Appendix A

***Proof*. (Proof of Lemma 1)** *We prove this lemma by contradiction. Suppose there is a non-anchor edge $e \in E$ with trussness increasing from k_1 to k_2 after anchoring edge x, where $k_2 > k_1 + 1$. Let S be the k_2-truss after anchoring edge x. Thus we have $sup(e, S) \geq k_2 - 2$ for $e \in S$. Now if we delete the anchored edge x, we have $sup(e, S \backslash \{x\}) \geq k_2 - 3$ for $e \in S$ because deleting an edge can only break one triangle. Thus $S \backslash x \subseteq (k_2-1)$-truss. And $e \in S$ and $x \neq e$, so $e \in (k_2-1)$-truss and $k_1 \geq k_2 - 1$ which contradicts $k_2 > k_1 + 1$.*

***Proof*. (Proof of Lemma 2)** *We first discuss the cases of triangles to prove condition i), then based on i), we prove the condition ii)*

As Fig. 12 shows, we list all the triangles that contain the anchor x (The dotted edges). Before anchoring x, we have $t(x) = t$ and $l(x) = l$. And the pair (t', l') represents the trussness and layer of the edge. Now, consider situations $(a) - (d)$ and we take situation (a) for example. we have $t(u, w) = t(v, w)$, before anchoring (u, v), (u, v) is deleted before (u, w) and (v, w) thus when deletion process reaches edge (u, w) or (v, w), Δ_{uvw} no longer exists. However, after anchoring (u, v), edge (u, w) and (v, w) can get extra support from Δ_{uvw} potentially leading to an increase in their trussness. Thus once edge e gets extra $sup(e, S)$ where $S = A \cup E(G) \backslash \{e' | \{t(e') < t(e)\} || \{t(e') = t(e) \& l(e') < l(e)\}\}$, which could lead to an increase in its trussness. The same reasoning applies to situations $(b) - (d)$. In contrast, in situations $(e) - (l)$, edge (u, w) and (u, v) are deleted either before or in the same round as x. So they do not gain any additional support. Thus, these situations can be pruned. Thus if condition i) is satisfied, e_t may become a follower of x.

Now suppose that there is an edge e_s satisfy condition i), and e_s can increase their trussness. We begin by discussing the first triangle Δ_{uvw} that contains $e_s(u, v)$. Let O be the deletion order without anchor x. According to Lemma 1, $e_s(u, v)$ can at most increase its trussness by 1. For edges e which $t(e) > t(e_s)$ in Δ_{uvw}, they can follower the same deletion order O, because $t(e_s)$ can't exceed $t(e)$ after trussness increase. Then for edges with $t(e) < t(e_s)$. The deletion order also remains unchanged because they are deleted before e_s. The same reason applies to edges with $t(e) = t(e_s)$ and $l(e) < l(x)$. Thus the above edges which conflict situation ii) and iii) in Definition 7 can't increase its trussness. Detailed triangle types are shown similarly as Fig. 12. Note that here we regard $e_s(u, v)$ as possible trussness increased edges, not the anchor.

Now if we continue to examine edges that may cause a trussness increase subsequently, the same conclusion still holds. By considering these potentially trussness increasing edges as common edges and iteratively stitching successive triangles together, we can form an upward route that satisfies condition i), as well as ii) and iii) in Definition 7. Thus, the lemma holds.

***Proof*. (Proof of Lemma 3)** *We begin by establishing why the upper bound is set to $t(e) - 1$. According to the definition of k-truss, each edge e in k-truss has at least $k - 2$ triangles containing e. Thus if $t(e)$ increase to $t(e) + 1$ after anchoring, there must be at least $t(e) - 1$ containing e triangles in $T_{t(e)+1}(G_{\{x\}})$.*

Next, we conclude the proof by identifying and pruning triangles that cannot exist in $T_{t(e)+1}(G_{\{x\}})$. We categorize the neighbor-edges e' of e into three groups, i)$t(e') < t(e)$, ii)$t(e') = t(e)$ and iii)$t(e') > t(e)$. It is clear that edges in group i) are deleted before e and therefore cannot be present in $T_{t(e)+1}(G_{\{x\}})$. Consequently, any triangle containing edges from this group can be pruned.

