

Spanning Trees in Graphs of High Minimum Degree with a Universal Vertex I: An Approximate Asymptotic Result

Bruce Reed* Maya Stein†

May 23, 2019

Abstract

In this paper and a companion paper, we prove that, if m is sufficiently large, every graph on $m+1$ vertices that has a universal vertex and minimum degree at least $\lfloor \frac{2m}{3} \rfloor$ contains each tree T with m edges as a subgraph. The present paper already contains an approximate asymptotic version of the result.

Our result confirms, for large m , an important special case of a recent conjecture by Havet, Reed, Stein, and Wood.

1 Introduction

A recurring topic in extremal graph theory is the use of degree conditions (such as minimum/average degree bounds) on a graph to prove that it contains certain subgraphs. One of the easiest classes of subgraphs for which this question is not yet properly understood are trees. This is the focus of the present paper.

Clearly, any graph of minimum degree exceeding $m - 1$ contains a copy of each tree with m edges: Just embed the root of the tree anywhere in the

*School of Computer Science McGill University. Research supported by NSERC.

†Department of Mathematical Engineering and Centro de Modelamiento Matemático, Universidad de Chile, UMI 2807 CNRS. Research supported by CONICYT + PIA/Apoyo a centros científicos y tecnológicos de excelencia con financiamiento Basal, Código AFB170001, and by Fondecyt Regular Grant 1183080.

host graph, and greedily continue, always embedding vertices whose parents have already been embedded. The bound on the minimum degree is sharp (see below).

Our paper is one of a large number which discuss possible strengthenings of the above observation by replacing the minimum degree condition with a different condition on the degrees of the host graph. One of these is the Loebel-Komlós-Sós conjecture from 1995 (see [EFLS95]), which replaces the minimum degree with the median degree. This conjecture has attracted a fair amount of attention over the last decades, and has been settled asymptotically [HKP^a, HKP^b, HKP^c, HKP^d]. More famously, Erdős and Sós conjectured in 1963 that every graph of average degree exceeding $m - 1$ contains each tree with m edges as a subgraph. This conjecture would be best possible, since no $(m - 1)$ -regular graph contains the star $K_{1,m}$ as a subgraph. Alternatively, consider a graph that consists of several disjoint copies of the complete graph K_m ; this graph has no connected $(m + 1)$ -vertex subgraph at all. Note that for these examples it does not matter whether we considered the average degree (as in the Erdős–Sós conjecture) or the minimum degree (as in the observation above).

The Erdős–Sós conjecture poses an extremely interesting question. It is trivial for stars, and it holds for paths by an old theorem of Erdős and Gallai [EG59]. It also holds when some additional assumptions on the host graph are made, see for instance [BD96, Hax01, SW97]. In the early 1990’s, Ajtai, Komlós, Simonovits and Szemerédi announced a proof of the Erdős–Sós conjecture for sufficiently large m .

It is well-known that every graph of average degree $> m$ has a subgraph of minimum degree $> \frac{m}{2}$. So, if it were true that every graph of minimum degree exceeding $\frac{m}{2}$ contained each tree on m edges, then the Erdős–Sós conjecture would immediately follow. Of course, the statement from the previous sentence is not true: It suffices to consider the examples given above. Still, for bounded degree spanning trees an approximate version of the statement does hold. Komlós, Sarközy and Szemerédi show in [KSS01] that every large enough $(m + 1)$ -vertex graph of minimum degree at least $(1 + \delta)\frac{m}{2}$ contains each tree with m edges whose maximum degree is bounded by $\frac{cn}{\log n}$, where the constant c depends on δ . Variations of the bounds and the size of the tree are given in [BPS18, CLNS10]. However, the result is essentially best possible in the sense that (even if the minimum degree of the host graph is raised) it does not hold for trees of significantly larger maximum degree [KSS01].

So, if we wish to find a condition that guarantees we can find *all* trees of

a given size as subgraphs, only bounding the minimum degree is not enough. Nevertheless, there can be at most one vertex of degree at least $\frac{m}{2}$ in any tree on $m + 1$ vertices, and so, we might not need many vertices of large degree in the host graph. Therefore, it seems natural to try to pose a condition on both the minimum and the maximum degree of the host graph.

The first conjecture of this type has been put forward recently by Havet, Reed, Stein, and Wood [HRSW16]. They believe that a maximum degree of at least m and a minimum degree of at least $\lfloor \frac{2m}{3} \rfloor$ is enough to embed all m -edge trees.

Conjecture 1.1 (Havet, Reed, Stein, and Wood [HRSW16]). *Let $m \in \mathbb{N}$. If a graph has maximum degree at least m and minimum degree at least $\lfloor \frac{2m}{3} \rfloor$ then it contains every tree with m edges as a subgraph.*

The conjecture holds if the minimum degree condition is replaced by $(1 - \gamma)m$, for a tiny but explicit¹ constant γ , and it also holds if the maximum degree condition is replaced by a large function² in m [HRSW16]. An approximate version of the conjecture holds for bounded degree trees and dense host graphs [BPS18].

As further evidence we shall prove, in this paper and its companion paper [RS19b], that Conjecture 1.1 holds for sufficiently large m , under the additional assumption that the graph has $m + 1$ vertices, i.e., when we are looking for a spanning tree. That is, building on the results from the present paper, we will show the following theorem in [RS19b].

Theorem 1.2. [RS19b] *There is an $m_0 \in \mathbb{N}$ such that for every $m \geq m_0$ every graph on $m + 1$ vertices which has minimum degree at least $\lfloor \frac{2m}{3} \rfloor$ and a universal vertex contains every tree T with m edges as a subgraph.*

Observe that Theorem 1.2 is easy if T has a vertex t that is adjacent to a set L of at least $\lceil \frac{m}{3} \rceil$ leaves. We root T at v , embed t in the universal vertex v^* of G , greedily embed $T - L$, and then embed L in neighbours of v^* . This is possible since v^* is universal.

It turns out that this approach can be extended if, for a small positive number δ , the tree T contains a vertex adjacent to at least δn leaves. Although the greedy argument no longer works, we will be able to prove a result, namely Lemma 1.3 below, which achieves the embedding of any tree T

¹Namely, $\gamma = 200^{-30}$.

²Namely, $f(m) = (m + 1)^{2m+6} + 1$.

as above. This lemma will be crucial for the proof of Theorem 1.2 in our companion paper [RS19b].

Lemma 1.3. *For every $\delta > 0$, there is an m_δ such that for any $m \geq m_\delta$ the following holds for every graph G on $m + 1$ vertices which has minimum degree at least $\lfloor \frac{2m}{3} \rfloor$ and a universal vertex.*

If T is a tree with m edges, and some vertex of T is adjacent to at least δm leaves, then G contains T .

Also, the results from the present paper alone imply an approximate asymptotic version of this theorem.

Theorem 1.4. *For every $\delta > 0$, there is an m_δ such that the following holds for every $m \geq m_\delta$ and every graph G on $m + 1$ vertices which has minimum degree at least $\lfloor \frac{2m}{3} \rfloor$ and a universal vertex.*

If T is a tree with at most $(1 - \delta)m$ edges, then G contains T .

Both Theorem 1.4 and Lemma 1.3 follow from Lemma 2.1, which is stated in Section 2, and whose proof occupies almost all the remainder of this paper. In the companion paper [RS19b], we will prove the full Theorem 1.2, building on Lemma 1.3 and another auxiliary result, namely Lemma 7.3, which is to be stated and proved in the last section of the present paper (Section 7).

Let us end the introduction with a very short overview of our methods of proof. A more detailed overview can be found in Section 3.

Given a tree T we wish to embed in the host graph G , we first cut T into a constant number of connecting vertices, and a large number of very small subtrees. Applying regularity to G , we can ensure that all those small trees that are not just leaves can be embedded into matching structures we find in the reduced graph of the regularised graph G . This is more complicated than in earlier work on tree embeddings using the regularity approach, as our assumptions are too weak to force one matching structure we can work with throughout the whole embedding. Instead, we have to employ ad-hoc matchings, plus some auxiliary structures, one for each of the connecting vertices. Finally, we have to deal with the leaves adjacent to connecting vertices. These are more difficult to embed than the other small trees, because an embedded vertex might only see two thirds of the graph, and there is no way to reach the remaining third of the graph in only one step. For this reason, we have to come up with a delicate strategy on where we place the connecting vertices, in order to ensure that at the very end of the embedding

process we will be in a position to embed all these leaves at once with a Hall-type argument.

2 The Proof of Theorem 1.4

The lemma behind the two results we stated in the introduction (Lemma 1.3 and Theorem 1.4) is the following.

Lemma 2.1. *For every $\delta > 0$, there is an m_δ such that for any $m \geq m_\delta$ and α with $\delta \leq \alpha \leq 1$ the following holds.*

Let G be an $(m + 1)$ -vertex graph of minimum degree at least $\lfloor \frac{2m}{3} \rfloor$, with $w \in V(G)$. Let T be a tree with at most $(1 - \alpha)m$ edges, with $t \in V(T)$. If no vertex of T is adjacent to more than αm leaves, then one can embed T in G , mapping t to w .

Let us now show how Lemma 2.1 implies the results from the introduction. Here is the proof of the lemma that we will need in the companion paper [RS19b].

Proof of Lemma 1.3. Let m_δ be given by Lemma 2.1 for input δ . Given G and T as in Lemma 1.3, we let t be a vertex of T having the maximum number of leaf neighbours. We let L be the set of its leaf neighbours and set $\alpha := \frac{|L|}{m}$. By assumption, $\delta \leq \alpha \leq 1$, so we may apply Lemma 2.1 to obtain an embedding of $T - L$ in G with t embedded in the universal vertex of G . We can then arbitrarily embed the vertices of L into the remaining vertices of G . \square

Now comes the proof of the approximate result.

Proof of Theorem 1.4. Let m_δ be the maximum of the numbers m_δ given by Lemma 2.1 and by Lemma 1.3 for input δ . Given G and T as in the theorem, consider a vertex of T with the maximum number of leaf neighbours, say these are βm leaf neighbours. If $\beta \leq \delta$, we are done by Lemma 2.1. If $\beta > \delta$, we are done by Lemma 1.3. \square

3 A Sketch of the Proof of Lemma 2.1

The purpose of this section is to give some more detailed insight into the proof of Lemma 2.1, going a little more below the surface than in the Introduction.

We remark that for the understanding of the rest of the paper, it is not necessary to read this section (but we hope it will be helpful).

The number m_δ will be chosen in dependence of the output of the regularity lemma for some constant depending on δ . Given now the approximation constant α , the tree T and the host graph G , we prepare each of T and G separately for the embedding.

Similar as in earlier tree embedding proofs [AKS95, HKP⁺d, PS12], we cut T into a set W of *seeds* (connecting vertices), such that W has constant size, and a large set \mathcal{T} of very small subtrees. The trees in \mathcal{T} are only connected through W , and they each have size $< \beta m$, where β is a small constant (smaller than all other constants in this paper).

Differently from earlier approaches to tree embeddings, we now categorise the small trees contained in \mathcal{T} : They fall into three categories: trees consisting of a leaf of T , trees that are smaller than a (huge) constant, and trees that are larger than this constant. We name the categories L , F_1 and F_2 . The last category is further subdivided into two sets, F'_2 and $F_2 \setminus F'_2$, according to whether the small tree is adjacent to one or more of the connecting vertices. Each seed from W may have trees of any (or possibly all) of these categories hanging from it (and there may also be seeds hanging from it). The details of this cut-up of T is explained in Section 5.1.

Next, in Section 5.2, we order and group the seeds obtained from this decomposition. Our strategy of ordering the seeds takes into account their position in a natural embedding order, but also the number of leaves hanging from them. We will come back to this point at a later stage during this outline, and will then explain the why and how of the ordering.

Independently, in Section 6.1.1, we regularize the host graph G , with parameter ε , such that $\beta \ll \varepsilon \ll \alpha$. (For an introduction to regularity, see Section 4.3.) Furthermore, we partition each of its clusters C arbitrarily into subsets $C_{\tilde{W}}, C_Z, C_{F_1}, C_{F_2}, C_{\tilde{V}}$ of appropriate sizes into which we aim to embed the different parts of the tree, namely, W , L , F_1 , and F_2 , while the last subset, $C_{\tilde{V}}$, is reserved for neighbours of seeds in trees of F_2 . The set of these neighbours will be denoted by \tilde{V} .

We fix a matching M_{F_2} in the reduced graph R_G . This matching will be used when we embed the trees from F_2 . More precisely, we will embed each tree $\bar{T} \in F_2 \cup F'_2$ into $C_{F_2} \cup D_{F_2}$ for a suitable (i.e. sufficiently unoccupied) edge $CD \in M_{F_2}$, except for the root $r_{\bar{T}}$ of \bar{T} . The root $r_{\bar{T}}$ will go to one of the subsets $C'_{\tilde{V}}$, for a suitable cluster C' that connects CD with the cluster

containing the seed adjacent to \bar{T} . In case $\bar{T} \in F'_2$, which means that \bar{T} contains a second vertex \tilde{v} from \tilde{V} , we embed \tilde{v} into one of the subsets $C''_{\tilde{v}}$, for a suitable cluster C'' . Throughout the embedding process, we will keep each of the edges of M_{F_2} as balanced as possible. That is, the sets of used vertices in the corresponding slices C_{F_2} or D_{F_2} on either side of such an edge never differ by more than βm .

