Structure Theorem for Steiner tree

0.1 Statement of subroutine lemmas

For the following three lemmas, G is a planar embedded graph, P is an $1+\epsilon$-short path forming part of the boundary of G, and T is a tree in G that intersects P only at leaves of T.

The first lemma is illustrated in Figure 1.

Lemma 0.1.1 *There is a procedure* Span0 *such that* Span0(P, T) *returns a subpath of P spanning $V(T) \cap V(P)$ whose total length is at most* $(1+\epsilon)$*length*(T).

Figure 1: A subgraph is squished to a $1+\epsilon$-short path. The resulting subpath includes all vertices common to the subgraph and the path, and is not much longer.

Proof: Let P' be the shortest subpath of P that spans all the vertices of $T \cap P$. There is a path Q in T between the endpoints of T. Since P is $1+\epsilon$-short, $\text{length}(P') < (1+\epsilon)\text{length}(Q) \leq (1+\epsilon)\text{length}(T)$. $\square$

Definition 0.1.2 (Joining vertex) *Let H be a subgraph of G such that P is a path in H. A* joining vertex *of H with P is a vertex of P that is the endpoint of an edge of $H - P$.*

The second lemma is illustrated in Figure 2. The proof is given in Section 0.2.7.

Lemma 0.1.3 *There is a procedure* Span1(P, T, r) *that, for a vertex f of T, returns a subgraph of $P \cup T$ of length at most* $(1+\epsilon)$*length*(T) *that spans all the vertices of $\{r\} \cup (V(T) \cap V(P))$ and has at most $\epsilon^{-1.45}$ joining vertices with P.*

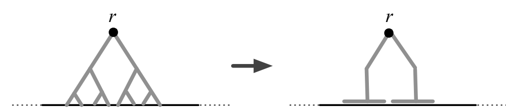


Figure 2: The output subgraph spans all vertices of P spanned by the input subgraph, and also spans x, but the output subgraph has fewer joining vertices with P.

The third lemma is illustrated in Figure 3. The proof is given in Section 0.2.8.

Lemma 0.1.4 *There is a procedure* $\textsc{Span2}(P, T, x, y)$ *that, for* x *and* y *vertices of* T*, returns a subgraph of* $P \cup T$ *of length at most* $(1 + 2\epsilon + \epsilon^2)\mathit{length}(T)$ *that spans all the vertices of* $\{x, y\} \cup (T \cap P)$ *and has at most* $2\epsilon^{-2.5}$ *joining vertices with* P*. are constants.*

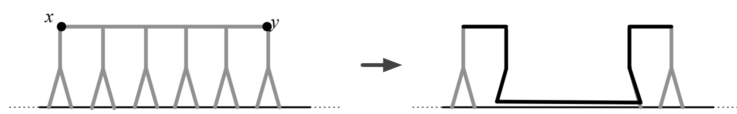


Figure 3: The output subgraph spans all vertices of P spanned by the input tree, and also spans x and y, but the output subgraph has fewer joining vertices with P.

0.2 Structure of Steiner tree within bricks

Lemma 0.2.1 *The counterclockwise boundary of a brick* B *equals* $W_B \circ S_B \circ E_B \circ N_B$*, where*

1. N_B *is 1-short in* B*, and every proper subpath of* S_B *is* $(1 + \epsilon)$*-short in* B*.*
2. $S_B = S_1 \circ S_1 \circ \cdots \circ S_\kappa$ *where, for each vertex* x *of* $S_i[\cdot, \cdot)$*,*

$$\mathit{length}(S_i[\cdot, x]) \leq \epsilon\, \mathit{dist}(x, N_B) \tag{1}$$

Note that some of the paths S_i might be empty.

Theorem 0.2.2 (Structural Property of Bricks) *Let* B *be a plane graph with boundary* $N \cup E \cup S \cup W$*, satisfying the brick properties of Lemma 0.2.1. Let* F *be a set of edges of* B*. There is a forest* $\tilde{F}$ *of* B *with the following properties:*

F1) *If two vertices of the boundary of* B *are connected in* F *then they are connected in* $\tilde{F}$*.*

F2) *The number of joining vertices of* $\tilde{F}$ *with* N *and with* S *is at most* $4(\kappa + 1)\epsilon^{-2.5}$*.*

F3) $\mathit{length}(\tilde{F}) \leq (1 + 5\epsilon)(\mathit{length}(F) + \mathit{length}(E) + \mathit{length}(W))$*.*

0.2.1 Paths $\bar{P}_0, \ldots, \bar{P}_\kappa$

We now present the proof of the theorem. We refer to N as *north*, etc. We define P_κ to be the eastern boundary E of the brick. We define $\bar{P}_\kappa = P_\kappa$. We inductively define $\bar{P}_{\kappa-1}, \bar{P}_{\kappa-2}, \ldots, \bar{P}_0$ as follows. (The definition is illustrated in Figure 4.) For $i = \kappa - 1, \kappa - 2, \ldots, 0$, if $F \cup W$ has an S_i-to-north path that does not intersect $\bar{P}_{i+1}, \bar{P}_{i+2}, \ldots, \bar{P}_\kappa$, let P_i be the rightmost such path, and define $\bar{P}_i = S_i[\cdot, \text{start}(P_i)] \circ P_i$. If there is no such path, define $\bar{P}_i = \emptyset$.

Let P be a nontrivial $\bar{P}_i$-to-north path or $\bar{P}_i$-to-south path in F. We call P a *sprit* of $\bar{P}_I$ if $P - \text{start}(P)$ avoids $\bar{P}_i, \ldots, \bar{P}_\kappa$. It is a *northern* sprit if end(P) belongs to N and a *southern* sprit if end(P) belongs to S.

