. *Problems in Graph Connectivity*. PhD thesis, Department of Computer Science, University of Texas at Dallas, Richardson, TX, 2006.

[15] A. Sebö and J. Vygen. Shorter tours by nicer ears: 7/5-approximation for the graph-TSP, 3/2 for the path version, and 4/3 for two-edge-connected subgraphs. *Combinatorica*, 34(5):597–629, 2014.

[16] S. Vempala and A. Vetta. Factor 4/3 approximations for minimum 2-connected subgraphs. In *Approximation Algorithms for Combinatorial Optimization, Third International Workshop (APPROX)*, pages 262–273, 2000.