Observe that since we do not have enough space in the slices $C_{\tilde{v}}$ for all roots of trees in F_1 (because we have no control over the number of trees in F_1), we need to proceed differently with the small trees from F_1 . For embedding these trees, we use a family of matchings M , one for each embedded seed s . Since these matchings M are possibly different for each s , we now will have to keep the set of *all* slices C_{F_1} balanced. This is not easy but possible since the trees from F_1 have constant size, and we choose M so that it intersects the neighbourhood of s in a nice way.

There are some problems with balancing the edges of M , since clearly, s might fail to see each side of each edge from M , which makes it difficult to balance those edges of M that are not completely contained in the neighbourhood of s in R_G . To overcome this problem, we employ two auxiliary matchings which we combine with M to obtain a partition of almost all of $V(R_G)$ with short paths. We call these structures *good path partitions* and, together with the matchings M , they will be defined and proved to exist in Subsection 4.2.

The actual embedding of the tree will be performed as follows. In Sections 6.2 and 6.3, we go through the seeds in a connected way, and embed each seed s together with all the trees from $F_1 \cup F_2$ at s in the corresponding slices in the way we discussed above. We leave out any leaves from L , as we will deal with them in the final phase of the embedding.

Whenever we have embedded one seed s and its trees, we proceed to the next seed and its trees. Because of the way we embed in the slices, and the way we chose our matchings, all of this will go through just fine, and we will always have enough space to embed. However, if we do not take care where exactly we embed the seeds, we may run into problems in the final phase when we want to embed the leaves. For instance, it might happen that all seeds that are adjacent to vertices in L have been embedded into vertices having the same neighbourhood in R_G . As this neighbourhood might be only two thirds of the vertices of R_G , the leaves might not fit.

For this reason, we take some extra care when choosing the target clusters and the actual images for the seeds (this happens in Section 5.2). As already

shortly mentioned above, we order the seeds into a system of groups according to the number of leaves hanging from them. Then we reorder this order a bit, according to the order the seeds appear in our planned embedding order. Also, each seed s will be assigned a *relevant* set X_s of seeds that come before it. In the actual embedding, in Subsection 6.2, we choose the image $\varphi(s)$ of a given seed s in a way that $\varphi(s)$ has many neighbours outside the union of the neighbourhoods of $\varphi(X_s)$. (We remark that it is crucial here that no vertex of T is adjacent to more than αm leaves.) This precaution will ensure that for each subset of seeds, their images have enough neighbours in $Z := \bigcup C_Z$. Therefore, we will be able to embed all the leaves in L at once by using Hall's theorem. The whole procedure will be explained in detail in Subsection 6.4.

The last section of this paper, Section 7, is devoted to the statement and proof of an auxiliary result, Lemma 7.3, that deals with a similar situation as the one treated by Lemma 2.1. We will need Lemma 7.3 in our companion paper [RS19b] (in addition to Lemma 2.1). The main difference to the situation here is that there, a small part of the tree is already embedded (and thus possibly blocking valuable neighbourhoods), but, on the positive side, throughout [RS19b], we will be able to assume that no seed is adjacent to many leaves, and so we can assume this as well in Lemma 7.3.

4 Preliminaries

4.1 An edge-double-counting lemma

We will need the following easy lemma.

Lemma 4.1. *Let G be a graph on n vertices, let $0 < \psi < \frac{1}{3}$, and let $S \subseteq V(G)$ be such that each vertex in S has degree at least $(\frac{2}{3} - \psi)n$. Then there are at least $(\frac{1}{3} + \frac{\sqrt{\psi}}{10})n$ vertices in G that each see at least $(\frac{1}{2} - \sqrt{\psi})|S|$ vertices of S .*

Proof. We let $A \subseteq V(G)$ denote the set of all vertices that see at least $(\frac{1}{2} - \sqrt{\psi})|S|$ vertices of S . Writing $e(S, V(G))$ for the number of all edges

touching S , where edges inside S are counted twice, we calculate that

$$\begin{aligned} \left(\frac{2}{3} - \psi\right)n \cdot |S| &\leq e(S, V(G)) \\ &\leq |V(G) \setminus A| \cdot \left(\frac{1}{2} - \sqrt{\psi}\right)|S| + |A| \cdot |S| \\ &\leq n \cdot \left(\frac{1}{2} - \sqrt{\psi}\right)|S| + |A| \cdot \left(\frac{1}{2} + \sqrt{\psi}\right)|S|, \end{aligned}$$

and conclude that

$$|A| \geq \frac{\frac{1}{6} + \frac{\sqrt{\psi}}{2}}{\frac{1}{2} + \sqrt{\psi}} \cdot n \geq \left(\frac{1}{3} + \frac{\sqrt{\psi}}{10}\right) \cdot n,$$

as desired. \square

4.2 Matchings and good path partitions

The purpose of this subsection is to find some matchings in a graph H (which will later be the reduced graph R_G of our host graph G , see Section 4.3 for a definition of the reduced graph). Actually we will combine some of the matchings we find to specific covers of H with short paths. These structures will be used for the embedding of T in the proof of Lemma 2.1, more specifically in Subsection 6.3. The important result of this section is Lemma 4.3, which provides us with the desired structures.

We need a quick definition before we start. For any graph H , and any $N \subseteq V(H)$, an N -good matching is one whose edges each have at most one vertex outside N .

We will start by proving the following lemma.

Lemma 4.2. *Let $0 < \xi < \frac{1}{20}$ and let H be a p -vertex graph of minimum degree at least $(\frac{2}{3} - \xi)p$. Let $N \subseteq V(H)$ be such that $|N| = \lceil (\frac{2}{3} - 2\xi)p \rceil$. Then $H - N$ contains a set Y of size at most $\lfloor 5\xi p \rfloor + 1$ such that $H - Y$ has an N -good perfect matching.*

Proof. First, note that we can greedily match all but a set X of at most $\lfloor 5\xi p \rfloor$ vertices from $V(H) \setminus N$ to N , simply because of the condition on the minimum degree. Now, take any maximal N -good matching M in the graph H that covers all vertices of $V(H) \setminus (N \cup X)$. We would like to see that M covers all but at most vertex of $H - X$, so for contradiction assume

that $N \setminus V(M)$ contains at least two vertices. By the maximality of M , no vertex $D \in N \setminus V(M)$ is adjacent to any of the other uncovered vertices in N , and no two vertices $D, D' \in N \setminus V(M)$ can be adjacent to different endpoints of an edge in M . So, for each edge $EF \in M$, we know that either one of the endvertices, say E , sees no vertex in $N \setminus V(M)$, or E and F each see only one vertex in $N \setminus V(M)$ (and that is the same vertex). This means that at least one of the vertices in $N \setminus V(M)$ sees at most half of the vertices in $V(M)$, and thus less than half of the vertices in $H - X$, a contradiction to our condition on the minimum degree. \square

Before we state the second lemma of this section, we need another definition.

An *N -out-good path partition* of a graph H , with $N \subseteq V(H)$, is a set \mathcal{P} of disjoint paths, together covering all the vertices of H , such that for each $P \in \mathcal{P}$ one of the following holds:

- $P = AB$, with $A, B \in N$;
- $P = ABCD$, with $B, C \in N$; or
- $P = ABCDEF$, with $B, C, D, E \in N$.

(Note that if P has four vertices, then there is no restriction on the whereabouts of A and D , and similar for six-vertex paths P .)

An *N -in-good path partition* of a graph H , with $N \subseteq V(H)$, is a set \mathcal{P} of disjoint paths, together covering all the vertices of H , such that for each $P \in \mathcal{P}$ one of the following holds:

- $P = AB$, with $A, B \in N$;
- $P = ABCD$, with $A, D \in N$; or
- $P = ABCDEF$, with $A, C, D, F \in N$.

Now we are ready to state the main result of this section. Note that the first item is a direct consequence of the previous lemma, and all structures exist independently of each other.

Lemma 4.3. *Let $0 < \xi < \frac{1}{20}$, and let H be a p -vertex graph of minimum degree at least $(\frac{2}{3} - \xi)p$. Let $N \subseteq V(H)$ be any set with $|N| = \lceil (\frac{2}{3} - 2\xi)p \rceil$. Then H contains a set X of at most $\lfloor 15\xi p \rfloor + 1$ vertices, such that*

- $H - X$ has an $(N \setminus X)$ -good perfect matching;
- $H - X$ has an $(N \setminus X)$ -in-good path partition; and
- $H - X$ has an $(N \setminus X)$ -out-good path partition.

Proof. Lemma 4.2 provides us with a set Y and an N -good perfect matching M of $H - Y$. Note that M would be as desired for the first item, if X was chosen as Y . Now, set $A := V(M) \setminus (N \cup Y)$, and let B be the set of all vertices from H that are matched by M to a vertex from A . Since M is N -good, we know that $B \subseteq N$.

We take a maximal matching \tilde{M}^A inside $H[A]$. Call \tilde{A} the set of vertices in A not covered by \tilde{M}^A . We augment \tilde{M}^A to a matching M^A by matching as many vertices of \tilde{A} as possible to a set A' of vertices in N . Because of the minimum degree condition of the lemma, and since \tilde{A} is an independent set, we can ensure that for the set $Z \subseteq \tilde{A}$ of vertices not covered by M^A we have $|Z| \leq 5\xi p$. Moreover, we can choose A' such that for each $D \in A'$, the other endvertex of the edge of M containing D also lies in N . Let Z' denote the set Z is matched to in M .

Now we define a second auxiliary matching M^B in a very similar way. Matching M^B consists of edges with both ends in $B \setminus Z'$, and a matching of almost all the remaining vertices of $B \setminus Z'$ to a set B' of vertices from N , such that for each $D \in B'$, the other endvertex of the edge of M containing D also lies in N . We can ensure that at most $\lceil 5\xi p \rceil$ vertices of $B \setminus Z'$ are not covered by M^B . Let X' denote the union of the set of these vertices and their partners in M , and set $X := X' \cup Y \cup Z \cup Z'$.

Then, discarding any edge that touches X from M , M^A and M^B , we find that the union of M and M^A gives an $(N \setminus X)$ -in-good path partition of $H - X$. Also, the union of M and M^B gives the desired N -out-good path partition. (Note that the four-vertex paths from these partitions have either one or two vertices outside $N \setminus X$.) Finally, M is an $(N \setminus X)$ -good matching of $H - X$. \square

4.3 Regularity

We need to quickly discuss Szemerédi's regularity lemma and a couple of other preliminaries regarding regularity. Readers familiar with this topic are invited to skip this subsection.

The *density* of a pair (A, B) of disjoint subsets $A, B \subseteq V(G)$ is $d(A, B) = \frac{|E(A, B)|}{|A| \cdot |B|}$. A pair (A, B) of disjoint subsets $A, B \subseteq V(G)$ is called ε -*regular* if

$$|d(A, B) - d(A', B')| < \varepsilon$$

for all $A' \subseteq A$, $B' \subseteq B$ with $|A'| \geq \varepsilon|A|$, $|B'| \geq \varepsilon|B|$. It is well known that regular pairs behave, in many ways, like random bipartite graphs with the same edge density.

If (A, B) is an ε -regular pair, then we call a subset A' of A ε -*significant* (or simply significant, if ε is clear from the context) if $|A'| \geq \varepsilon|A|$. We call a vertex from A ε -*typical* (or simply typical, if ε is clear from the context) with respect to a set $B' \subseteq B$ if it has degree at least $(1 - \varepsilon)d(A, B)|B'|$ to B' .

The following well known and easy-to-prove facts (see for instance [KSS02]) state that in a regular pair almost every vertex is typical to any given significant set, and also that regularity is inherited by subpairs. More precisely, if (A, B) is an ε -regular pair with density d , then

- for any ε -significant $B' \subseteq B$, all but at most $\varepsilon|A|$ vertices from A are ε -typical to B' ; and
- for each $\delta \geq 0$, and for any subsets $A' \subseteq A$, $B' \subseteq B$, with $|A'| \geq \delta|A|$ and $|B'| \geq \delta|B|$, the pair (A', B') is $\frac{2\varepsilon}{\delta}$ -regular with density between $d - \varepsilon$ and $d + \varepsilon$.

Szemerédi's regularity lemma states that every large enough graph has a partition of its vertex set into a bounded number of parts, of almost equal sizes, such that almost all pairs of partition sets are ε -regular.

Lemma 4.4 (Szemerédi's regularity lemma). *For every $\varepsilon > 0$ and $M_0 \in \mathbb{N}$ there are $M_1, n_0 \in \mathbb{N}$ such that for all $n \geq n_0$ the following holds.*

Every n -vertex graph G has a partition $V_0 \cup V_1 \cup \dots \cup V_p$ of V into $p + 1$ partition classes (or clusters) such that

- (a) $M_0 \leq p \leq M_1$;
- (b) $|V_1| = |V_2| = \dots = |V_p|$ and $|V_0| < \varepsilon n$;
- (c) *apart from at most $\varepsilon \binom{p}{2}$ exceptional pairs, the pairs (V_i, V_j) are ε -regular, for $i, j > 0$ with $i \neq j$.*

As usual, we define the *reduced graph* R_G corresponding to this decomposition of G as follows. The vertices of R_G are all clusters V_i ($i = 1, \dots, p$), and R_G has an edge between V_i and V_j if the pair (V_i, V_j) is ε -regular, and has density at least $10\sqrt{\varepsilon}$.