Because P_i is rightmost, we obtain the following lemma, whose proof is outlined in Figure 5.

Lemma 0.2.3 (Sprit Lemma) *For $i = 0, 1, \ldots, \kappa$,*

- *if P is a northern sprit of $\bar{P}_i$ then end(P) is strictly left of end(P_i) on N, and*
- *if P is a southern sprit of $\bar{P}_i$ then end(P) is strictly left of start(P_i) on S, and*

Inequality 1 implies that, for $i = 0, \ldots, \kappa - 1$,

$$\text{length}(\bar{P}_i) \leq (1 + \epsilon)\text{length}(P_i) \tag{2}$$

0.2.2 The forest F' and paths $Q_0, \ldots, Q_\kappa$

Let F' be a minimal subgraph of $F \cup W \cup \bigcup_{i=0}^{\kappa} \bar{P}_i$ that contains $\bigcup_i \bar{P}_i$ and that preserves connectivity among vertices of the boundary of B. Since F' is a subgraph of $F \cup W \cup \bigcup_{i=0}^{\kappa} \bar{P}_i$, Inequality 2 implies that

$$\begin{aligned} \text{length}(F') &\leq \text{length}(F \cup W \cup \bigcup_{i=0}^{\kappa} \bar{P}_i) \\ &\leq \text{length}(E) + \text{length}(W) + (1 + \epsilon)\text{length}(F) \end{aligned}$$

For $i = 0, \ldots, \kappa - 1$, if there is a path in F' from $\bar{P}_i$ to $\bar{P}_{i+1} \cup \bar{P}_{i+2} \cup \cdots \cup \bar{P}_\kappa$ whose internal vertices are not in $\bar{P}_i \cup \cdots \cup P_\kappa$, let Q_i be such a path, as shown in Figure 6. Otherwise let $Q_i = \emptyset$.

Claim 0.2.4 *Every internal vertex of Q_i has degree two in F'.*

Proof: Assume for a contradiction that some internal vertex u of Q_i has an incident edge e not on Q_i. By minimality of F', the edge e must be required to preserve connectivity among vertices of the boundary of B. Let v be a boundary vertex of B such that removing e separates u and v. Let P be the u-to-v path in F'.

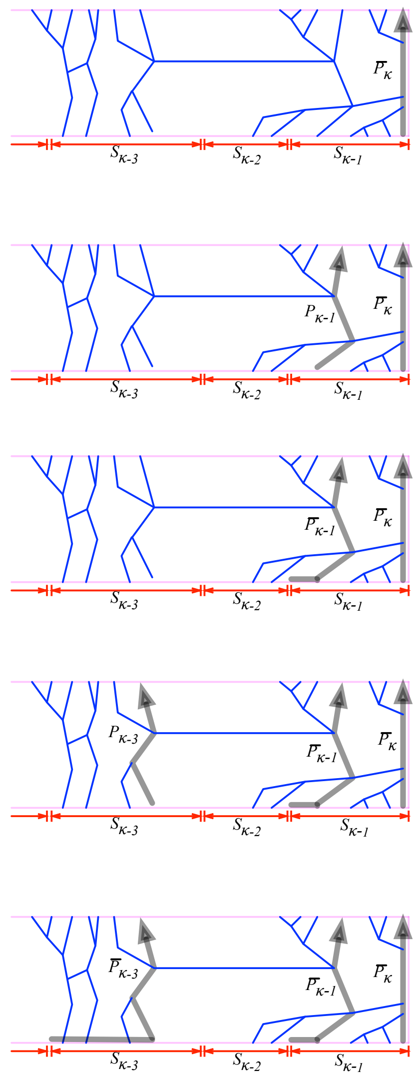


Figure 4: The top figure shows a fragment of a brick with $\bar{P}_\kappa$ defined as the eastern boundary. The second figure shows $P_{\kappa-1}$, defined as the rightmost south-to-north path that avoids P_κ. The third figure shows $\bar{P}_{\kappa-}$, which is obtained from $P_{\kappa-1}$ by prepending the to-start(P_κ) prefix of $S_{\kappa-1}$. There is *no* south-to-north path that originates in $S_{\kappa-2}$ and avoids $\bar{P}_{\kappa-1}$, so $\bar{P}_{\kappa-2}$ is empty. The fourth figure shows $P_{\kappa-3}$, defined as the rightmost south-to-north path that does not intersect $P_{\kappa-1}$ or P_κ. The fifth figure shows $\bar{P}_{\kappa-3}$, which is obtained from $P_{\kappa-3}$ by prepending a prefix of $S_{\kappa-3}$. Note that south-to-north paths originating in this prefix become $\bar{P}_{\kappa-3}$-to-north paths.

