AN APPROXIMATE VERSION OF SUMNER'S UNIVERSAL TOURNAMENT CONJECTURE

DANIELA KÜHN, RICHARD MYCROFT, AND DERYK OSTHUS

Abstract. Sumner's universal tournament conjecture states that any tournament on $2n-2$ vertices contains a copy of any directed tree on n vertices. We prove an asymptotic version of this conjecture, namely that any tournament on $(2+o(1))n$ vertices contains a copy of any directed tree on n vertices. In addition, we prove an asymptotically best possible result for trees of bounded degree, namely that for any fixed Δ, any tournament on $(1+o(1))n$ vertices contains a copy of any directed tree on n vertices with maximum degree at most Δ.

1. Introduction

1.1. **Introduction.** A tournament is an orientation of a complete graph. One of the most well-known problems on tournaments is Sumner's universal tournament conjecture, which was posed in 1971 (see e.g. [19, 22]).

Conjecture 1.1. *Let T be a directed tree on n vertices. Then every tournament on $2n-2$ vertices contains a copy of T.*

The following simple example shows that the bound would be best possible: let G be a regular tournament on $2n-3$ vertices (so every vertex has $n-2$ outneighbours), and let T be a star with all edges directed outwards. Then the central vertex of T has $n-1$ outneighbours, and so G does not contain a copy of T.

A large number of partial results towards Sumner's conjecture have been obtained. Let $f(n)$ denote the smallest integer such that any tournament on $f(n)$ vertices contains any directed tree on n vertices. So Conjecture 1.1 states that $f(n) = 2n-2$. Chung (see [22]) observed that $f(n) \leq n^{1+o(1)}$, and Wormald [22] improved this bound to $f(n) \leq n \log_2(2n/e)$. The first linear bound on $f(n)$ was established by Häggkvist and Thomason [6], who showed that $f(n) \leq 12n$, and also that $f(n) \leq (4+o(1))n$. Havet [7] showed that $f(n) \leq 38n/5-6$, and then Havet and Thomassé [9] used the notion of median orders to improve this to $f(n) \leq (7n-5)/2$. The current best bound is due to El Sahili [5].

Theorem 1.2 ([5]). *Let T be a directed tree on n vertices. Then every tournament on $3n-3$ vertices contains a copy of T.*

The conjecture has also been verified for some classes of trees, such as directed paths. Indeed, a classical result of Redei [18] implies that we can even find a spanning directed path in any tournament.

Theorem 1.3 ([18]). *For any positive integer n, any tournament on n vertices contains a directed path on n vertices.*

Thomason [21] proved a much stronger result, namely that whenever n is sufficiently large, every tournament on n vertices contains every orientation of the path on n vertices (this was a conjecture of Rosenfeld). Havet and Thomassé [10] showed that this even holds for all $n \neq 3, 5, 7$. Reid and Wormald [19] also proved Sumner's conjecture for other (very restricted) classes of trees. Havet and Thomassé [9] proved that Conjecture 1.1 holds for arborescences, i.e. where T has a specified root r so that either every edge of T is directed towards r, or every edge of T is directed away from r.

D. Kühn was partially supported by the EPSRC, grant no. EP/F008406/1. D. Osthus was partially supported by the EPSRC, grant no. EP/E02162X/1 and EP/F008406/1.

We will prove an approximate version of Sumner's conjecture. We also prove an asymptotically sharp bound for trees with bounded maximum degree.

Theorem 1.4. *Let $\alpha > 0$. Then the following properties hold.*

(1) *There exists n_0 such that for any $n \geq n_0$, any tournament G on $2(1+\alpha)n$ vertices contains any directed tree T on n vertices.*
(2) *Let Δ be any positive integer. Then there exists n_0 such that for any $n \geq n_0$, any tournament G on $(1+\alpha)n$ vertices contains any directed tree T on n vertices with $\Delta(T) \leq \Delta$.*

In [15], we prove Sumner's conjecture for large n. The proof relies on the results (and not just the methods) that we prove in this paper.