By standard calculations (see for instance [KSS02]), and assuming we take $M_0 \geq \lceil \frac{1}{\varepsilon} \rceil$, it follows that

$$\delta_w(R_G) \geq (1 - 12\sqrt{\varepsilon}) \cdot \frac{p}{|V(G)|} \cdot \delta(G), \quad (1)$$

where $\delta_w(R_G)$ is the weighted minimum degree. (That is, the densities of the pairs of clusters provide weights on the edges of R_G , and the weighted degree of a vertex is the sum of the corresponding edge-weights. The weighted minimum degree is the minimum of these degrees. Observe that $\delta_w(R_G) \leq \delta(R_G)$ since weights do not exceed 1.)

Almost all vertices of any cluster $C \in V(R_G)$ are typical to almost all significant sets, in the following sense. If \mathcal{Y} is a set of significant subsets of clusters in $V(R_G)$, then

$$\begin{aligned} &\text{all but at most } \sqrt{\varepsilon}s \text{ vertices } v \in C \text{ are typical with respect to} \\ &\text{all but at most } \sqrt{\varepsilon}|\mathcal{Y}| \text{ clusters in } \mathcal{Y}. \end{aligned} \quad (2)$$

To see this well-known observation, assume that the set $C' \subseteq C$ of vertices not satisfying (2) is larger than $\sqrt{\varepsilon}s$. Then

$$\begin{aligned} \sum_{Y \in \mathcal{Y}} |\{v \in C : v \text{ is not typical to } Y\}| &\geq \sum_{v \in C'} |\{Y \in \mathcal{Y} : v \text{ is not typical to } Y\}| \\ &\geq |C'| \sqrt{\varepsilon} |\mathcal{Y}| \\ &> \varepsilon \frac{|V(G)|}{p} |\mathcal{Y}|. \end{aligned}$$

Thus there is a $Y \in \mathcal{Y}$ such that more than $\varepsilon|C|$ vertices in C are not typical to Y , a contradiction.

Regularity will help us when embedding small trees into a pair of adjacent clusters of R_G .

Lemma 4.5. *Let CD be an edge of R_G , and let $U \subseteq G$ with $|C \setminus U|, |D \setminus U| \geq \sqrt{\varepsilon}|C|$. Let \bar{T} be a tree of size $\leq \varepsilon|C|$ with root $r_{\bar{T}}$. Then \bar{T} can be embedded into G , with $\bar{T} - r_{\bar{T}}$ going to $(C \cup D) \setminus U$, and with*

$r_{\bar{T}}$ going to any prescribed set of $\geq 2\varepsilon|C|$ vertices of C , or to any prescribed set of $\geq 2\varepsilon|C|$ vertices of C' , where C' is any other cluster of R_G that is adjacent to D .

Proof. We construct the embedding \bar{T} levelwise, starting with the root, which is embedded into a typical vertex of $(C \cup D) \setminus U$. At each step i we ensure that all vertices of level i are embedded into vertices of $C \setminus U$ (or $D \setminus U$) that are typical with respect to the unoccupied vertices of $D \setminus U$ (or $C \setminus U$). This is possible, because at each step i , and for each vertex v of level i , the degree of a typical vertex into the unoccupied vertices on the other side is at least $4\varepsilon|C|$, and there are at most $\varepsilon|C|$ nontypical vertices and at most $|\bar{T}| \leq \beta|C|$ already occupied vertices. \square

5 Preparing the tree

5.1 Cutting a tree

In this section, we will show how any tree T can be cut up into small subtrees and few connecting vertices. The idea is that later, we can use regular pairs to embed many tiny trees.

We will make use of a procedure which in a very similar shape has already appeared in [AKS95, HKP⁺d, PS12], although there, no distinctions between the trees from L , F_1 , F_2 , were made. The resulting cut-up is given in the following statement.

Lemma 5.1. *For any $m \in \mathbb{N}$, for any tree T on $m + 1$ vertices, and for any $\beta > 0$, there is a set $W \subseteq V(T)$, and a partition $\mathcal{T} = L \cup F_1 \cup F_2$ of the family \mathcal{T} of components of $T - W$, distinguishing a subset $F'_2 \subseteq F_2$, such that*

- (a) $r \in W$;
- (b) $|W| \leq \frac{2}{\beta^2}$;
- (c) $|V(\bar{T})| = 1$ for every tree $\bar{T} \in L$;
- (d) $1 < |V(\bar{T})| \leq \frac{1}{\beta}$ for every tree $\bar{T} \in F_1$;
- (e) $\frac{1}{\beta} < |V(\bar{T})| \leq \beta m$ for every tree $\bar{T} \in F_2$;
- (f) each $\bar{T} \in L \cup F_1 \cup (F_2 \setminus F'_2)$ has exactly one neighbour in W ;

(g) each $\bar{T} \in F'_2$ has exactly two neighbours in W ; and

(h) $|\tilde{V}| < 2\beta m$, where \tilde{V} is the set of all neighbours of vertices of W in $\bigcup F'_2$.

The vertices in W will also be called the seeds of T .

Proof. In a sequence of at most $\frac{1}{\beta}$ steps i , we define vertices w_i and trees T_i as follows. Set $T_0 := T$. Now, for each $i > 0$, let $w_i \in V(T_{i-1})$ be a vertex at maximal distance from r (the root of T) such that the components of $T_i - w_i$ that do not contain r each have size at most βm . Delete w_i and all of these components from T_{i+1} to obtain T_i . We stop when we reach r , which will be the last vertex w_i to be defined.

Let W_0 be the union of all w_i , and let \mathcal{T}_0 be the family of the trees in $T - W_0$. These two sets already fulfill items (a) and (b). (To see (b), note that at each step i , we cut off βm vertices. Hence we actually have that $|W_0| \leq \frac{1}{\beta}$.)

In order to obtain sets W , \mathcal{T} that also fulfill items (f) and (g), we add some vertices to W_0 as follows. For each $\bar{T} \in \bigcup \mathcal{T}_0$ that has $\ell > 2$ neighbours v_1, v_2, \dots, v_ℓ in W_0 , we add to W_0 a set of at most $\ell - 1$ vertices w'_j from $V(\bar{T})$ that separate all v_i 's from each other. Note that these are at most $\frac{1}{\beta}$ vertices in total (counting over all affected \bar{T}), since each of the newly added vertices w'_j can be associated to one of the 'old' vertices v_j from W_0 such that w'_j lies between v_j and r .

So, letting \mathcal{T}_1 be the family of the trees in $T - W_1$, the new sets W_1 , \mathcal{T}_1 still fulfill (a) and (b) (actually, we have that $|W_1| \leq \frac{2}{\beta}$). They furthermore have the property that each of the trees in $\bigcup \mathcal{T}_1$ has at most two neighbours in W_1 .

We modify our sets once more to ensure that only the large trees can have two seed neighbours. We proceed as follows. For each $\bar{T} \in \bigcup \mathcal{T}_1$ that has at most $\frac{1}{\beta}$ vertices and is adjacent to two seeds $w_1, w_2 \in W_1$, we add to W_1 all vertices on the path that connects the two seeds w_1, w_2 . In total, these are at most $\frac{1}{\beta} \cdot |W_1| \leq \frac{2}{\beta^2}$ vertices. Call the new set of seeds W .

Defining \mathcal{T} as the family of the trees in $T - W$, and adequately dividing the family \mathcal{T} into three families L, F_1, F_2 , and letting F'_2 be the appropriate subset of F_2 , we thus obtained sets that fulfill all properties of the claim. (Note that (h) follows directly from (e), (g) and the observation that $|\tilde{V}| \leq 2|F'_2| < \frac{2|T-r|}{\frac{1}{\beta}} = 2\beta m$.) \square

5.2 Ordering the seeds of a tree

In order to be able to choose well the clusters of $V(R_G)$ into which we will embed the seeds other than r later on, let us now define a convenient ordering of the seeds of a tree T with a cut-up as in Section 5.1. Together with this ordering we will define a set of relevant seeds X_s for each seed s of the tree, and ensure that the seeds in X_s come before s in the ordering.

The importance of this ordering and the sets X_s is that later, when we embed the parts of the cut-up tree in G , it will turn out that the small trees that are most difficult to embed are those from L , this is, the leaves of T that hang off seeds. An embedded seed has only degree $\frac{2}{3}m$ in G , of which a large part might already be used, so if we do not take sufficient care, we will get stuck when embedding the leaves. For this reason we have to choose very well into which clusters the seeds go, and the sets X_s will help us with this.

The reader might wish to skip the remainder of this unfortunately rather technical section at a first reading, because everything we do here is only necessary for the embedding of L . Even the embedding of L can be followed with only a vague understanding of the definitions of the present section if the reader takes the ‘Degrees of the embedded seeds into Z ’ as stated in Subsection 6.2.4 for granted.

5.2.1 Grouping and ordering

Let us start with the ordering. Assume we are given a tree T which has been treated by Lemma 5.1 for some $\beta > 0$, let W denote the set of seeds we obtained. Throughout the rest of this section we will assume that

$$|W| = 47 \cdot 2^{j^*}, \text{ where } j^* = \lceil \log \frac{2}{\beta^2} \rceil. \quad (3)$$

(This can be assumed by adding some artificial seeds to the tree T . We will explicitly discuss why this can be done in Subsection 6.1.2.)

We will order the seeds in two different ways, before we get to the third and final order. The first order is determined by the number of leaves hanging from each seed, the second order is determined by the position of the seeds in the tree T , and the third order is a mixture of both. We explain the orderings in detail in the following.

We start by ordering the seeds in a way that the number of leaf children of the seeds is non-increasing, and we call this the *size order* σ on the seeds.

For each $j = 0, \dots, j^*$, we partition the set of all seeds into 2^{j^*-j} consecutive groups of size $2^j \cdot 47$, under the size order σ . We call these the *large groups*. Clearly, each large group of size exceeding 47 is the union of two large groups half its size.

We break up each group B of size 47 into twelve consecutive groups (consecutive under σ) of the following sizes:

$$4, \mathbf{4}, 4, 4, 5, 4, \mathbf{4}, 4, 4, 5, 4, 1. \quad (4)$$

We call these the *small groups*. (So the small subgroup of size 1 of B consists of the very last seed of B in the size order σ .) We say the second and sixth group of size four are of type 1 (they are marked in boldface in (4)). The remaining groups of size four (i.e. the first, third, fourth, fifth, seventh, eighth and ninth group of size four) will be called type 2.

It would be difficult to embed the seeds in the size order σ , as this enumeration might not be suitable for embedding the tree in a connected way. For this reason, we employ a second order τ , which we call the *transversal order*, obtained by performing a preorder transversal on T , starting with the root r , and then restricting this order to W . (The transversal order is the actual order the seeds will be embedded in.)

The third order, which we call the *rearranged order* ρ , is obtained by reordering the order σ . First, we reorder the seeds in each small group so that each small group is ordered according to τ . Next, for every large group B of size 47, we reorder all its subgroups so that their first seeds form an increasing sequence in the transversal order τ . Finally, for every large group B of size > 47 (in successive steps according to the group size), we reorder the two subgroups within B so that the first subgroup contains the first seed in B under the transversal order τ (i.e. we reorder them such that the first seed of B under τ becomes the first seed of B). This finishes the definition of the rearranged order ρ .

We note that ρ maintains the structure given by breaking down the set of seeds into large and small groups. That is, if we partition the set of all seeds into 2^{j^*-j} consecutive groups under ρ of sizes $2^j \cdot 47$, we obtain the same groups as above for σ . Further, each group of size 47 breaks down into twelve small groups as above, although these are no longer ordered as in the sequence from (4). We note that each of the small and large groups under ρ remains a consecutive set of elements if viewed under the size order σ .

We will embed according to τ but momentarily work with ρ . We write $s <_{\rho} s'$ to denote that s comes before s' in order ρ (and similar for τ).

5.2.2 Sequences

In this subsection, we will follow the rearranged order ρ . We define for each large group B two sequences

$$(x_i^B)_{i=1\dots,j+6} \text{ and } (y_i^B)_{i=1\dots,j+7},$$

where j is such that $|B| = 2^j \cdot 47$, and vertices $x_i^B, y_i^B \in B$ are as specified in what follows.

We construct our sequences inductively. For $j = 0$, we have to deal with all groups of size 47. For each such group B , we take $x_1^B = y_1^B$ as the first seed of the group. The first seed of the second, third, fourth, fifth and sixth small subgroup of B is chosen as $x_2^B, x_3^B, x_4^B, x_5^B, x_6^B$, respectively. The first seed of the seventh, eighth, ninth, tenth, eleventh and twelfth small subgroup of B is chosen as $y_2^B, y_3^B, y_4^B, y_5^B, y_6^B, y_7^B$, respectively. (We always work under ρ , both when talking about the ‘first seed of a group’ and when talking about the ‘ i th subgroup’.)

For $j \geq 1$, we have to deal with all large groups of size $2^j \cdot 47$. For each such group B , do the following. By construction B is made up of two large subgroups of size $47 \cdot 2^{j-1}$, say these are B' and B'' (in this order, under ρ). We let

$$x_i^B := y_i^{B'} \text{ for all } 1 \leq i \leq j + 6,$$

and we set

$$y_1^B := x_1^B = y_1^{B'}, \text{ and } y_i^B := y_{i-1}^{B''} \text{ for all } 2 \leq i \leq j + 7.$$

This finishes the definition of the sequences. We remark that we will only use the sequences (x_i) in what follows (the sequences (y_i) were only used to make the definition of (x_i) more convenient).

Observe that for all blocks B , and for all $i < j$, we have that $x_i^B <_\tau x_j^B$.