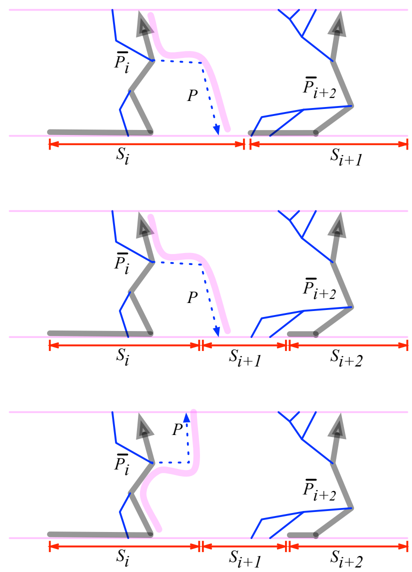


Figure 5: Suppose F contains a path P from P_i to a southern vertex to the right of start(P_i) (in the first and second figures) or from P_i to a northern vertex to the right of end(P_i) (in the third figure). In the first and third figure, the magenta contour indicates that P_i is not the rightmost S_i-to-north path avoiding $\bar{P}_{i+1}, \dots, \bar{P}_\kappa$, a contradiction. In the second figure, end(P) belongs to S_{i+1}. Ordinarily end(P) would therefore belong to $\bar{P}_{i+1}$, but in this case $\bar{P}_{i+1} = P_{i+1} =$. However, the magenta contour indicates that there is an S_{i+1}-to-north path avoiding $\bar{P}_{i+2}, \dots, \bar{P}_\kappa$, a contradiction.

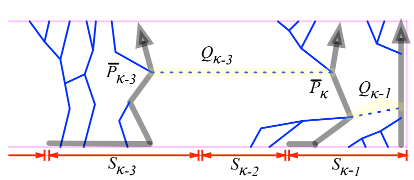


Figure 6: The paths $Q_{\kappa-1}$ and $Q_{\kappa-3}$ are signified by the dashed lines.

If v is on $\bar{P}_j$ for some $j > i$ then u and v are connected via $\bar{P}_j$ and a suffix of Q_i, a contradiction. Otherwise, a prefix of Q_i together with P violates the Sprit Lemma (Lemma 0.2.3). $\square$

0.2.3 The forest $\tilde{F}$

The construction of $\tilde{F}$ is as follows. Each connected component K of $F' - \bigcup_i Q_i$ is replaced with a component $\tilde{K}$ that achieves at least K's connectivity among boundary vertices of B and endpoints of paths Q_i. This ensures that $\tilde{K} = \bigcup_K \tilde{K} \cup \bigcup_i Q_i$ achieves the connectivity of F' among boundary vertices of B, which is property F1 of Theorem 0.2.2.

For each component K, moreover, $\text{length}(\tilde{K}) \leq (1 + 3\epsilon + \epsilon^2)\text{length}(K)$. Therefore

$$\begin{aligned}
\text{length}(\tilde{F}) &\leq \sum_i \text{length}(Q_i) + \sum_K \text{length}(\tilde{K}) \\
&\leq \sum_i \text{length}(Q_i) + (1 + 3\epsilon + \epsilon^2) \sum_K \text{length}(K) \\
&\leq (1 + 3\epsilon + \epsilon^2)\text{length}(F') \\
&\leq (1 + 3\epsilon + \epsilon^2)((1 + \epsilon)\text{length}(F) + \text{length}(E) + \text{length}(W))
\end{aligned}$$

which proves part F3 of the theorem, assuming $\epsilon \leq 1/5$.

Our construction will ensure that there are at most $\kappa + 1$ components K for which $\tilde{K}$ has joining vertices with the boundary of B, and for each of these components, $\tilde{K}$ has at most $4\epsilon^{-2.5}$ joining vertices. Thus the total number of joining vertices is $4(\kappa + 1)\epsilon^{-2.5}$.

0.2.4 Type-1 and type-2 components

For each connected component K of $F' - \bigcup_i Q_i$, the construction of $\tilde{K}$ depends on what kind of component it is. We say K is a *type-1* component if the boundary vertices in K are all internal vertices of S or all internal vertices of N, and is a *type-2* component otherwise.

For $i = 0, \ldots, \kappa$, let K_i be the connected component of $F' - \bigcup_j Q_j$ that contains $\bar{P}_i$ (if $\bar{P}_i \neq \emptyset$).

Lemma 0.2.5 *If K is a type-2 component then $K = K_i$ for some i.*

Proof: By minimality of F', every component of F' contains some boundary vertices.

- Suppose K contains a vertex of E. Since $\bar{P}_\kappa = E$ and $\bar{P}_\kappa$ belongs to $F - \bigcup_i Q_i$, K contains a vertex of S (namely $\text{start}(\bar{P}_\kappa)$) and a vertex of N (namely $\text{end}(\bar{P}_\kappa)$).

- Suppose K contains a vertex of W. Recall that F' is a subgraph of $F \cup W \cup \bigcup_i \bar{P}_i$ that preserves connectivity among vertices of the boundary. It follows that K contains a vertex of S and a vertex of W.

- Suppose K does not contain a vertex of E and a vertex of W. Since K is not a type-1 component, it must therefore contain a vertex of S and a vertex of N.

K, therefore, contains a vertex of S and a vertex of N. Let P be a south-to-north path in K, and let i be the integer such that start(P) belongs to $S_i[\cdot, \cdot)$. If start(P) belongs to $S_i[\cdot, \text{start}(P_i)]$ then start(P) belongs to $\bar{P}_i$, so start(P) belongs to K_i. If not, then, by choice of P_i, the rightmost P intersects $\bar{P}_j$ for some $j > i$, so start(P) belongs to K_j. □

0.2.5 Construction of $\tilde{K}$

First suppose K is of type 1. If its boundary vertices are in S, we let $\tilde{K} :=$ Span0(S, K). If its boundary vertices are in N, we let $\tilde{K} :=$ Span0(N, K). In either case, $\tilde{K}$ has no joining vertices, and length$(\tilde{K}) \leq (1+\epsilon)$length$(K)$.