Part (2) of Theorem 1.4 implies that Sumner's conjecture is true with room to spare for large trees of small maximum degree. The following example shows that (2) is best possible in the sense that the term αn cannot be completely omitted: take a regular tournament H_1 on $2k-1$ vertices, take an arbitrary tournament H_2 on $n-k-1$ vertices and obtain a tournament G on $n+k-2$ vertices from $H_1 \cup H_2$ by adding all edges directed from H_1 to H_2. Also, let T be the tree on n vertices obtained from a directed path on $n-k$ vertices by adding k extra vertices which all send an edge to the initial vertex of the path. Then G contains no copy of T. (We are grateful to P. Allen and O. Cooley for pointing out this example to us.) It would be interesting to know whether the term αn can be reduced to a constant depending only on Δ.

Another class of trees where Sumner's conjecture can be strengthened are trees with few leaves. The first result in this direction was proved by Häggkvist and Thomason [6]. Havet and Thomassé (see [8]) then proposed the following generalization of Sumner's conjecture.

Conjecture 1.5 ([8])**.** *Let T be a directed tree on n vertices with k leaves. Then every tournament on $n+k-1$ vertices contains a copy of T.*

Céroi and Havet [4] proved that this conjecture holds for $k \leq 3$, from which they deduced that Sumner's conjecture holds for all trees with at most 4 leaves.

For our proof of Theorem 1.4 we introduce a decomposition of an arbitrary tournament which searches for dense expanding subgraphs. We then introduce a randomized algorithm for embedding arbitrary trees into such dense expanding graphs. Both tools may be useful for other problems. For example, it would be interesting to know whether our methods can be extended to prove an approximate version of Conjecture 1.5.

1.2. **Outline of the proof.** The notion of a robust outexpander (which was introduced for dense graphs in [16]) is crucial to the proof. Informally, a digraph G is a robust outexpander if for any set $S \subseteq V(G)$ which is not too large or too small, the number of vertices with many inneighbours in S is substantially bigger than $|S|$. Kühn, Osthus and Treglown [16] showed that any robust outexpander G of linear minimum semidegree contains a Hamilton cycle. (Here the minimum semidegree is the minimum of the minimum indegree and the minimum outdegree.) Applying this to the 'reduced digraph' obtained from the Szemerédi regularity lemma, this implies that we can split most of the vertices of G into sets $V_1, V_2, \dots, V_k$ so that the set of edges from V_i to V_{i+1} for each i (addition of the indices taken modulo k) forms a quasirandom and dense bipartite graph. As we shall see, this structure is very useful for embedding trees. On the other hand, it is easy to show that if a tournament G is not a robust outexpander of linear minimum semidegree, then the vertices of G can be split into two parts so that almost all of the edges between the two parts are directed the same way (see Lemma 2.8). We shall then consider whether either of these two parts are robust outexpanders, and so on.

To begin, in Section 2 we shall define the concepts we shall use, and prove various lemmas which will be of use to us later on. Then in Sections 3 and 4 we show that Theorem 1.4 holds with the added condition that G is a robust outexpander of linear minimum semidegree. Indeed, in Section 3, we consider the case where the tournament G is a robust outexpander of linear minimum semidegree on $(1+\alpha)n$ vertices, and T is a directed tree on n vertices of bounded maximum degree. As described above, we can split most of the vertices of G into clusters $V_1, V_2, \dots, V_k$ so that the set of edges from V_i to V_{i+1} is quasirandom and dense for each i. Given this structure on G, one attempt to embed T in G would be to embed each vertex $t \in T$ in the cluster either preceding or succeeding the cluster containing the parent t' of t, according to the direction of the edge between t and t'. However, for many trees this method will fail to give an approximately uniform allocation of vertices of T to the clusters of G, which we require for the embedding to be successful. Instead, we modify this method so that each vertex is embedded as above with probability $1/2$ and is embedded in the same cluster as its parent with probability $1/2$. We show that with high probability this randomised algorithm will indeed give an approximately uniform allocation of vertices of T to the clusters of G, and so will successfully embed T in G.