5.2.3 Relevant seeds

In order to be later able to choose well the clusters we embed the seeds into (which in turn will enable us to embed the leaves at an even later stage), we need to define, for each seed s , a set X_s of relevant seeds for s , as follows.

Definition 5.2 (Relevant seeds for s).

Let s be a seed of T , and let B be the small group s belongs to.

(a) If B is a group of four of type 2, and s is the last seed of B , then we set

$$X_s := \{x : x \text{ is the third seed in } B \text{ (under } \rho)\}.$$

(b) If s is not the first seed of B , and, in case B is a group of four of type 2, s is not its last seed, then we set

$$X_s := \{x : x \in B, x <_\rho s\}.$$

(c) If s is the first seed of B , then we set

$$X_s := \{x : \exists \tilde{B}, i, i' \text{ such that } i' < i, s = x_i^{\tilde{B}} \text{ and } x = x_{i'}^{\tilde{B}}\}.$$

Observe that if s only appears as a first vertex in any of the sequences $(x_i^{\tilde{B}})$, then $X_s = \emptyset$.

Let us make a quick observation which follows directly from the definition of the order ρ , of the sequences (x_i) and of the sets X_s .

Observation 5.3. *Let s be a seed. Then for all $x \in X_s$ it holds that $x <_\tau s$.*

6 The Proof of Lemma 2.1

6.1 Preparations

6.1.1 Setting up the constants

First of all, given δ , we choose

$$\varepsilon \leq \frac{\delta^4}{10^{18}}, \tag{5}$$

and apply Lemma 4.4 with input ε^2 and $M_0 := \frac{1}{\varepsilon^2}$. This yields numbers M_1 and n_0 .

We then set

$$\beta := \frac{\varepsilon}{100M_1}.$$

Finally, we choose

$$m_\delta := (n_0 + 1) \cdot \frac{400M_0}{\beta^{10} \cdot \varepsilon \cdot \delta} \tag{6}$$

for the output of Lemma 2.1. So, given the approximation constant α , satisfying $1 \geq \alpha \geq \delta$, we will have that

$$0 < \frac{1}{m_\delta} \ll \beta \ll \varepsilon \ll \delta \leq \alpha, \quad (7)$$

with the explicit dependencies given above.

Now, given $m \geq m_\delta$, and given an $(m + 1)$ -vertex graph G of minimum degree at least $\lfloor \frac{2m}{3} \rfloor$, and a tree T with at most $m - \alpha m$ edges, rooted at r , we will prepare both T and G for the embedding.

6.1.2 Preparing T for the embedding

We apply Lemma 5.1 to obtain a partition of T into a set W of seeds and a set \mathcal{T} of small trees. The small trees divide into F_1 and F_2 , with two-seeded trees $F'_2 \subseteq F_2$, and the lemma also gives us a set \tilde{V} .

Set

$$f_1 := \sum_{\bar{T} \in F_1} |V(\bar{T})| \quad \text{and} \quad f_2 := \sum_{\bar{T} \in F_2} |V(\bar{T})|.$$

Next, add a set W' of vertices to T , each adjacent to r , such that, setting $\tilde{W} := W \cup W'$, we have

$$|\tilde{W}| = 47 \cdot 2^{j^*}$$

for

$$j^* := \lceil \log \frac{2}{\beta^2} \rceil. \quad (8)$$

The only reason for this is that we plan to apply the grouping and ordering of seeds from Subsection 5.2, that is, we would like to see (3) fulfilled. We are going to embed $T \cup W'$ instead of T . Since the number of vertices in W' is a constant, space is not a problem. Indeed, clearly,

$$|\tilde{W}| + |L| + f_1 + f_2 = |V(T)| + |\tilde{W} \setminus W| \leq m - \alpha m + \frac{200}{\beta^2}. \quad (9)$$

6.1.3 Preparing G for the embedding

As a preparation of G for the embedding, we take an ε^2 -regular partition of G as given by Lemma 4.4 (the regularity lemma), into p clusters, for some p with $M_0 < p < M_1$. Consider the reduced graph R_G of G with respect to this partition, as defined below Lemma 4.4.

Observe that because of the minimum degree of G and by (1), we have that

$$\delta_w(R_G) \geq \left(\frac{2}{3} - 13\varepsilon\right)p. \quad (10)$$

Let us now partition the clusters of R_G further. We will divide each cluster into several slices, into which we plan to embed the distinct parts of the tree T which we identified above.

First of all, we choose a set Z of vertices into which we plan to embed L . More precisely, we arbitrarily choose a set $Z \subseteq V(G)$ of size

$$|Z| = |L| + \lceil (\alpha - \frac{\alpha^4}{10^6})m \rceil, \quad (11)$$

choosing the same number of vertices in each part of the partition (plus/minus one vertex).

Now, we will split up the remainder $C \setminus Z$ of each cluster $C \in V(R_G)$ arbitrarily into four sets $C_{\tilde{V}}, C_{\tilde{W}}, C_{F_1}, C_{F_2}$, and a leftover set $C \setminus (Z \cup C_{\tilde{V}} \cup C_{\tilde{W}} \cup C_{F_1} \cup C_{F_2})$ which will not be used. The sets are chosen having the following sizes:

$$|C_{\tilde{V}}| = |C_{\tilde{W}}| = \lceil \frac{\alpha^4 m}{5} \rceil; \quad (12)$$

$$|C_{F_1}| = \lceil \frac{f_1 + \frac{\alpha^4}{5}m}{p} \rceil; \quad (13)$$

and

$$|C_{F_2}| = \lceil \frac{f_2 + \frac{\alpha^4}{5}m}{p} \rceil. \quad (14)$$

This is possible because of (7) and (9).

As we mentioned above, the idea behind this slicing up is that we are planning to put each part X of the tree ($X \in \{\tilde{W}, \tilde{V}, F_1, F_2, L\}$) into the parts C_X of the clusters of R_G , or into Z , respectively. We reserve a bit more than is actually needed for the embedding, in order to always be able to choose well-behaved (typical) vertices, and also in order to account for slightly unbalanced use of the regular pairs when embedding the trees from \mathcal{T} . Since the sets C_X are large enough, regularity properties will be preserved between these sets (cf. Section 4.3).

Let us remark that it is not really necessary to slice the clusters C up as much as we do: the vertices destined to go into slices $C_{\tilde{V}}$ and $C_{\tilde{W}}$ are

actually so few that they could go to any other slice without a problem. But we think the exposition might be clearer if everything is well-controlled.

Finally, we fix a perfect matching M_{F_2} of R_G which exists because of (10). This matching will be used for embedding the larger trees from \mathcal{T} , namely those in F_2 .

6.1.4 The plan

For convenience, for each seed $s \in \tilde{W}$, let \mathcal{T}_s denote the set of all trees from $\mathcal{T} \setminus L$ that are adjacent to s . We are going to traverse the seeds in the transversal order τ , placing each seed s into a suitable cluster $S(s)$ (we will determine this cluster right before embedding s into it). We then embed $\bigcup \mathcal{T}_s$ before embedding any other seed. After having embedded all seeds $s \in \tilde{W}$ and all corresponding trees from $\bigcup \mathcal{T}_s$, we embed all of L in one step at the very end of the embedding process. So, if the seeds are ordered as $s_1, s_2, s_3, \dots, s_{|\tilde{W}|}$ in τ , then we embed in the order

$$s_1, \bigcup \mathcal{T}_{s_1}, s_2, \bigcup \mathcal{T}_{s_2}, s_3, \bigcup \mathcal{T}_{s_3}, \dots, s_{|\tilde{W}|}, \bigcup \mathcal{T}_{s_{|\tilde{W}|}}, L,$$

and at every point in time, the embedded parts of the tree will form a connected set in T .

Each of the three different embedding procedures will be described in detail in one of the following subsections, namely, in Subsection 6.2 (embedding a seed s), in Subsection 6.3 (embedding $\bigcup \mathcal{T}_s$) and in Subsection 6.4 (embedding L).

6.2 Embedding the seeds

6.2.1 Preliminaries

Assume we are about to embed some seed s . Denote by U the set of vertices that, up to this point, have been used for embedding seeds and small trees. So $U \cap Z = \emptyset$ (we will ensure that this will always remain so), and every cluster $C \in V(R_G)$ divides into six sets: $C \cap U$, $C \cap Z$, $C_{\tilde{W}} \setminus U$, $C_{\tilde{V}} \setminus U$, $C_{F_1} \setminus U$, and $C_{F_2} \setminus U$.

Apart from U , it will be useful to have a set $U' \subseteq \bigcup_{C \in V(R_G)} (C_{F_1} \setminus U)$ of vertices for which at some point we decided that they will never be used for the embedding. The main use of this set U' is that after embedding certain trees from $\mathcal{T}_s \cap F_1$ for some seed s , we can just make all sets C_{F_1} of clusters C

equally ‘occupied’ by discarding some of the vertices of the emptier sets C_{F_1} by putting them into U' . This will be the only time we add vertices to U' . We will make sure that for each seed s the number u'_s of vertices we add to U' while, or directly after, embedding \mathcal{T}_s is bounded by

$$u'_s \leq 600\epsilon m. \quad (15)$$

Since there at most $\frac{2}{\beta}$ (original) seeds in the tree, this means that the set U' will always stay so small that we can ignore it while embedding.

Throughout the embedding, we will ensure that for each parent u of a seed (the parent u might be a seed, or a vertex from \tilde{V}) the following holds. If u was embedded in vertex $\varphi(u)$, then we have that

$$\varphi(u) \text{ is typical to slice } C_{\tilde{W}} \text{ for all but at most } \epsilon p \text{ clusters } C \text{ of } R_G. \quad (16)$$

Note that by Observation 5.3, by the time we reach a seed s , the ‘relevant’ seeds in X_s have already been embedded into a set $\varphi(X_s)$. Let $N_Z(X_s)$ denote the set of all neighbours of vertices from X_s in Z . Let \mathcal{N}_s be the set of corresponding subclusters of the clusters of R_G (i.e., $\bigcup \mathcal{N}_s = N_Z(X_s)$).

6.2.2 Finding the target cluster $S(s)$ for s

Before actually choosing the vertex $\varphi(s)$ we will embed s into, we will determine the target cluster $S(s)$ for a seed s .

Observe that by Lemma 4.1, with $\psi := 13\epsilon$, we know that at least $(\frac{1}{3} + \epsilon^{\frac{1}{3}})m$ of the vertices of G see a $(\frac{1}{2} - \epsilon^{\frac{1}{3}})$ -portion of the vertices in $Z \setminus N_Z(X_s)$. So, for significantly more than a third of the clusters of R_G we have that a significant portion of their vertices see at least $(\frac{1}{2} - \epsilon^{\frac{1}{3}}) \cdot |Z \setminus N_Z(X_s)|$ vertices in $Z \setminus N_Z(X_s)$. Because of regularity, and because of (2), this means that for any such cluster C , all but at most an ϵ -fraction of the vertices in $C_{\tilde{W}}$ has at least $(\frac{1}{2} - 3\epsilon^{\frac{1}{3}}) \cdot |Z \setminus N_Z(X_s)|$ neighbours in $Z \setminus N_Z(X_s)$.

Choose $S(s)$ as any one of the clusters as above. That is, we choose $S(s)$ such that

- (α) all but at most $\epsilon |S(s)_{\tilde{W}}|$ vertices of the set $S(s)_{\tilde{W}}$ have degree at least $(\frac{1}{2} - 3\epsilon^{\frac{1}{3}}) \cdot |Z \setminus N_Z(X_s)|$ into $Z \setminus N_Z(X_s)$;

and such that in addition (unless $s = r$, in which case the following two conditions are void),

(β) $S(s)$ is adjacent to $S(p(s))$;

(γ) $\varphi(p(s))$ is typical with respect to $S(s)_{\tilde{W}}$,

where $S(p(s))$ denotes the cluster the parent $p(s)$ of s was embedded into. Such a choice of $S(s)$ is possible since by (10), cluster $S(p(s))$ has degree almost $\frac{2p}{3}$ in R_G , and because of (16).

6.2.3 Embedding seed s into target cluster $S(s)$

We place s in a vertex $\varphi(s)$ from $S(s)_{\tilde{W} \setminus U}$ such that

(A) $\varphi(s)$ is a neighbour of $\varphi(p(s))$ (where $p(s)$ is the parent of s , and if $s = r$ this restriction is empty);

(B) $\varphi(s)$ is typical to $C_{\tilde{W}}$ for all but at most εp clusters $C \in V(R_G) \setminus S(s)$;

(C) $\varphi(s)$ is typical to $C_{\tilde{V}}$ for all but at most εp clusters $C \in V(R_G) \setminus S(s)$;

(D) $\varphi(s)$ is typical to $C_{F_1} \setminus (U \cup U')$ for all but at most εp clusters $C \in V(R_G) \setminus S(s)$; and

(E) $\varphi(s)$ is typical to C_Z for all but at most εp clusters $C \in V(R_G) \setminus S(s)$.

Such a choice is possible since by (2), almost all vertices in any given cluster are typical with respect to any fixed significant subsets of almost all other clusters.

In particular, (E) implies that

$$\deg_Z(\varphi(s)) \geq \left(\frac{2}{3} - \varepsilon^{\frac{1}{3}}\right)|Z|. \quad (17)$$

6.2.4 Degrees of the embedded seeds into Z

The reason for our choice of $S(s)$ as a cluster fulfilling property (α) from Subsection 6.2.2 is that it allows us to accumulate degree into Z . More precisely, if we consider a seed s together with its relevant seeds X_s , then we know that the union of their neighbourhoods in Z is significantly larger than the neighbourhood of s alone. Better still, the more vertices X_s contains, the larger becomes our bound on this neighbourhood.