Now we consider type-2 components. By Lemma 0.2.5, $K_0, \ldots, K_\kappa$ are the only type-2 components. For $i = 0, \ldots, \kappa$, we obtain $\tilde{K}_i$ from K_i by

(a) separating K_i into two parts, K_i^N and K_i^S,

(b) applying Span2 or Span1 to each part, and

(c) adding a subpath of $S_i[\cdot, \text{start}(P_i)]$.

By Lemmas 0.1.3 and 0.1.4, the total number of joining vertices is at most $4\epsilon^{-2.5}$, and the total length is at most $(1+2\epsilon+\epsilon^2)$length(K_i) + length$(S_i[\cdot, \text{start}(P_i)])$, which is in turn at most $(1+3\epsilon+\epsilon^2)$length(K_i). These are the properties needed in the analysis in Section 0.2.3.

0.2.6 Decomposition of K_i into K_i^N and K_i^S

For $i = 0, \ldots, \kappa$, if $\bar{P}_i \neq \emptyset$, let x_i be the first vertex on $\bar{P}_i$ such that there is an x_i-to-north sprit P^N. If end(P^N) were right of end$(\bar{P}_i)$ on N then it would violate the Sprit Lemma, so it is strictly left of end$(\bar{P}_i)$ on N.

Lemma 0.2.6 *For any vertex x of $\bar{P}_i(x_i, \cdot]$, there is no x-to-south sprit of P_i.*

Proof: Let P^N be an x_i-to-north sprit. Suppose P is an x-to-south sprit. By the Sprit Lemma, P^N and P are both left of $\bar{P}_i$, so they cross, forming a cycle with $\bar{P}_i$. This contradicts the minimality of F'. □

Lemma 0.2.7 *If there is an integer $j < i$ such that Q_j connects to $\bar{P}_i$ then end$(Q_j) = x_i$.*

Proof: The proof is illustrated in Figure 7. Combining Q_j with the to-start(Q_j) prefix of P_j yields a southern spar of $\bar{P}_i$. Therefore, by Lemma 0.2.6, end(Q_j) is not strictly after x_i on $\bar{P}_i$. Combining Q_j with the from-start(Q_j) prefix of P_j yields a northern spar of $\bar{P}_i$. Therefore, by choice of x_i, end(Q_j) is not strictly before x_i on $\bar{P}_i$. □

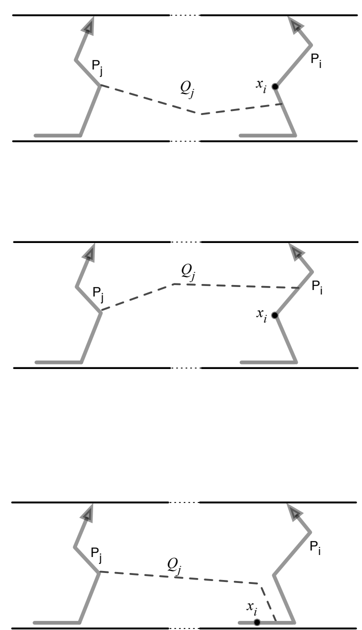


Figure 7: The first configuration is impossible since rev(Q_j) connects P_i to N (via P_j), which would contradict the choice of x_i. The second and third configurations are impossible since there is no way for an x_i-to-N path to avoid crossing Pj or Q_j.

As illustrated in Figure 8, we decompose K_i into edge-disjoint subgraphs K_i^N and K_i^S as follows. K_i^N consists of the subpath $\bar{P}_i[x_i, \cdot]$ and paths between this subpath and N. K_i^S consists of the subpath $\bar{P}_i[\cdot, x_i]$ and paths between this subpath and S.

Our intention is to apply the procedure Span1 or Span2 to each of K_i^N and K_i^S, as shown in Figure 9, obtaining edge-disjoint trees $\tilde{K}_i^N$ and $\tilde{K}_i^S$, each having at most $2\epsilon^{-2.5}$ joining vertices with N and S, respectively. Because each of these two trees contains x_i, their union is connected.

There are two additional issues, however. First, consider the case, depicted in Figure 10, in which the vertex x_i is not on P_i but is on the subpath $S_i[\cdot, \text{start}(P_i)]$ prepended to P_i to form $\bar{P}_i$. In this case, the tree $\tilde{T}_i^S$ is not required to include the vertices of $S_i[x_i, \text{start}(P_i)]$ other than x_i. In this case, therefore, we include this subpath in $\tilde{K}_i$.

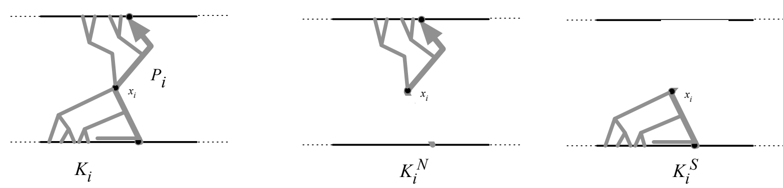


Figure 8: The component K_i is split at x_i into the northern part, K_i^N, and the southern part, K_i^S.

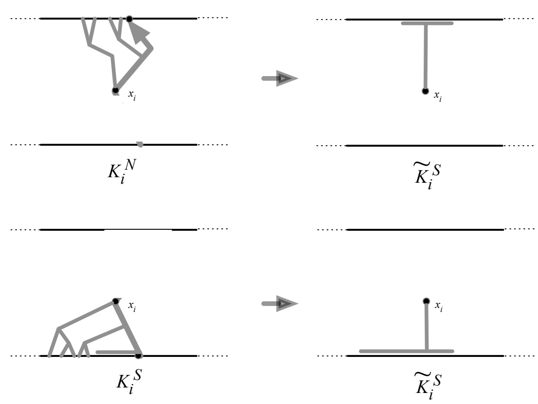


Figure 9: The northern subtree and the southern subtree are separately simplified (to reduce their number of joining vertices) using Span1.