In Section 4 we begin by strengthening the result from Section 3, showing that if T is a directed tree on n vertices of bounded maximum degree, and G is a tournament on $(1+\alpha)n$ vertices whose reduced graph defined on the clusters $V_1, \dots, V_k$ contains a Hamilton cycle, then we can embed T in G so that the vertices of a chosen small set $H \subseteq V(T)$ are embedded within a specified set $U \subseteq V(G)$. To do this, we embed all vertices 'far' from H by the method described above, which ensures that the vertices of T are allocated approximately uniformly amongst the clusters of G. The remaining vertices of T are instead embedded to ensure that every vertex of H is embedded within U. This result allows us to consider directed trees T of unbounded maximum degree. Indeed, we define for a tree T a 'core tree' T_c, which has the properties that T_c has bounded maximum degree, but each component of $T - T_c$ is small. This enables us to show that any tournament G which is a robust outexpander of linear minimum semidegree on $(2+\alpha)n$ vertices contains any directed tree on n vertices. To do this, we again split most of the vertices of G into sets $V_1, V_2, \dots, V_k$ as described above. We then choose subsets $V_i' \subseteq V_i$ at random so that $|\bigcup_i V_i'|$ is roughly equal to $|T_c|$, and embed T_c into these subsets (actually we first extend T_c to an 'extended tree' T_{ext} and embed T_{ext} into these subsets), using the strengthened result for bounded degree trees to restrict certain vertices of T_c to vertices of G with many inneighbours and outneighbours in $G - \bigcup_i V_i'$. Since each component of $T - T_c$ is small, this will allow us to embed the components of $T - T_c$ one by one in the unoccupied vertices of G to complete the embedding of T in G.

It is a simple exercise to demonstrate that any transitive tournament on n vertices contains any directed tree on n vertices. In Section 5, we prove an analogue of this for almost-transitive tournaments G. This means that the vertices of G can be ordered so that almost all of the edges of G are directed towards the endvertex which is greater in this order. We show that if G is an almost-transitive tournament on $(1+\alpha)n$ vertices and T is a directed tree on n vertices then G contains T.

Finally, in Section 6, we shall use the robust outexpander dichotomy to prove Theorem 1.4. Here we shall describe the proof of the first statement; the proof of the second is very similar. So let G be a tournament on $2(1+\alpha)n$ vertices and let T be a directed tree on n vertices. If G is a robust outexpander of linear minimum semidegree, then our results of Sections 3 and 4 show that G contains T, as desired. On the other hand, if G is not a robust outexpander of linear minimum semidegree then we may split G into two parts as described above. We now examine the larger of these two parts. If this is a robust outexpander of linear minimum semidegree then we stop; otherwise we again split this part into two. We continue in this fashion, always choosing the largest part of G, stopping if this is a robust outexpander and splitting it into two smaller parts if not. If we continue this process but do not find a robust outexpander of linear minimum semidegree, then G must be almost transitive. Indeed, each time we split G most of the edges across the split are directed the same way. So once all of the parts of G are sufficiently small, we can be sure that for some ordering of the vertices of G, almost all of the edges of G are directed according to this order. So by the result from Section 5, G contains T, as desired.

So suppose instead that at some stage we stop because the largest part of G is a robust outexpander of linear minimum semidegree. Then we divide T into parts to be embedded amongst the parts of G, so that each part of G receives a part of T approximately proportional to its size. However, the robust outexpander part of G will actually receive slightly more vertices of T than it would from a proportional split. The results from Sections 3 and 4 guarantee that this part of T can still be embedded into the corresponding part of G. Since then the other parts of G will receive slightly fewer vertices of T than they would from a proportional split it will be possible to embed the remainder of T.

2. Definitions.

2.1. **Notation.** For a graph G, we shall write $V(G)$ for the vertex set of G, and $|G|$ for the number of vertices of G. $E(G)$ denotes the set of edges of G, and $e(G) := |E(G)|$. Similarly for sets $X, Y \subseteq V(G)$, $e(X, Y)$ denotes the number of edges between X and Y. We shall sometimes write $v \in G$ to mean $v \in V(G)$. The *degree* of a vertex $v \in G$, denoted $d(v)$, is the number of edges $e \in E(G)$ incident to v. We denote the minimum and maximum degree (taken over all vertices of G) by $\delta(G)$, and $\Delta(G)$ respectively. The *distance* $d(u, v)$ between vertices $u, v \in G$ is the length of the shortest path connecting u and v.