We make this informal observation more precise in the following claim.

Claim 6.1. *Let B be a group of seeds.*

(i) *If B has size five, then*

$$|N(\varphi(B)) \cap Z| \geq \left(\frac{47}{48} - \varepsilon^{\frac{1}{4}}\right) \cdot |Z|.$$

(ii) *If B has size four and is of type 1, then*

$$|N(\varphi(B)) \cap Z| \geq \left(\frac{23}{24} - \varepsilon^{\frac{1}{4}}\right) \cdot |Z|.$$

(iii) *If $B = \{b_1, b_2, b_3, b_4\}$ (with the seeds b_i appearing in this order in σ) is of type 2, then*

$$|N(\varphi(B)) \cap Z| \geq \left(\frac{11}{12} - \varepsilon^{\frac{1}{4}}\right) \cdot |Z|,$$

and

$$\min \{|N(\varphi(\{b_1, b_2\})) \cap Z|, |N(\varphi(\{b_3, b_4\})) \cap Z|\} \geq \left(\frac{5}{6} - \varepsilon^{\frac{1}{4}}\right) \cdot |Z|.$$

(iv) *If B is large, say of size $47 \cdot 2^j$, then*

$$|N(\varphi(\{x_i^B : i = 1 \dots, j + 6\})) \cap Z| \geq \left(1 - \frac{1}{96 \cdot 2^j} - \varepsilon^{\frac{1}{4}}\right) \cdot |Z|,$$

where $(x_i^B)_{i=1 \dots, j+6}$ is the sequence defined in Subsection 5.2.2.

Proof. This follows rather directly from (α) and (E) (from Subsections 6.2.2 and 6.2.3, respectively), from (17), and from the definition of the set X_s of relevant seeds (Definition 5.2). For instance, we can calculate the bound in item (i) by using (17), (α) , (E) , and Definition 5.2 (b) to see that

$$|N(\varphi(B)) \cap Z| \geq \left(\frac{2}{3} + \frac{1}{6} + \frac{1}{12} + \frac{1}{24} + \frac{1}{48} - 5 \cdot 3\varepsilon^{\frac{1}{3}}\right) \cdot |Z| \geq \left(\frac{47}{48} - \varepsilon^{\frac{1}{4}}\right) \cdot |Z|.$$

For item (iv), we need to take slightly more care with the calculation. Note that the degree in to Z of the image of the first seed is off $\frac{2}{3}|Z|$ by at most $3\varepsilon^{\frac{1}{3}}|Z|$. The degree of second seed's image is only off $\frac{1}{2}|Z|$ by less than $3\varepsilon^{\frac{1}{3}}\frac{|Z|}{2}$. For the third seed we are only off by $3\varepsilon^{\frac{1}{3}}\frac{|Z|}{4}$, and so on, which means we can actually bound the error in our estimate for the size of the joint neighbourhood in item (iv) by $2 \cdot 3\varepsilon^{\frac{1}{3}}|Z| \leq \varepsilon^{\frac{1}{4}} \cdot |Z|$. \square

6.3 Embedding the small trees

Assume we have successfully embedded a seed s , and are now, before we proceed to the next seed, about to embed all small trees from \mathcal{T}_s (while still leaving any leaves from L adjacent to s unembedded).

Our plan is to embed those trees of \mathcal{T}_s that belong to F_1 into $\bigcup_{C \in V(R_G)} C_{F_1}$, and those trees of \mathcal{T}_s that belong to F_2 into $\bigcup_{C \in V(R_G)} C_{F_2}$. We first explain how we deal with the larger trees, i.e. those in $F_2 \setminus F'_2$, and those in F'_2 . After that we explain how we deal with the constant sized trees, i.e. those in F_1 . Note that actually, it does not matter in which order we deal with the sets $F_1, F_2 \setminus F'_2, F'_2$.

6.3.1 Embedding the trees from $F_2 \setminus F'_2$

For each $\bar{T} \in \mathcal{T}_s \cap (F_2 \setminus F'_2)$, let $r_{\bar{T}}$ denote its root. We plan to put $r_{\bar{T}}$ into $C'_{\tilde{V}} \setminus U$ for some suitable cluster C' . (We will explain below how exactly we do that.) For the rest of $V(\bar{T})$, we proceed as follows.

Recall that we defined a perfect matching of R_G near the end of Section 6.1.1. Choose an edge CD of M_{F_2} that contains at least $3\varepsilon \cdot \frac{m}{p}$ unused vertices in each of C_{F_2}, D_{F_2} . If $|C_{F_2} \setminus U| \geq |D_{F_2} \setminus U|$, we will aim at putting the larger colour class of $\bar{T} - r_{\bar{T}}$ into $C_{F_2} \setminus U$, and otherwise, we aim at putting it into $D_{F_2} \setminus U$. Observe that if we manage to do this for every tree \bar{T} we embed, we can ensure that throughout the process (even when embedding trees from $\mathcal{T}_{s'}$, for some $s' \neq s$), the edges from M_{F_2} keep their free space in a more or less balanced way, that is, for all edges $C'D'$ in M_{F_2} ,

$$|C'_{F_2} \setminus U| \text{ and } |D'_{F_2} \setminus U| \text{ differ by at most } \beta \frac{m}{p}. \quad (18)$$

Let us now explain how we manage to embed \bar{T} in this way. Assume our aim is to embed the children of $r_{\bar{T}}$ into $C_{F_2} \setminus U$, the grandchildren into $D_{F_2} \setminus U$, the grand-grandchildren into $C_{F_2} \setminus U$, and so on. The embedding of $\bar{T} - r_{\bar{T}}$ will be easy using Lemma 4.5 once we found a vertex $\varphi(r_{\bar{T}})$ to embed $r_{\bar{T}}$ into, that is, a vertex that is both a neighbour of $\varphi(s)$ and typical with respect to $C_{F_2} \setminus U$.

So we only need to find a suitable vertex for $\varphi(r_{\bar{T}})$, the root of \bar{T} (which belongs to \tilde{V}). In order to do so, we first determine a cluster C' that is adjacent to both C and $S(s)$, and that fulfills $d(S(s), C') \geq \frac{1}{4}$. At least a third of the clusters in R_G qualify for this, because of (10). Now, by (C) in

the choice of $\varphi(s)$ (in Subsection 6.2.3), we know that $\varphi(s)$ has typical degree into the set $C'_{\tilde{V}}$ for all but very few clusters C' . Typical degree means that $\varphi(s)$ has at least $(\frac{1}{4} - \varepsilon^2) \cdot |C'_{\tilde{V}}|$ neighbours in $C'_{\tilde{V}}$, and by Lemma 5.1 (h), at most $2\beta m$ vertices have been used for earlier vertices from \tilde{V} . So, by (7), we can choose a suitable C' such that $\varphi(s)$ has a large enough neighbourhood in $C'_{\tilde{V}}$ to ensure it contains a vertex $\varphi(r_{\tilde{T}})$ that is typical with respect to $C_{F_2} \setminus U$.

Finally, observe that (18) ensures that the space we had assigned to F_2 is enough for embedding all of $F_2 \setminus F'_2$.

6.3.2 Embedding the trees from F'_2

For each $\tilde{T} \in \mathcal{T}_s \cap F'_2$, we proceed exactly as in the preceding paragraph, except that now, we have to make a small adjustment when we are close to embedding \tilde{v} , the second vertex from \tilde{V} contained in $V(\tilde{T})$.

Suppose s' is the seed which is adjacent to \tilde{v} in T . Because we embed the seeds following the transversal order, we know that s' is not yet embedded by the time we deal with \tilde{T} . We take care to embed \tilde{v} into a vertex that is typical with respect to almost all the sets $C_{\tilde{W}}$. That is, the image of \tilde{v} will be chosen such that (16) holds.

6.3.3 Embedding the trees from F_1

We now explain how we embed the trees from $\mathcal{T}_s \cap F_1$. Note that because of (13) and (15), we have enough space to embed all of $\mathcal{T}_s \cap F_1$. Furthermore, because the trees from F_1 are small, and because of regularity, we have no problem with the actual embedding of them into the regular pairs of G . The only thing we need to make sure is that the roots of the trees from $\mathcal{T}_s \cap F_1$ are embedded into neighbours of $\varphi(s)$, and that we maintain the unused parts of the cluster slices C_{F_1} balanced at all times.

Since there is no matching like M_{F_2} that can be used throughout the whole embedding (i.e., for all seeds), we will have to simultaneously keep *all of the clusters* reasonably balanced. This will be possible because of the rather delicate embedding strategy we employ, and which we will start to explain now.

Preparing the slices C_{F_1} . Assume we are about to start the embedding process of the trees from $\mathcal{T}_s \cap F_1$. First of all, note that we can partition the free space $C_{F_1} \setminus (U \cup U')$ of the slices C_{F_1} of each of the clusters $C \in$

$V(R_G) \setminus \{S(s)\}$ into sets Q_0^C, \dots, Q_r^C for some r , such that $|Q_0^C| < 2\lceil \varepsilon \frac{m}{p} \rceil$ and $|Q_i^C| = \lceil \varepsilon \frac{m}{p} \rceil$ for $i = 1, \dots, r$, and such that for each $i = 1, \dots, r$, either all or none of the vertices in Q_i^C are adjacent to $\varphi(s)$. The reason for doing this is that we now have total control over where exactly the neighbours of $\varphi(s)$ are (since the sets Q_0^C are small enough to be ignored during this step of the embedding). Observe that the sets Q_i^C , for $i = 1, \dots, r$, are large enough to preserve regularity properties, although now we have to replace the regularity parameter ε^2 with $\frac{\varepsilon^2}{\varepsilon} = \varepsilon$.

Consider the graph H with vertex set $\{Q_i^C\}_{i=1, \dots, r, C \in V(R_G)}$ and an edge for each ε -regular pair of sufficient density. Say H has p' vertices. By (10), the weighted minimum degree of R_G is bounded by $\delta_w(R_G) \geq (\frac{2}{3} - 13\varepsilon)p$, and therefore, the weighted minimum degree of H is bounded by $\delta_w(H) \geq (\frac{2}{3} - 17\varepsilon)p'$.

So, by our choice of $\varphi(s)$, in particular by (D) of Subsection 6.2.3, we know that $\varphi(s)$ has neighbours in at least $(\frac{2}{3} - 20\varepsilon)p'$ of the sets Q_i^C . Let N consist of a set of $\lceil (\frac{2}{3} - 20\varepsilon)p' \rceil$ sets Q_i^C that contain neighbours of s . We apply Lemma 4.3 with $\xi := 17\varepsilon$ to H to obtain a set X , of size at most $\lceil 255\varepsilon p' \rceil + 1$, and an $(N \setminus X)$ -good matching M , an $(N \setminus X)$ -in-good path partition \mathcal{P}_A and an $(N \setminus X)$ -out-good path partition \mathcal{P}_B of $H - X$.

Set $\mathcal{Q} := \bigcup_{C \in V(R_G)} \{Q_1^C, Q_2^C, \dots, Q_r^C\} \setminus X$. By (13), and by (15), we know that $\bigcup \mathcal{Q}$ is large enough to host all of $\bigcup \mathcal{T}_s \cap F_1$. In fact, if $\bigcup \mathcal{T}_s \cap F_1$ could be embedded absolutely balanced into the sets $Q \in \mathcal{Q}$, then there would even be a leftover space of more than $100\varepsilon \frac{m}{p}$ in each of the sets Q .

Recall that during the embedding of the trees from $\mathcal{T}_s \cap F_1$, we will add some vertices to a set U' , for keeping better track of the balancing of the edges. We will keep U' small, that is, we will ensure that (15) holds.

Preparing $\mathcal{T}_s \cap F_1$. We now partition the set of trees from $\mathcal{T}_s \cap F_1$ into three sets³: the set T_{Bal} contains all the balanced trees, i.e. those trees whose color classes have the same size; the set $T_{NearBal}$ contains all trees having the property that their colour classes differ by exactly one, with the bigger class containing the root; and the set T_{Unbal} contains all the remaining trees, that is all unbalanced trees not belonging to $T_{NearBal}$.

Phase 1. In the first phase of our embedding, we embed all trees from T_{Bal} , using the matching M . We try to spread these trees as evenly as possible

³We remark that it is not really necessary to treat the trees from T_{Bal} separately (as they could be treated together with the trees from T_{Unbal} in Phase 2), but embedding $\bigcup T_{Bal}$ first (in Phase 1) is more instructive.

among the edges of M . It is not difficult to see that by Lemma 5.1 (d), it is possible to make the used part of the clusters differ by at most $\frac{1}{\beta}$ (but even the more obvious weaker bound $\frac{1}{\beta^2}$ is sufficient for our purposes). At the end of this phase of the embedding, we add to U' at most $\frac{1}{\beta}$ unused vertices from each of the clusters $C \in V(M)$, and can thus make sure each of the clusters has exactly the same number of vertices in $C_{F_1} \setminus (U \cup U')$.

Phase 2. In the second phase of our embedding, we embed all trees from T_{Unbal} . We group the trees from T_{Unbal} by their number of vertices, which is some number between 3 and $\frac{1}{\beta}$. Then we subdivide these groups according to the number of vertices belonging to the same colour class as their root. The final groups represent the *types* of trees. Since all trees we consider have order at most $\frac{1}{\beta}$, there are at most $\frac{1}{\beta^2}$ different types.