The second issue is this: if $Q_i \neq \emptyset$, we need the new tree $\tilde{K}_i$ to include start(Q_i). Fortunately, the procedure Span2 allows us to specify *two* vertices to be spanned. The construction of $\tilde{K}_i$ is as follows. First we define $\tilde{K}_i^N$ and $\tilde{K}_i^S$:

- If $Q_i = \emptyset$, we define $\tilde{K}_i^N := Span1(N, K_i^N, x_i)$ and $\tilde{K}_i^S := Span1(S, K_i^S, x_i)$.
- If start(Q_i) belongs to K_i^N, we define $\tilde{K}_i^N := Span2(N, K_i^N, x_i, \text{start}(Q_i))$ and $\tilde{K}_i^S := Span1(S, K_i^S, x_i)$.
- Otherwise, we define $\tilde{K}_i^S := Span2(S, K_i^S, x_i, \text{start}(Q_i))$ and $\tilde{K}_i^N := Span1(N, K_i^N, x_i)$

We then define $\tilde{K}_i$ to be the union of $\tilde{K}_i^N$, $\tilde{K}_i^S$, and $S_i[x_i, \text{start}(P_i)]$. The analysis of length and number of joining vertices is as described in Section 0.2.5.

0.2.7 Span1

In this section, G is a planar embedded graph, P is an $1+\epsilon$-short path forming part of the boundary of G, r is a vertex of G, and T is an r-rooted tree of G that intersects P only at leaves of T.

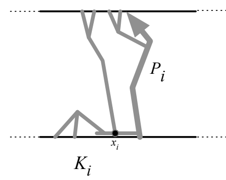


Figure 10: When x_i does not belong to P_i, the tree T_i^S does not include the vertices of $S_i(x_i, \text{start}(P_i)]$ so $\tilde{T}_i^S$ need not. In this case, therefore, we include the subpath $S[x_i, \text{start}(P_i)]$ in $\tilde{T}$.

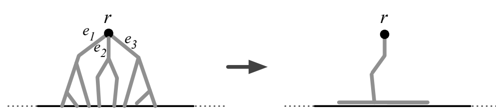


Figure 11: Replace the tree T with the minimal subpath of P that contains all leaves of T, together with the path to P that starts at a middle child edge of the root.

For a rooted subtree T' of T, every root-to-leaf path of T' ends on P, so these paths are ordered according to the positions of the leaves along P. In this section and the next, we are particularly interested in the leftmost and rightmost root-to-leaf paths.

In this section, we give the proof of Lemma 0.1.3, which is paraphrased here:

> *There is a procedure* Span1(P, T, r) *that returns a subgraph of* $T \cup P$ *that (a) has length at most* $(1+\epsilon)$*length*(T)*, (b) spans all the vertices of* $\{r\} \cup (V(T) \cap V(P))$*, and (c) has at most* $\epsilon^{-1.45}$ *joining vertices with* P*.*

We start with a subprocedure.

Lemma 0.2.8 *There is a subprocedure* ReduceDegree(P, T, r) *that, if the root* r *of* T *has more than two children, returns a subpath* P' *of* P *and an* r*-to-*P' *path* Q *consisting of edges of* T *such that that*

- $P' \cup Q$ *spans* $\{r\} \cup (V(T') \cap V(P))$*, and*
- *length*$(P') \leq (1+\epsilon)$*length*$(T - Q)$.

Proof: Let Q_1 and Q_3 denote, respectively, leftmost and rightmost root-to-leaf paths in T, and let e_1 and e_3 be the first edge in, respectively Q_1 and Q_2. Because G is planar, P is on the boundary of G, and r has at least two children, we have $e_1 \neq e_3$. Let e_2 be another child edge of r, and let Q_2 be the root-to-leaf path in T that starts with e_2.

The procedure returns the tree consisting of Q_2 and the minimal subpath of P that contains all leaves of T. The only joining vertex is the end of Q_2. Since

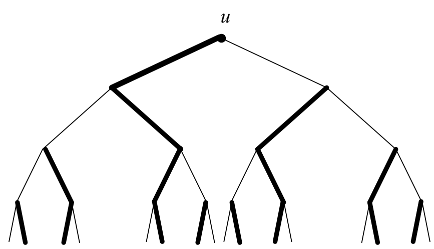


Figure 12: The edges in bold are the zig-zag edge.

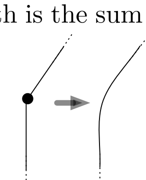
rev$(Q_1) \circ Q_3$ is a start(P')-to-end(P') path and P is $1+\epsilon$-short, length$(P') \leq$ length$(Q_1) +$ length(Q_3). $\square$

By repeated application of ReduceDegree, we obtain

Lemma 0.2.9 *There is a subprocedure* ReduceDegrees(P, T, r) *that returns a subtree T' of T and a collection of subpaths $P_1, \ldots, P_k$ of P such that*

- $T' \cup \bigcup_i P_i$ *spans* $\{r\} \cup (V(T) \cap V(P))$ *and*
- *length*$(\bigcup_i P_i) \leq (1+\epsilon)$*length*$(T - T')$

Now we prove Lemma 0.1.3 by describing Span1(P, T, r). Let T' be the tree derived from T in Lemma 0.2.9. Every vertex of T' has at most two children. We will use an argument that requires that every nonleaf vertex has two children. We therefore modify T' by splicing out each nonroot vertex with exactly one child, merging the two incident edges into a single edge whose length is the sum of the lengths of the merged edges.