A *tree* is a connected graph which does not contain any cycles. We will often use the fact that for any subtree T' of a tree T and any vertex $x \in T$ there is a unique vertex $y \in T'$ which minimises $d(x, y)$ over all $y \in T'$. For any vertex $x \in T$ and edge $e \in E(T)$ incident to x, the *weight of e from x*, denoted $w_e(x)$, is the number of vertices $y \neq x$ of T for which e is the first edge of the path from x to y. Each vertex $y \neq x$ of T contributes to the weight from x of precisely one edge incident to x, so the sum of the weights from x over all edges incident to x is $|T| - 1$. Also, if xy is an edge of T, then $w_e(x) + w_e(y) = |T|$.

A *rooted tree* is a tree with a specified vertex r as a *root*. In a rooted tree every vertex x other than the root has a *parent*; this is defined to be the unique neighbour y of x with $d(y, r) < d(x, r)$. If y is the parent of x then we say that x is a *child* of y. A *leaf* in a tree is a vertex of degree one; so every vertex other than the root is a child of some vertex, and every vertex apart from a leaf is a parent of some vertex. An *ancestral ordering* of the vertices of a tree is a linear order in which the root appears first and every other vertex appears after its parent.

A *directed graph* $G = (V, E)$, or digraph, is formed by a vertex set V and a set of edges E, where every edge $e \in E$ is an ordered pair (u, v) of vertices of G. For $u, v \in V$ we write $u \to v$ or $v \leftarrow u$ if $(u, v) \in E(G)$. Also, for any vertex v of G, $N^+(v)$ denotes the set of vertices u such that $(v, u) \in E(G)$, and $N^-(v)$ denotes the set of vertices u such that $(u, v) \in E(G)$. $d^+(v)$ and $d^-(v)$ denote $|N^+(v)|$ and $|N^-(v)|$ respectively, and $\delta^+(G)$ and $\delta^-(G)$ are then defined to be the minimum of $d^+(v)$ and $d^-(v)$ respectively over all vertices $v \in G$. The minimum *semidegree* is $\delta^0(G) = \min\{\delta^+(G), \delta^-(G)\}$. A *tournament* on n vertices is a digraph G on n vertices in which for any distinct $u, v \in V(G)$ precisely one of $u \to v$ and $u \leftarrow v$ holds. So a tournament can be thought of as an orientation of the complete graph on n vertices. Given a digraph G, the *underlying graph* G_{under} is the graph on $V(G)$ in which there is an edge between u and v if and only if either $u \to v$ or $u \leftarrow v$. We define the distance $d(u, v)$ between distinct vertices u and v of a digraph G to be the distance between those two vertices in the underlying graph G_{under}. Also, if G is a graph or digraph, and H is a subgraph of G, then we write $G - H$ to denote $G[V(G) \setminus V(H)]$, that is, the subgraph of G induced on those vertices not in H.

A directed tree is a digraph T for which the underlying graph T_{under} is a tree and in which at most one of $x \to y$ and $x \leftarrow y$ holds for any pair of vertices x and y of T. We use the notation $x \to y$ to distinguish a directed edge from an undirected edge, for which we use the notation xy. Given a specified vertex r as a root, we define parents and children of vertices of the directed tree T exactly as in the underlying tree T_{under}. Similarly $\Delta(T) = \Delta(T_{\text{under}})$, and the weight $w_e(x)$ of an edge e incident to a vertex x is defined as in T_{under}. Also, for each vertex v, $w^+(x)$ is the sum of $w_e(x)$ over all edges e incident to x directed away from x, and $w^-(x)$ is the sum of $w_e(x)$ over all edges e incident to x directed towards x. More generally, for a subtree T' of T, $w^+(T')$ is the sum of $w_e(x)$ over all edges e directed from a vertex x of T' to a vertex of $T - T'$, and $w^-(T')$ is the sum of $w_e(x)$ over all edges e directed from a vertex of $T - T'$ to a vertex x of T'. We say that a vertex of a digraph is a *sink* vertex if it has no outneighbours, and a *source* vertex if it has no inneighbours. Since a directed tree on n vertices has $n - 1$ edges, any directed tree must contain at least one sink vertex and at least one source vertex.