For each of the types \bar{T} , say with t vertices, and colour classes of sizes t_1 and t_2 , where the class of size t_1 contains the root, we proceed as follows. We go through the elements of our N -out-good path partition \mathcal{P}_B in some fixed order, always embedding only a constant number of trees of type \bar{T} . In each round, we keep the clusters of $H - X$ perfectly balanced. Only when we run out of trees of type \bar{T} , we will (necessarily) have to make a last round, possibly not reaching all elements of \mathcal{P}_B , and thus unbalancing some of the clusters a bit (by at most $\frac{1}{\beta}$).

To make the above description more precise, let us recall that \mathcal{P}_B consists of

- (M1) single edges AB with both ends in N ;
- (M2) paths $ABCD$ with $B, C \in N$; and
- (M3) paths $ABCDEF$ with $B, C, D, E \in N$.

Let us now analyse how the sets $Q \in \mathcal{Q}$ lying in edges or paths from (M1)–(M3) fill up when we embed small trees of type \bar{T} into them in the following specific ways. Clearly, the sets A, B of any edge as in (M1) will each get filled up with t vertices whenever we embed one tree of type \bar{T} in one ‘direction’ and a second tree of type \bar{T} in the other ‘direction’.

The sets Q on paths $ABCD$ as in (M2) will get filled as follows. If we

- perform x rounds in which we embed one tree of the current type \bar{T} in the edge AB , with the root going to B ;

- perform x rounds in which we embed a tree of type \bar{T} in the edge CD , with the root going to C ;
- perform y rounds of embedding a tree of type \bar{T} with the root going to C , but the rest of the tree going to AB ; and
- perform y rounds of embedding a tree of type \bar{T} with the root going to B , but the rest of the tree going to CD ,

then after these $2x+2y$ rounds, A and D have received $xt_2+y(t_1-1)$ vertices, while B and C have received $xt_1+y(t_2+1)$ vertices.

So, if $t_1 > t_2$ (observe that then actually $t_1 \geq t_2 + 2$, since $\bar{T} \notin T_{NearBal}$), we will have the four sets A, B, C, D completely balanced out if we choose $x = t_1 - t_2 - 2$ and $y = t_1 - t_2$. If $t_2 \geq t_1$, we can balance out the four clusters by taking $x = t_2 - t_1 + 2$ and $y = t_2 - t_1$.

For the paths $ABCDEF$ from (M3) we can calculate similarly: Say we do x rounds of embedding of a tree of type \bar{T} in the edge AB and another x rounds embedding it into EF . We then do y rounds of embedding the tree into AB , but with the roots of the tree going into C , and another y rounds putting it into EF , with the root going into D . Moreover, we perform $2z$ rounds where we embed the tree into CD , of which z rounds in each ‘direction’. Then after these $2x + 2y + 2z$ rounds, we filled each of A and F with $xt_2 + y(t_1 - 1)$ vertices, each of B and E with $xt_1 + yt_2$ vertices, and each of C and D with $y + zt$ vertices.

So, if $t_2 \geq t_1$, then with $x = t \cdot (t_2 - t_1 + 1)$, $y = t \cdot (t_2 - t_1)$, and $z = (t - 1) \cdot (t_2 - t_1) + t_1$, we have filled all six sets A, B, C, D, E, F with exactly the same amount of vertices. If $t_1 > t_2$, we choose $x = t \cdot (t_1 - t_2 - 1)$, $y = t \cdot (t_1 - t_2)$, and $z = (t - 1) \cdot (t_1 - t_2) - t_1$.

Resuming all these observations, if $t_1 > t_2$ we can embed $d := 2(t_1 - t_2 - 2)(t_1 - t_2 - 1)(t_1 - t_2)t^2 \cdot ((t - 1) \cdot (t_1 - t_2) - t_1)$ trees of the current type \bar{T} , in a way that all sets $Q \in \mathcal{Q}$ will be used in a completely balanced way. If $t_2 \geq t_1$, we can embed $d' := 2(t_2 - t_1 + 2)(t_2 - t_1 + 1)(t_2 - t_1)t^2 \cdot ((t - 1) \cdot (t_2 - t_1) + t_1)$ trees of type \bar{T} , perfectly balancing all sets $Q \in \mathcal{Q}$. Now, the number of trees of type \bar{T} might fail to be a multiple of d , or of d' , but, by putting some unused vertices into U' , we can finish the embedding of all the trees of the current type without any problems, perfectly balancing all $Q \in \mathcal{Q}$.

Note that the number of vertices we add to U' when working on one t -vertex tree from T_{Unbal} is at most $\max\{dt, d't\} \leq 2t^8 \leq \frac{2}{\beta^8}$. So, the number

of vertices we add to U' after working on all trees from T_{Unbal} is at most the number of types of trees multiplied by $\frac{2}{\beta^8}$, and thus at most $\frac{2}{\beta^{10}}$.

Phase 3. In the third phase of our embedding, we embed the trees from $T_{NearBal}$. Note that each of these trees has at least one leaf in its heavier colour class. Instead of the root, as in phase 2, we will sometimes put the leaf into a different cluster, this time using the N -in-good path partition \mathcal{P}_A .

Easier calculations than in the previous case show that we can completely balance all slices C_{F_1} of clusters $C \in V(M)$ if we embed six trees of type \bar{T} . So, putting at most $\frac{1}{\beta^2} \cdot \frac{6}{\beta} \leq \frac{6}{\beta^3}$ unused vertices into U' , we can finish the embedding of all the trees of $T_{NearBal}$ balancing all slices as desired.

After finishing Phase 3, we still put some more vertices into U' , before declaring the embedding procedure of the trees in $\mathcal{T}_s \cap F_1$ finished. Namely, we put an appropriate number of vertices from the sets in X into U' . That is, the number of vertices from any of the sets of X we add to U' is the same as the number of vertices from any of the sets $Q \in \mathcal{Q}$ that went to U or to U' during the embedding of $\mathcal{T}_s \cap F_1$. This cleaning-up is only done because it will be nicer to be able to start the embedding of the trees at the next seed with all slices C_{F_1} perfectly balanced.

Observe that the number of vertices we added to U' while dealing with the trees from $\mathcal{T}_s \cap F_1$ is at most⁴

$$u'_s \leq \frac{1}{\beta} + \frac{2}{\beta^{10}} + \frac{6}{\beta^3} + |X| \cdot \frac{2|\mathcal{T}_s \cap F_1|}{p'} \leq \frac{3}{\beta^{10}} + 599\epsilon m \leq 600\epsilon m,$$

where we used (6) for the last inequality. Hence the bound (15) we had claimed above is correct. This ensures we have enough space for all future trees from F_1 .

6.4 Embedding the leaves

This section is devoted to the embedding of the leaves. That is, we are now at a stage where we have successfully embedded all seeds and all small trees, and all that is left to embed is L , the set of leaves adjacent to seeds. We will show we can embed all of L at once.

If we cannot embed L into Z , then by Hall's theorem, there is some subset $K \subseteq \tilde{W}$ such that

$$|N(\varphi(K)) \cap Z| < |L_K|, \tag{19}$$

⁴Note that we did not add $\bigcup_{C \in V(R_G)} (Q_0^C \cup Q_1^C)$ to U' .

where L_K is the set of leaves adjacent to elements of K , the set $\varphi(K)$ is the set of images of K , and $N(\varphi(K)) \cap Z$ is the union of the neighbours in Z of the elements of $\varphi(K)$.

Recall that by (E) from Subsection 6.2.3, we chose as the image of a seed s a vertex $\varphi(s)$ that is typical with respect to C_Z for almost all clusters C of R_G . Because of (10), this means that

$$\text{each element of } \varphi(K) \text{ sees at least } \left(\frac{2}{3} - 20\varepsilon\right)|Z| \text{ vertices of } Z. \quad (20)$$

In particular, by (11), each element of $\varphi(K)$ sees more than $\frac{5}{8}(|L| + \frac{9}{10}\alpha m)$ vertices of Z . Thus, it follows that

$$|L| > \frac{3\alpha m}{2}, \quad (21)$$

as otherwise $\frac{5}{8}(|L| + \frac{9}{10}\alpha m) \geq |L|$, which means that we could have embedded L without a problem.

Our aim is to reach a contradiction to the assumption that the set K exists. We will reach this contradiction by proving in Claims 6.2–6.6 that K misses a vertex in each of the large groups and also in most of the small groups of seeds we defined in Subsection 5.2.1. In some of the small groups K actually misses more than one vertex. We will prove these claims by repeatedly using (19).

This means that in total, K misses many vertices from \tilde{W} , and these vertices spread out among the blocks (and thus have a corresponding proportion of the leaves hanging from them). Therefore, we can conclude that $|L_K|$ is smaller than the bound for the neighbourhood of $\varphi(K)$ given in (20), and thus, L_K could have been embedded without a problem, which is a contradiction.

Let us make this outline more precise. We start by proving that each of the large groups has a vertex outside K .

Claim 6.2. *No large group is completely contained in K .*

Proof. Assume otherwise, and consider the largest j for which there is a group of size $47 \cdot 2^j$ completely contained in K . Then by Claim 6.1 (iv) from Subsection 6.2.4, we know that

$$|N(\varphi(K)) \cap Z| \geq \left(1 - \frac{1}{95 \cdot 2^j} - \varepsilon^{\frac{1}{4}}\right)|Z|. \quad (22)$$

If $j \geq j^{\circledast} := \lceil \log \frac{1}{95 \cdot \frac{999}{1000} \alpha} \rceil$, then by (11), this number exceeds $|L|$, which yields a contradiction to (19). So

$$j < j^{\circledast}. \quad (23)$$

In particular, because of (7) and (8), we know that $j < j^*$, and hence, there exists a large group of size $47 \cdot 2^{j+1} = 94 \cdot 2^j$. For any such group B , we know that, by the choice of j , there is a vertex $v_B \in B$ that is not in K . Let L_B be the set of leaves adjacent to seeds in B , and let $\ell_{\max}(B)$ and $\ell_{\min}(B)$ be the number of leaves adjacent to the first and the last seed in B , respectively, under the size order σ . Thus, every seed $b \in B$ is adjacent to a number ℓ_b of leaves, with $\ell_{\min}(B) \leq \ell_b \leq \ell_{\max}(B)$.

Set

$$dif(B) := \ell_{\max}(B) - \ell_{\min}(B).$$

Then

$$\begin{aligned} \ell_{v_B} \geq \ell_{\min}(B) &\geq \frac{|L_B|}{|B|} - dif(B) \cdot \frac{|B| - 1}{|B|} \\ &= \frac{|L_B|}{94 \cdot 2^j} - dif(B) \cdot \left(1 - \frac{1}{94 \cdot 2^j}\right). \end{aligned}$$

Since the groups are consecutive in the size order, and since no seed has more than αm leaves adjacent to it, we know that

$$\sum_{B: |B|=94 \cdot 2^j} dif(B) \leq \alpha m.$$

So, the number of leaves adjacent to seeds that are not in K can be bounded by calculating

$$\begin{aligned} |L| - |L_K| &\geq \sum_{B: |B|=94 \cdot 2^j} \ell_{v_B} \\ &\geq \frac{|L|}{94 \cdot 2^j} - \sum_{B: |B|=94 \cdot 2^j} dif(B) \cdot \left(1 - \frac{1}{94 \cdot 2^j}\right) \\ &\geq \frac{|L|}{94 \cdot 2^j} - \alpha m \cdot \left(1 - \frac{1}{94 \cdot 2^j}\right). \end{aligned}$$

Therefore,

$$\begin{aligned}
|L_K| &\leq \left(1 - \frac{1}{94 \cdot 2^j}\right) \cdot (|L| + \alpha m) \\
&\leq \left(1 - \frac{1}{94 \cdot 2^j}\right) \cdot \left(|Z| + \frac{\alpha^4}{10^6} m\right) \\
&\leq \left(1 - \frac{1}{95 \cdot 2^j} - \varepsilon^{\frac{1}{4}}\right) |Z| \\
&\leq |N(\varphi(K)) \cap Z|,
\end{aligned} \tag{24}$$

where the second and last inequalities follow from (11) and (22), respectively, and the third inequality follows from the observation that

$$\frac{\alpha^4}{10^6} m \leq \frac{2}{3} \cdot \frac{\alpha^3}{10^6} |Z| \leq \left(\frac{1}{94 \cdot 95 \cdot 2^j} - \varepsilon^{\frac{1}{4}}\right) |Z|,$$

where for the first inequality we used that $|Z| \geq |L| > \frac{3\alpha m}{2}$ (by (21)), and the second inequality follows from the facts that $\varepsilon \leq \frac{\alpha^4}{10^{18}}$ (by (5)) and $j \leq j^*$ (by (23)).

Now, inequality (24) gives a contradiction to (19). We have thus proved Claim 6.2. \square

Next, we will show that a similar statement holds for all small groups of size five.