This will ensure that every nonleaf vertex (except possibly the root r) has two children.

- If r has two children, we define T'' to be the resulting modified tree. We show how to replace T'' with an r-rooted tree $\widehat{T}$ that satisfies properties (a) through (c) of Lemma 0.1.3.
- If r has only one child, r', we define T'' to be the r'-rooted subtree, and apply the argument of Case 1 to obtain a replacement r'-rooted tree $\widehat{T}$. Then $\widehat{T} \cup \{r\text{-to-}r' \text{ path}\}$ satisfies properties (a) through (c) of Lemma 0.1.3.

Say that an edge uv of T'' is a *zig-zag* edge if the two-step path from the parent $p(u)$ of u to u to v either goes from $p(u)$ to a left child and from u to a right child, or goes from $p(u)$ to a right child and from u to a left child.

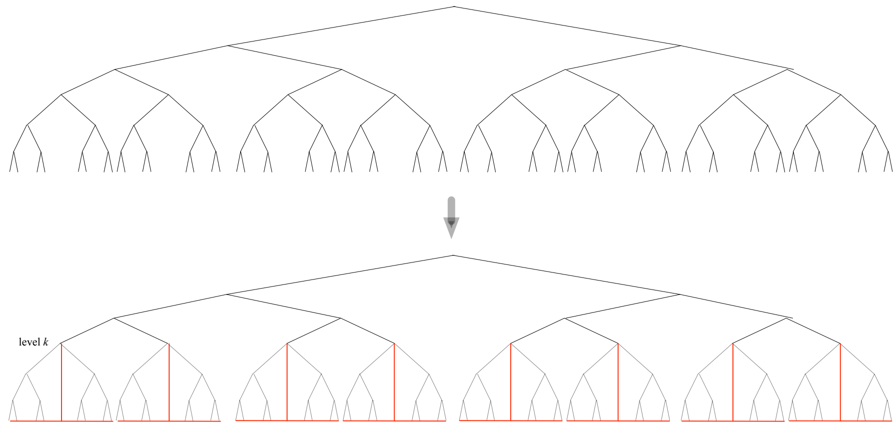


Figure 13: For each level-k vertex u, the subtree rooted at u is replaced with the minimal subpath P' of P containing the leaves of that subtree, together with a shortest u-to-P' path. The new subtrees are indicated in red.

The above definition is inapplicable if u is the root of T'. Therefore we (rather arbitrarily) define the left edge of the root of T'' to be a zig-zag edge.

As in breadth-first search, the *level* of a vertex is equal to the number of edges traversed when going from the root of T' to the vertex. The level of an edge is equal to the level of its endpoint that is closer to the root. For each level i, let L_i denote the total length of the zig-zag edges at level i.

Let k be a level to be determined later. The procedure obtains a tree $\widehat{T}$ from T'' as follows (see Figure 13). For each level-k vertex u, the procedure applies a subprocedure similar to ReduceDegree: replace the u-rooted subtree of T'' (which we denote T''_u) with another u-rooted tree $\widehat{T}_u$ consisting of

- the minimal subpath P' of P spanning the vertices of $T''_u \cap P$, and
- the u-to-P' path that includes u's zig-zag child edge.

The construction ensures that $\widehat{T}$ spans all the vertices of $V(T'') \cap V(P)$. Moreover, the number of joining vertices is 2^k. We shall ensure that $k \leq \log_\phi \epsilon^{-1}$, where $\phi = \frac{1+\sqrt{5}}{2}$ is the golden ratio. Hence the number of joining vertices is at most $\epsilon^{-1.45}$.

It remains to show that there is a choice of k for which $\text{length}(\widehat{T}) \leq (1+\epsilon)\text{length}(T'')$. The argument is illustrated in Figure 14. The length of the path P' is not much longer than the path Q_1 through T''_u between the endpoints of P'. The length of the shortest u-to-P' path is no longer than any u-to-P' path Q_2 in T''_u. Thus

$$\text{length}(\widehat{T}_u) \leq \text{length}(Q_1) + \text{length}(Q_2) \tag{3}$$

Since Q_1 and Q_2 are contained in T''_u, we would like to argue that $\text{length}(Q_1) + \text{length}(Q_2) \leq \text{length}(T''_u)$. However, that might not be true because Q_1 and Q_2 overlap.