For group ii), edges that remain at $t(e)$, have a negative impact on the trussness of edges that increase in trussness, as these edges are deleted prior to those with increased trussness. Such edges are classified as eliminated edges in Definition 8, allowing us to prune triangles containing them. For edges with $l(e') \geq l(e)$ and $l(e') \leq l(e)$ but are considered as survived edges", they are considered in Definition 8, unless they are subsequently marked as eliminated.

Finally, for group iii), edges with higher trussness are always deleted after $t(e)$, which is also considered in Definition 8. Thus the lemma holds.

***Proof*. (Proof of Lemma 4)** *We prove the lemma by pruning all tree nodes TN where $TN.I \notin sla(x)$. Let O be the deletion order of truss decomposition without anchoring edge x.*

Firstly, consider the tree node TN with $TN.K < t(x)$. we can follow the deletion order O because the support of edge $e \in TN.E$ remains the same. Therefore, anchoring edge x has no effect on those tree nodes. We can safely prune this condition.

Next, we consider tree node TN with $TN.K \geq t(x)$. During the construction of the tree, it is obvious that edges in the same $TN.E$ are triangle-connected in $(TN.K)$-truss component. Thus we can prune these k-truss components without including x and tree node built by such k-truss component. We only focus on the k-truss component including x. According to lemma 2, there must exist an upward route with the same trussness from x to followers. With the definition of $sla(x)$, the $sla(x)$ is the set of $TN.I$ with $e \in TN.E$. The upward route of x must be contained in $TN.E$. Thus, we can also prune the remaining tree nodes built by the k-truss component. The lemma holds.

***Proof*. (Proof of Lemma 5)** *Recall the tree building process, we categorized edges based on their trussness and triangle connectivity. If the tree node TN remains the same, it is obvious that $F[e][TN.I]$ remains unchanged. Thus we only need to identify tree nodes with structure change, as following cases: i) According to Lemma 4, $F(x)$ only comes from TN where $TN.I \in sla(x)$. After we anchor x, $F(x)$ can increase their trussness, which directly leads to the structure change of corresponding TN where $TN.I \in sla(x)$. ii) Then, due to the trussness increase of $F(e)$, we need to merge these edges into tree nodes with higher k, which also leads to the structure change of these tree nodes. iii) Finally, due to the support of the anchor being set to infinity, $t(x)$ also becomes infinity, such*

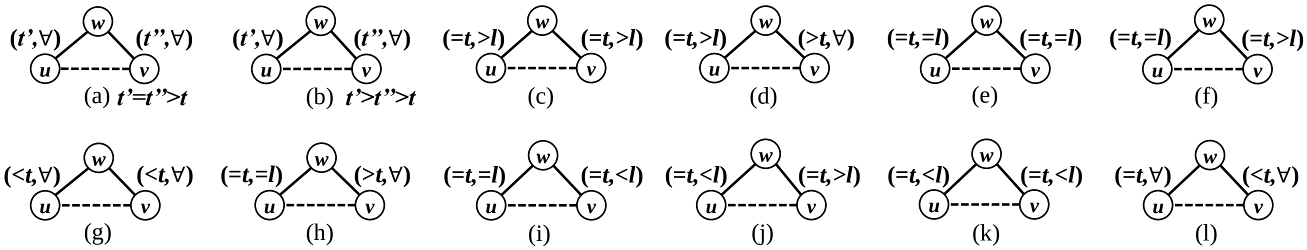

Fig. 12: Cases of triangles

that we need to exclude it from $TN.E$ where $x \in TN.E$. which also leads to the corresponding tree node structure change. In algorithm 5, we collect expired ids in case i) (lines 2-4), ii) (line 11) and iii) (line 1). And we delete these ids from $rn(\cdot)$ (lines 12-13), all cases are included. Thus the lemma holds.