Claim 6.3. *No small group of size 5 is completely contained in K .*

Proof. Indeed, otherwise, because of Claim 6.1 (i), we know that

$$|N(\varphi(K)) \cap Z| \geq \left(\frac{47}{48} - \varepsilon^{\frac{1}{4}}\right) |Z|. \tag{25}$$

Moreover, Claim 6.2 implies that every large group B of size 47 has a vertex v_B which is not in K , and thus we can calculate, similar as above for Claim 6.2, that

$$|L| - |L_K| \geq \sum_{B: |B|=47} \ell_{v_B} \geq \frac{|L|}{47} - \alpha m \cdot \left(1 - \frac{1}{47}\right),$$

and thus, employing (11), we find that

$$|L_K| \leq \left(1 - \frac{1}{47}\right) \cdot \left(|Z| + \frac{\alpha^4}{10^6} m\right),$$

which, with the help of (21) and (25), and using the fact that $\alpha \gg \varepsilon$, yields a contradiction to (19). This proves Claim 6.3. \square

Next, we turn to the groups of size four, separating the treatment of these into two cases depending on their type.

Claim 6.4. *No small group of size 4 and of type 1 is completely contained in K .*

Proof. Otherwise, because of Claim 6.1 (ii), we know that

$$|N(\varphi(K)) \cap Z| \geq \left(\frac{23}{24} - \varepsilon^{\frac{1}{4}}\right)|Z|. \quad (26)$$

By Claim 6.3 we know that every group of size five has a vertex which is not in K . Hence, every large group B of size 47 contains at least two vertices v_B^1 and v_B^2 that are not in K . Moreover, by the definition of the small groups, the first of these vertices, v_B^1 , is one of the first 23 vertices of B under the size order, and the second vertex v_B^2 , is one of the next 23 vertices of B under the size order.

So, we can split the group B minus its last vertex into two groups B_1, B_2 containing the first 23 and the next 23 consecutive elements in the size order, respectively, with $v_B^1 \in B_1$ and $v_B^2 \in B_2$. Defining $dif(B_1), dif(B_2)$ as in Claim 6.2 for each of these two subgroups B_1, B_2 of the group B of size 47, and letting $\ell_{v_B^i}$ denote the number of leaves at v_B^i , for $i = 1, 2$, we can calculate that

$$\ell_{v_B^1} \geq \frac{|L_{B_1}|}{23} - \frac{22}{23} \cdot dif(B_1),$$

and

$$\ell_{v_B^2} \geq \frac{|L_{B_2}|}{23} - \frac{22}{23} \cdot dif(B_2),$$

where L_{B_1} and L_{B_2} denote the sets of leaves adjacent to vertices from B_1 and B_2 , respectively. Thus, letting L_{B_3} denote the set of leaves adjacent to the very last vertex of the group B (of size 47), and noticing that $|L_{B_3}| \leq \frac{|L_B|}{47}$, we can calculate that

$$\begin{aligned} |L| - |L_K| &\geq \sum_{B: |B|=47} (\ell_{v_B^1} + \ell_{v_B^2}) \\ &\geq \frac{|L| - \sum_{B: |B|=47} |L_{B_3}|}{23} - \frac{22}{23} \cdot \sum_{B: |B|=47} (dif(B_1) + dif(B_2)) \\ &\geq \frac{|L|}{23} - \frac{|L|}{23 \cdot 47} - \frac{22}{23} \alpha m. \end{aligned}$$

Therefore, using (11) and (26), we obtain that

$$\begin{aligned}
|L_K| &\leq \frac{22}{23}(|L| + \alpha m) \cdot \left(1 + \frac{1}{22 \cdot 47}\right) \\
&\leq \frac{22}{23}\left(|Z| + \frac{\alpha^4}{10^6}m\right) \cdot \left(1 + \frac{1}{22 \cdot 47}\right) \\
&\leq \left(\frac{23}{24} - \varepsilon^{\frac{1}{4}}\right)|Z| \\
&\leq |N(\varphi(K)) \cap Z|,
\end{aligned}$$

a contradiction to (19). This proves Claim 6.4. \square

Claim 6.5. *No small group of size 4 and of type 2 is completely contained in K .*

Proof. Otherwise, because of Claim 6.1 (iii), we know that

$$|N(\varphi(K)) \cap Z| \geq \left(\frac{11}{12} - \varepsilon^{\frac{1}{4}}\right)|Z|. \quad (27)$$

However, by Claims 6.3 and 6.4, we know that every small group of size 5 and every small group of size 4 and of type 1 has a vertex which is not in K . So, we can split every large group B of size 47 into five subgroups B_1, B_2, B_3, B_4, B_5 , each consecutive in the size order, and with $|B_i| = 11$ for $i = 1, 2, 3, 4$ and $|B_5| = 3$, such that each of B_1, B_2, B_3, B_4 contains a vertex $v_B^1, v_B^2, v_B^3, v_B^4 \notin K$.

Similar as above, we can calculate that

$$\begin{aligned}
|L| - |L_K| &\geq \sum_{B: |B|=47} (\ell_{v_B^1} + \ell_{v_B^2} + \ell_{v_B^3} + \ell_{v_B^4}) \\
&\geq \frac{|L|}{11} - \frac{3|L|}{11 \cdot 47} - \frac{10}{11}\alpha m,
\end{aligned}$$

where numbers $\ell_{v_B^i}$ are defined as in the previous claim.

Now, using (11) and (27), we obtain that

$$\begin{aligned}
|L_K| &\leq \frac{10}{11}(|L| + \alpha m) \cdot \left(1 + \frac{1}{10 \cdot 16}\right) \\
&\leq \left(\frac{11}{12} - \varepsilon^{\frac{1}{4}}\right)|Z| \\
&\leq |N(\varphi(K)) \cap Z|,
\end{aligned}$$

a contradiction to (19). This proves Claim 6.5. \square

Next, we will show that we can actually get some more out of the groups of type 2.

Claim 6.6. *No small group of size 4 and of type 2 has three or more vertices in K .*

Proof. Indeed, otherwise, because of the second part of Claim 6.1 (iii), we know that

$$|N(\varphi(K)) \cap Z| \geq \left(\frac{5}{6} - \varepsilon^{\frac{1}{4}}\right)|Z|. \quad (28)$$

By Claims 6.3, 6.4 and 6.5, every small group, except possibly those of size one, has a vertex which is not in K . So, similar as in the previous claims, we can calculate that

$$|L| - |L_K| \geq \frac{|L|}{5} - \frac{|L|}{5 \cdot 47} - \frac{3}{4}\alpha m,$$

and then use (11) and (28), to obtain that

$$|L_K| \leq \frac{4}{5}(|L| + \alpha m) \cdot \left(1 + \frac{1}{5 \cdot 47}\right) \leq |N(\varphi(K)) \cap Z|,$$

a contradiction to (19). This proves Claim 6.6. \square

Resumingly, Claims 6.3–6.6 tell us that K misses at least one vertex of each small group of size four or five, and misses at least two vertices from each small group of size four and type 2. Recalling our ordering 4, 4, 4, 4, 5, 4, 4, 4, 4, 5, 4, 1 of the small groups inside each group B of size 47 as given in (4) in Subsection 5.2 (under ordering σ), we see that we can split B into five groups B_1, B_2, B_3, B_4, B_5 such that

- for $i = 1, 3$, the group B_i has 8 vertices, at least 3 of which are not in K ;
- for $i = 2, 4$, the group B_i has 13 vertices, at least 5 of which are not in K ; and
- B_5 has 5 vertices, at least two of which are not in K .

Therefore, similarly as in the calculations for earlier claims, we can deduce that

$$\begin{aligned} |L| - |L_K| &\geq \min\left\{\frac{3}{8}, \frac{5}{13}, \frac{2}{5}\right\} \cdot |L| - \max\left\{\frac{5}{8}, \frac{8}{13}, \frac{3}{5}\right\} \cdot \alpha m \\ &\geq \frac{3|L|}{8} - \frac{5}{8}\alpha m, \end{aligned}$$

and thus by (11), we get

$$|L_K| \leq \frac{5}{8}(|L| + \alpha m) < \left(\frac{2}{3} - 20\varepsilon\right)|Z|,$$

a contradiction to (20).

This means the Hall-obstruction K cannot exist, and we can thus finish the embedding of T by embedding all leaves from L in one step. This finishes the proof of Lemma 2.1.

7 Extending a given embedding

For the companion paper [RS19b], which proves the exact version of Theorem 1.4, we will need a second result, namely Lemma 7.3 below, apart from Lemma 2.1. Lemma 7.3 is actually very similar to Lemma 2.1. The difference is that in the context of [RS19b], we will be in a situation where a small tree T^* is already embedded, except for a small but still considerably sized subset $Q \subseteq V(T^*)$. Vertices in Q are very well chosen, and their neighbours in T^* are embedded in very versatile vertices of G , which means we can leave the embedding of Q for later. Indeed, we will be able to embed vertices from Q into almost any leftover set of the right size.⁵

So, we wish to embed $T - T^*$, and we can count on extra free space of considerable size because of the not-yet-embedded set Q . This is very much like the situation we face in Lemma 2.1 (where we have free space because of the approximation), except that now, we have to cope with the already embedded $T^* - Q$. However, as we will outline below, it is possible to adapt our proof of Lemma 2.1 to the new setting, with one possible exception. That is, if the graph G has a very specific structure (given in Definition 7.1 below), our embedding scheme will fail. The reason for it to fail is that we are not able to find the matchings M we need for the embedding of F_1 . It will be shown in [RS19b] how we can make use of the specific structure of G so that the tree can be embedded also in that case.

Another important point is that, for all proofs in [RS19b], we will count on a significant advantage. That is, we will be in a position to assume that none of the seeds from the cut-up of the tree has many leaves hanging from

⁵This is a bit oversimplified: The truth is that we will be able to absorb Q , that is, we will reserve a set S of vertices of G , embed $T - (T^* - Q)$ into $G - S$, and then complete the embedding by using the leftover plus S for the embedding of Q .

it. This will be crucial for our proof of Lemma 7.3, and it means that we can forget about all the extra work in the proof of Lemma 2.1 that was necessary for embedding the set L of leaves hanging from seeds.

In order to be able to properly state the result of this section, Lemma 7.3, we need two definitions. The first definition describes the structure of a graph to which the method from the present paper cannot be applied.

Definition 7.1. *We say a graph G on $m + 1$ vertices is γ -special if $V(G)$ consists of three mutually disjoint sets X_1, X_2, X_3 such that*

- (i) $\frac{m}{3} - 3\gamma m \leq |X_i| \leq \frac{m}{3} + 3\gamma m$ for each $i = 1, 2, 3$; and
- (ii) there are at most $\gamma^{10}|X_1| \cdot |X_2|$ edges between X_1 and X_2 .

The next definition describes the already embedded subtree T^* .

Definition 7.2 (γ -nice subtree). *Let T be a tree with m edges. Call a subtree T^* of T with root t^* a γ -nice subtree if*

- (i) $|T^*| < \gamma m$; and
- (ii) every component of $T - T^*$ is adjacent to t^* .

We are now ready for the result we will need in [RS19b].

Lemma 7.3. *For all $\gamma < \frac{1}{10^6}$ there are $m_0 \in \mathbb{N}$ and $\lambda > 0$ such that the following holds for all $m \geq m_0$.*

Let G be an $(m + 1)$ -vertex graph of minimum degree at least $\lfloor \frac{2m}{3} \rfloor$, which is not γ -special. Let T be a tree with m edges such that no vertex in T is adjacent to more than λm leaves. Let T^ be a γ -nice subtree of T , with root t^* , let $Q \subseteq V(T^*) \setminus \{t^*\}$, and let $S \subseteq V(G)$ with $|S| \leq |Q| - (\frac{\gamma}{2})^4 m$.*

Suppose there is an embedding of $T^ - Q$ into $G - S$. Then there is an embedding of $T - Q$ into $G - S$ extending the embedding of $T^* - Q$.*

Proof. This proof follows the lines of the proof of Lemma 2.1. Note that the new difficulty is that we have to deal with the already embedded part of the tree T , and the unusable set S , together occupying up to almost γm vertices, which means that the usable degree of the other vertices might drop by almost γm (as they might see all of the used part of G). However, we have the advantage that we do not have to worry much about the leaves hanging from seeds, as there are very few.

Setting the constants. We start by setting the constants. Given γ , we note that we can count on an approximation factor of

$$\alpha := \left(\frac{\gamma}{2}\right)^4$$

for the embedding of $T - Q$. We choose a suitable $\varepsilon \ll \alpha$, in particular, we will need that

$$\varepsilon \leq \gamma^{20}. \tag{29}$$

We then apply Lemma 4.4 to ε^2 and $M_0 := \frac{1}{\varepsilon^2}$ to obtain numbers M_1 and n_0 . We choose $\beta \ll \varepsilon$, and

$$\lambda \leq \frac{\beta^2 \cdot \varepsilon}{3000}.$$

Finally, we choose a sufficiently large m_0 for the output of Lemma 7.3. Re-summingly, we will have that

$$\frac{1}{m_0} \ll \lambda \ll \beta \ll \varepsilon \ll \alpha \ll \gamma. \tag{30}$$

Now, assume we are given a graph G as in the lemma which is not γ -special, a set S , a tree T and a subtree T^* of T , with

$$|V(T^*)| < \gamma m, \tag{31}$$

a vertex $t^* \in V(T^*)$, and a set $Q \subseteq V(T^*) \setminus \{t^*\}$, such that there is an embedding of $T^* - Q$ into a set $\varphi(V(T^*) - Q) \subseteq G - S$, with t^* embedded into $\varphi(t^*)$.

Regularising the host graph. We regularise $G' := G \setminus (\varphi(V(T^* - Q)) \cup S)$, obtaining a reduced graph $R_{G'}$ on p' vertices. Our plan is to extend the embedding of $T^* - Q$ to an embedding of all of $T - Q$ into $G - S$.