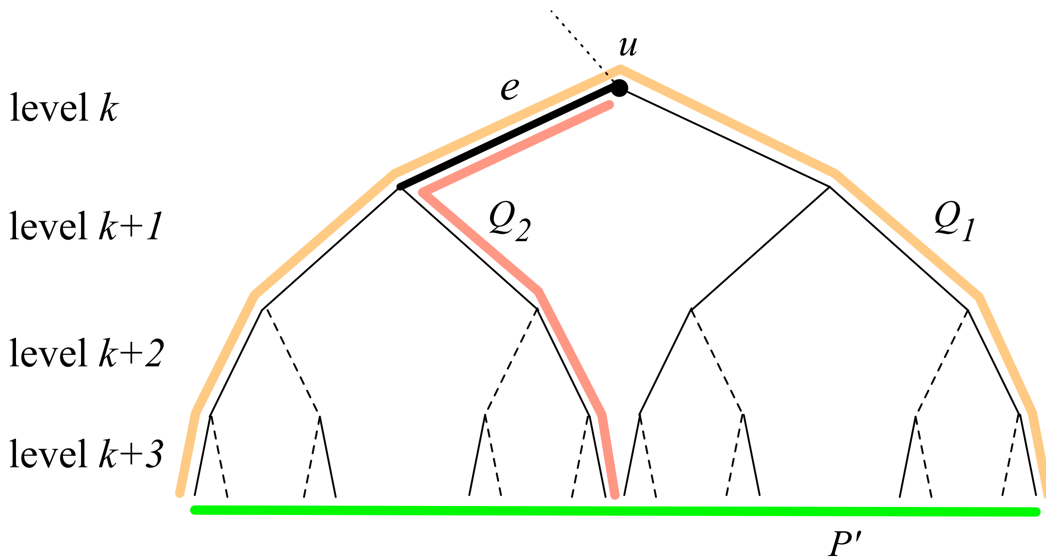


Figure 14: We bound the length of the replacement tree $\widehat{T}_u$ by the length of the path Q_1 through the tree from its leftmost leaf to its rightmost leaf, plus the length of the path Q_2 from the root to one of its leaves. We choose Q_2 to first traverse two zig-zag edges and subsequently not traverse any zig-zag edges. The dashed edges in the figure are zig-zag edges that are in neither Q_1 nor Q_2. The total length of these edges is a credit against the debit represented by the edge e that appears in both Q_1 and Q_2.

We address this difficulty by selecting the path Q_2 carefully and by selecting the level k carefully. As shown in Figure 14, we can select Q_2 so it shares only one edge e with Q_1. Moreover, in arguing that $\widehat{T}_u$ is not much longer than T''_u, we can use the fact that there are many edges that belong to T'_u but do *not* belong to $\widehat{T}_u$, including in particular the dashed edges in Figure 14.

For each level-k vertex u, we choose the path Q_2 to be the path starting at u that traverses the next two zig-zag edges and then continues to a leaf without taking any more zig-zag edges. For example, if, as in Figure 14, u is a right child of its parent, then Q_2 traverses u's left child edge, then goes right and continues going right until reaching a leaf.

The advantage of this choice of path is that, after the first two edges, Q_2 avoids all zig-zag edges. Note also that Q_1 also avoids all zig-zag edges except for the child zig-zag child edge of u. Let e denote this edge. Since e is the only edge common to Q_1 and Q_2, and none of the zig-zag edges at levels $k+2$ and above belong to either Q_1 or Q_2,

$$\begin{aligned}\text{length}(Q_1) + \text{length}(Q_2) + \text{length}(\text{zig-zag edges at levels } k+2, k+3, \ldots)\\ \leq \text{length}(T''_u) + \text{length}(e)\end{aligned}$$

where we include here only zig-zag edges belonging to T''_u.

We combine this inequality with Inequality 3, obtaining

$$\text{length}(\widehat{T}_u)+\text{length}(\text{zig-zag edges at levels } k+2, k+3, \ldots) \leq \text{length}(T''_u)+\text{length}(e) \tag{4}$$

Note that edge e is a level-k zig-zag edge. Now we sum 4 over all level-k vertices u, obtaining

$$\sum_u \text{length}(\widehat{T}_u) + L_{k+2} + L_{k+3} + \cdots \leq \sum_u \text{length}(T''_u) + L_k \tag{5}$$

We add the lengths of edges at levels less than k to both sides. These edges appear in both T'' and $\widehat{T}$, so we obtain

$$\text{length}(\widehat{T}) + L_{k+2} + L_{k+3} + \cdots \leq \text{length}(T'') + L_k \tag{6}$$

Combining this inequality with the following claim completes the proof of property (c).

Claim: *There exists $k \leq \log_\phi \epsilon^{-1}$ such that $L_k \leq \epsilon$ length$(T'') + L_{k+2} + L_{k+3} + \cdots$, where $|phi = \frac{1+\sqrt{5}}{2}$ is the golden ratio*

Let $k_0 = \lfloor \log_\phi \epsilon^{-1} \rfloor$. Define $F_{-2}, F_{-1}, F_0, F_1, F_2, \ldots, F_{k_0}$ by the recurrence

$$\begin{aligned} F_{k_0} &= 1 \\ F_{k_0-1} &= 1 \\ F_k &= F_{k+2} + F_{k+3} + F_{k+4} + \cdots + F_{k_0} \end{aligned}$$

The recurrence implies that $F_k = F_{k+1} + F_{k+2}$ for $-2 \leq k \leq k_0 - 2$. Therefore $F_k \geq \phi^{k_0-k-1}$, so in particular $F_{-2} \geq \phi^{k_0+1} > \epsilon^{-1}$.

Assume the claim is false. Then, for each integer $0 \leq k \leq k_0$, $L_k > \epsilon\,\text{length}(T'')F_k$, so

$$\begin{aligned} L_0 + L_1 + L_2 + \cdots + L_{k_0} &> \epsilon\,\text{length}(T'')(F_0 + F_1 + F_2 + \cdots + F_{k_0}) \\ &= \epsilon\,\text{length}(T'')(F_{-2}) \\ &> \epsilon\,\text{length}(T'')(\epsilon^{-1}), \end{aligned}$$

which is a contradiction. Thus the claim is true.

0.2.8 Span2

Again G is a planar embedded graph, P is an $1+\epsilon$-short path forming part of the boundary of G, and T is an r-rooted tree of G that intersects P only at leaves of T.