Recall that our main problem is that the minimum degree in $R_{G'}$ is no longer bounded from below by $(\frac{2}{3} - 13\varepsilon)p$, as it was in the proof of Lemma 2.1. Because of the possible degree into the set $\varphi(V(T^* - Q))$, we can only guarantee the following bound:

$$\delta_w(R_{G'}) \geq \left(\frac{2}{3} - \gamma + \left(\frac{\gamma}{2}\right)^4 - 13\varepsilon\right)p' \geq \left(\frac{2}{3} - \gamma\right)p'. \tag{32}$$

Cutting the tree. We cut up the forest induced by $V(T - T^*) \cup \{t^*\}$ using Lemma 5.1, making t^* a seed (the root seed). After this, we make additional seeds of all neighbours t of t^* that belong to a tree from F_2 ; let R denote the set of these new seeds. (Note that by Lemma 5.1 (h), these are very few.) Note that this transforms the partition of the tree a little, as any seed in R cuts the tree from F_2 it belonged to. We just add the newly formed small trees to L , F_1 , $F_2 \setminus F'_2$ or F'_2 , as appropriate, and, slightly abusing notation, continue to call these sets L , F_1 , $F_2 \setminus F'_2$ or F'_2 . It will not be necessary to add more extra seeds, as we did in the proof of Lemma 2.1, so the total number of seeds will be bounded by $\frac{3}{\beta^2}$.

We now describe how we embed $T - T^*$.

Embedding leaves at t^* , and reserving for R . We start by embedding the leaves at t^* , into any cluster(s). This is possible because of the minimum degree, and because t^* has at most $\lambda m \ll \varepsilon m$ leaves hanging from it. We denote by L' the set of the remaining leaves from L . Note that this will not disturb the rest of the embedding process, as there are very few leaves at t^* , compared to the approximation.

Next, we choose a cluster C^* such that at least a third of its vertices are neighbours of $\varphi(t^*)$. We reserve a set $C_R^* \subseteq C^*$ of size $\varepsilon^{\frac{1}{3}}m$ consisting of neighbours of $\varphi(t^*)$ in C^* . These reservations will give us some control later on, or more precisely, these reservation ensure that we will not block the neighbourhood of $\varphi(t^*)$ before embedding the seeds from R .

Embedding the trees from F_1^* . Now we embed the trees from the set $F_1^* := F_1 \cap \mathcal{T}_{t^*}$. We provisionally slice up the clusters according to the neighbourhood of $\varphi(t^*)$, so that almost all of the new slices exclusively contain neighbours or non-neighbours of $\varphi(t^*)$. Say \mathcal{S} is the set of all these slices. Call the new reduced graph $R'_{G'}$, whose vertex set is \mathcal{S} (we momentarily disregard those slices that are not of use to us), say $|\mathcal{S}| = p''$.

Next, consider a subset N of size $\lfloor (\frac{2}{3} - \frac{1001}{1000}\gamma)p'' \rfloor$ of the neighbourhood of the cluster that contains $\varphi(t^*)$. We will now see that since G is not γ -special, it is possible to find a matching M^* and path partitions \mathcal{P}_A^* and \mathcal{P}_B^* as in the proof of Lemma 2.1, where we employed Lemmas 4.2 and 4.3.

Indeed, the only possible reason we would not be able to find an N -good matching M^* of $R'_{G'} - Y$ for some set Y of order around $500\varepsilon p''$ in a similar way as in Lemma 4.2 is the following: Every matching from $V(R'_{G'}) \setminus N$ to N

leaves more than $\lfloor 500\varepsilon p'' \rfloor$ vertices from $V(R'_{G'}) \setminus N$ uncovered. But then, there is a Hall-obstruction consisting of a set $X'_1 \subseteq V(R'_{G'}) \setminus N$ having less than $|X'_1| - \lfloor 500\varepsilon p'' \rfloor$ neighbours in N . Now, because of condition (32) applied to any vertex from X'_1 , and setting $X_3 := N_N(X'_1)$ and $X_1 := V(R'_{G'}) \setminus N$, we have

$$|X_1 \cup X_3| \geq \left(\frac{2}{3} - \gamma \right) p''. \quad (33)$$

Therefore,

$$\left(\frac{1}{3} + \gamma \right) p'' \geq |X_1| \geq |X'_1| \geq \left(\frac{1}{3} - \frac{\gamma}{2} + 250\varepsilon \right) p'' \geq \left(\frac{1}{3} - \frac{\gamma}{2} \right) p'', \quad (34)$$

and thus,

$$\left(\frac{1}{3} + \gamma \right) p'' > |X_1| > |X_3| = |X_1 \cup X_3| - |X_1| \geq \left(\frac{1}{3} - 2\gamma \right) p''. \quad (35)$$

Letting X_2 denote the non-neighbours of X'_1 in N , we obtain from (32) in a similar way as for X_1 that

$$|X_2| \geq \left(\frac{1}{3} - 2\gamma \right) p''. \quad (36)$$

where we also used (35). Also, by (33), we have that

$$|X_2| \leq \left(\frac{1}{3} + \gamma \right) p''. \quad (37)$$

Because of (31), this means that G is γ -special. Indeed, observe that condition (i) of Definition 7.1 holds for $\bigcup X_1$, $\bigcup X_2$, and $\bigcup X_3 \cup \varphi(V(T^*) - Q) \cup S \cup (V(G') \setminus \bigcup_{S \in \mathcal{S}} V(S))$, because of (34), (35), (36) and (37). Furthermore, condition (ii) of Definition 7.1 holds because of the definition of X_2 , and because we know that by (29), any non-edge in $R'_{G'}$ corresponds to a very sparse pair of clusters in G . However, G being γ -special is against the assumptions of Lemma 7.3, which means that no Hall-type obstruction can exist, and we thus find the matching M^* as desired.

If either of the auxiliary matchings M^A and M^B from Lemma 4.3 which we need to construct the good path partitions \mathcal{P}_A^* and \mathcal{P}_B^* does not exist, then we can conclude in a similar manner that G is γ -special. So we can assume all these matchings and path partitions exist. We embed $\bigcup F_1^*$ into M^* , \mathcal{P}_A^* and \mathcal{P}_B^* , all the time avoiding the set C_R^* .

It is very important that the embedding of $\bigcup F_1^*$ leaves all clusters balanced. For this, we will again use a small set U' of pseudo-used vertices.

Slicing up the clusters. After having embedded all trees from F_1^* , we go back to work in $R_{G'}$. We slice up the yet unused parts of the clusters as before, into sets C_Z , C_W , $C_{\tilde{V}}$, $C_{F_1 \setminus F_1^*}$, and C_{F_2} . The slices C_X reflect the sizes of the corresponding sets X , but we leave sufficient buffer space in each (which we need for the same reasons as in the proof of Lemma 2.1, but also in order to deal with some new problems arising from having $T^* - Q$ already embedded).

We then go through the subtree induced by the seeds of $(T - T^*) \cup \{t^*\}$ and the non-trivial trees hanging from them in a connected way (starting with the root t^*). We embed as usual each seed together with all small trees from $F_1 \cup F_2$ hanging from it⁶, but leave out the leaves for later⁷. We always avoid the set C_R^* , unless we are about to embed a seed $t \in R$.

Embedding the seeds. We embed each seed s in a neighbour of the image $\varphi(p(s))$ of its parent $p(s)$, and typical with respect to the slices C_Z , $C_{\tilde{W}}$, $C_{\tilde{V}}$ and $C_{F_1 \setminus F_1^*}$. Usually, seeds go to $C_{\tilde{W}}$, but seeds t from R go to their reserved space C_R^* . Let us remark that we do not need to group and order our seeds as in the proof of Lemma 2.1, and we also do not need to choose the target clusters as carefully.

Embedding the trees from F_2 . These trees are easy to embed, because we can find the matching M_{F_2} just as before, and it does not matter that the minimum degree is only bounded by (32). In order to make the connections through the slices $C_{\tilde{V}}$, for the vertices from \tilde{V} , we only need to observe that every seed adjacent to trees from F_2' is embedded in a vertex that is typical with respect to almost all slices $C_{\tilde{V}}$.

Embedding the trees from $F_1 \setminus F_1^*$. As above, we slice up the clusters once more, so that almost all of the obtained slices behave uniformly with respect to being a neighbour or not of the image of the most recently embedded seed s . Let us call $R_{G''}$ the graph on these slices, after momentarily discarding those slices that are not of use to us.

Since G is not γ -special, we can find the perfect matchings M and good path partitions as in Lemmas 4.2 and 4.3 in $R_{G''}$, in spite of only having

⁶Except when we deal with t^* , which, together with F_1^* , is already embedded.

⁷As L is small, we could embed the leaves together with their respective seed. However, we prefer to follow the structure of the proof of Lemma 2.1, for the sake of conformity.

condition (32). This can be shown exactly as above, when we embedded the trees from F_1^* . So we can embed the trees from $F_1 \setminus F_1^*$ exactly as in the proof of Lemma 2.1.

Embedding the leaves from L' . In (30), we chose λ small enough so that

$$|L'| \leq \frac{3}{\beta^2} \cdot \lambda m \leq \frac{1}{1000} \varepsilon m.$$

Since the slices C_Z are much larger than $\varepsilon \frac{m}{p}$, and since the embedded seeds (except possibly t^*) see approximately two thirds of almost all of these slices, we have no problem to embed the leaves from L' greedily into $\bigcup_{C \in V(R_{G'})} C_Z$. \square

References

- [AKS95] M. Ajtai, J. Komlós, and E. Szemerédi. On a conjecture of Loeb. In *Graph theory, combinatorics, and algorithms, Vol. 1, 2 (Kalamazoo, MI, 1992)*, 1135–1146. Wiley, New York, 1995.
- [BPS18] G. Besomi, M. Pavez-Signé, and M. Stein. Degree conditions for embedding trees. To appear in *SIAM Journal on Discrete Mathematics*.
- [BD96] S. Brandt and E. Dobson. The Erdős–Sós conjecture for graphs of girth 5. *Discr. Math.*, 150:411–414, 1996.
- [CLNS10] B. Csaba, I. Levitt, J. Nagy-György, and E. Szemerédi. Tight bounds for embedding bounded degree trees. In *Katona G.O.H., Schrijver A., Szenyi T., Sági G. (eds) Fête of Combinatorics and Computer Science (2010)*, vol. 20.
- [EFLS95] P. Erdős, Z. Füredi, M. Loeb, and V. T. Sós. Discrepancy of trees. *Studia Sci. Math. Hungar.*, 30(1-2):47–57, 1995.
- [EG59] P. Erdős and T. Gallai. On maximal paths and circuits of graphs. *Acta Mathematica Academiae Scientiarum Hungarica* 10, 3 (1959), 337–356.

- [HRSW16] F. Havet, B. Reed, M. Stein, and D. Wood. A Variant of the Erdős-Sós Conjecture. *Preprint 2016* arXiv:1606.09343
- [Hax01] P. E. Haxell. Tree embeddings. *J. Graph Theory*, 36(3):121–130, 2001.
- [HKP⁺a] J. Hladký, J. Komlós, D. Piguet, M. Simonovits, M. Stein, and E. Szemerédi. The approximate Loebel–Komlós–Sós Conjecture I: The sparse decomposition. *SIAM Journal on Discrete Mathematics* 31-2 (2017), 945-982.
- [HKP⁺b] J. Hladký, J. Komlós, D. Piguet, M. Simonovits, M. Stein, and E. Szemerédi. The approximate Loebel–Komlós–Sós Conjecture II: The rough structure of LKS graphs. *SIAM Journal on Discrete Mathematics* 31-2 (2017), 983-1016.
- [HKP⁺c] J. Hladký, J. Komlós, D. Piguet, M. Simonovits, M. Stein, and E. Szemerédi. The approximate Loebel–Komlós–Sós Conjecture III: The finer structure of LKS graphs. *SIAM Journal on Discrete Mathematics* 31-2 (2017), 1017-1071.
- [HKP⁺d] J. Hladký, J. Komlós, D. Piguet, M. Simonovits, M. Stein, and E. Szemerédi. The approximate Loebel–Komlós–Sós Conjecture IV: Embedding techniques and the proof of the main result. *SIAM Journal on Discrete Mathematics* 31-2 (2017), 1072-1148.
- [KSS01] J. Komlós, G. Sárközy, and E. Szemerédi. Spanning Trees in Dense Graphs. *Combinatorics, Probability and Computing*, Vol. 5, 397–416 (2001).
- [KSS95] J. Komlós, G. Sárközy, and E. Szemerédi. *Combinatorics, Probability and Computing*, Vol. 4 (3), Proof of a Packing Conjecture of Bollobás. 241–255 (1995).
- [KSS02] J. Komlós, A. Shokoufandeh, M. Simonovits, and E. Szemerédi. The regularity lemma and its applications in graph theory. *Theoretical aspects of computer science*, Lecture Notes in Comput. Sci., Vol. 2292, 84–112 (2002).

- [PS12] D. Piguet and M. J. Stein. An approximate version of the Loebel-Komlós-Sós conjecture. *J. Combin. Theory Ser. B*, 102(1):102–125, 2012.
- [RS19b] B. Reed and M. Stein. Embedding Spanning Trees in Graphs of High Minimum Degree which have a Universal Vertex II: A Tight Result. *Preprint 2019*.
- [SW97] J.-F. Saclé and M. Woźniak. A note on the Erdős–Sós conjecture for graphs without C_4 . *J. Combin. Theory (Series B)*, 70(2):229–234, 1997.