In this section, we give the proof of Lemma 0.1.4, which is reproduced here:

> There is a procedure Span2$(\cdot,\cdot,\cdot,\cdot)$ such that, for vertices x, y of K, Span2(P, K, x, y) returns a subgraph of $P \cup K$ of length at most $(1+2\epsilon+\epsilon^2)$length(T) that spans all the vertices of $\{x, y\} \cup (K \cap P)$ and has at most $2\epsilon^{-2.5}$ joining vertices with P, where c is a constant.

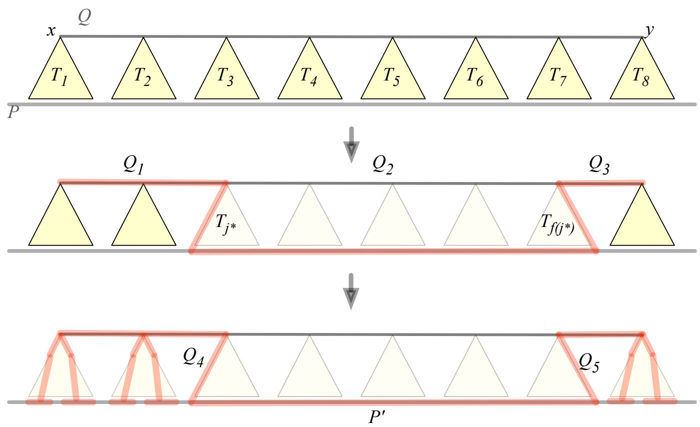


Figure 15: The original tree T is shown at the top. It consists of an x-to-y path Q and trees $T_1, \ldots, T_k$ rooted at vertices of Q. In the first step, the subpath Q_2 and the middle trees are replaced by the paths Q_4 and Q_5 and the subpath P' of P. In the second step, Span1 is applied to the non-middle trees.

Let Q be the unique x-to-y path in T. Removing the edges of Q from T breaks T into a forest consisting of trees rooted at vertices of Q with leaves on P. Let $r_1, \ldots, r_k$ be the roots in order along Q and let $T_1, \ldots, T_k$ be the trees.

If $k < 2\lceil\epsilon^{-1}\rceil$ then obtain a tree $\widehat{T}$ from T by applying Span1 to each tree T_i, replacing it with a tree $\widehat{T_i}$ that has at most $\epsilon{-}1.45$ joining vertices. It follows that $\widehat{T}$ has at most $2\epsilon^{-2.45}$ joining vertices.

Assume therefore that $k \geq 2\lceil\epsilon^{-1}\rceil$. For $j = 1, 2, \ldots, \lceil\epsilon^{-1}\rceil$, define $f(j) = k - \lceil\epsilon^{-1} + j$, and define $L_j = \text{length}(T_j) + \text{length}(T_{f(j)})$. Let $j^* = \text{minarg}\ _j L_j$. Then

$$L_{j^*} \leq \epsilon\,\text{length}(T_1 \cup T_2 \cup \cdots \cup T_k) \tag{7}$$

The transformations are illustrated in Figure 15. Write $Q = Q_1 \circ Q_2 \circ Q_3$ where $\text{start}(Q_2) = r_{j^*}$ and $\text{end}(Q_2) = r_{f(j^*)}$. Let Q_4 be the leftmost root-to-leaf path in T_{j^*} and let Q_5 be the rightmost root-to-leaf path in $T_{f(j^*)}$. Say a tree T_j is a *middle tree* if $j^* \leq j \leq f(j^*)$. As illustrated in Figure 15, we obtain $\widehat{T}$ from T as follows:

1. Remove Q_2 and the middle trees, and add Q_4, Q_5, and the end(Q_4)-to-end(Q_5) subpath P' of P.

2. Apply Span1 to each of the non-middle trees.

Since there are at most ϵ^{-1} non-middle trees, and each is replaced with a tree with at most $\epsilon^{-1.45}$ joining vertices, there are at most $\epsilon^{-2.45} + 2$ joining vertices. (The two come from Q_4 and Q_5.)

The increase in length due to the second step is at most $1 + \epsilon$ times the length of the non-middle trees. Since P is $1 + \epsilon$-short and $\text{rev}(Q_4) \circ Q_2 \circ Q_5$ is a start(P')-to-end(P') path,

$$\text{length}(P') \leq (1+\epsilon)\text{length}(Q_4) + \text{length}(Q_2) + \text{length}(Q_5) \tag{8}$$

Since Q_4 is part of T_{j^*} and Q_5 is part of $T_{f(j^*)}$,

$$\text{length}(Q_4) + \text{length}(Q_5) \leq L_{j^*} \tag{9}$$

The increase in length due to the first step is

$$\begin{aligned}
&\text{length}(P') + \text{length}(Q_4) + \text{length}(Q_5) - \text{length}(Q_2) - \text{length}(\text{middle trees}) \\
&\quad \leq \quad \text{length}(P') - \text{length}(Q_2) \\
&\quad \leq \quad (1+\epsilon)[\text{length}(Q_4) + \text{length}(Q_2) + \text{length}(Q_5)] - \text{length}(Q_2) \\
&\quad \leq \quad (1+\epsilon)[\text{length}(Q_4) + \text{length}(Q_5)] + \epsilon\,\text{length}(Q_2) \\
&\quad \leq \quad (1+\epsilon)\epsilon\,\text{length}(T_1 \cup \cdots \cup T_k) + \epsilon\,\text{length}(Q_2)
\end{aligned}$$

Hence the total increase is at most $(1+\epsilon)\epsilon + \epsilon$ times the length of T.