Throughout the paper we use the notation $x \ll y$ to indicate that for any $y > 0$ there exists $x_0 > 0$ such that for any $0 < x \leq x_0$ the subsequent statements hold. Such statements with more variables are defined similarly. Also, we will sometimes write 'let $x \ll y$' when y has an already fixed positive value; by this we mean that there exists some $x_0 > 0$ such that for any $0 < x < x_0$ the subsequent statements hold. When we use asymptotics such as $o(1)$ we mean that these hold as $n \to \infty$ and all the other parameters are fixed.

2.2. **Probabilistic estimates.** The next lemma, relating to binomial distributions, will be used to show that in the randomised algorithm we use in Section 3, the cluster to which a vertex is allocated is almost independent of the cluster to which a vertex far away is allocated. We use $\mathcal{B}(n, p)$ to denote the binomial distribution with parameters n and p, that is, the number of successes in n independent trials, each of which has probability p of success. So $\mathbb{E}(\mathcal{B}(n, p)) = np$.

Lemma 2.1. *Suppose that $1/k \ll p, (1 - p), \varepsilon$, that $n \geq k^3/6$, and that $X = \mathcal{B}(n, p)$. Then for any $0 \leq r \leq k - 1$,*

$$\mathbb{P}(X \equiv r \mod k) = (1 \pm \varepsilon)/k.$$

Proof. For each $x \in \{0, \ldots, n\}$ let p_x denote $\mathbb{P}(X = x)$, so $p_x = \binom{n}{x} p^x (1-p)^{n-x}$. Let $\mu = np$, so $\mathbb{E}(X) = \mu$, and let $p_\mu = \max\{p_{\lfloor \mu \rfloor}, p_{\lceil \mu \rceil}\}$, so $p_x \leq p_\mu$ for any x. Moreover, if $x \leq y \leq \mu$ or $\mu \leq y \leq x$ then $p_x \leq p_y$. So for any $r, i \in [k]$,

$$\begin{aligned}
\mathbb{P}(X \equiv r \mod k) &= \sum_{\substack{0 \leq x \leq \mu - k \\ x \equiv r \mod k}} p_x + \sum_{\substack{\mu - k < x \leq \mu + k \\ x \equiv r \mod k}} p_x + \sum_{\substack{\mu + k < x \leq n \\ x \equiv r \mod k}} p_x \\
&\leq \sum_{\substack{0 \leq x \leq \mu - k \\ x \equiv r \mod k}} p_{x+i} + 2p_\mu + \sum_{\substack{\mu + k < x \leq n \\ x \equiv r \mod k}} p_{x-k+i} \\
&\leq \mathbb{P}(X \equiv r + i \mod k) + 2p_\mu.
\end{aligned}$$

So $\mathbb{P}(X \equiv r \mod k) = 1/k \pm 2p_\mu = (1 \pm \varepsilon)/k$ for any $r \in [k]$, using a standard result (e.g. [3], Section 1.2) on the binomial distribution which states that $p_\mu = O(n^{-1/2}) = O(k^{-3/2})$. □

The following two results give useful tail estimates for random variables. The first is an Azuma-type inequality which bounds the sum of many small and almost independent random variables. This is derived in [20] from a result in [17]. ([20] uses a random walk to embed trees in sparse undirected graphs.) The second gives standard Chernoff-type bounds for the binomial and hypergeometric distributions. The hypergeometric random variable X with parameters (n, m, k) is defined as follows. Let N be a set of size n, and fix a set $S \subseteq N$ of size $|S| = m$. Now choose a set $T \subseteq N$ of size $|T| = k$ uniformly at random. Then $X = |T \cap S|$. Note that $\mathbb{E}X = km/n$.

Lemma 2.2 ([20], Proposition 1.1)**.** *Let $X_1, \dots, X_n$ be random variables taking values in $[0, 1]$ such that for each $k \in [n]$,*

$$\mathbb{E}(X_k \mid X_1, \dots, X_{k-1}) \leq a_k.$$

Let $\mu \geq \sum_{i=1}^{n} a_i$. Then for any $0 < \delta < 1$,

$$\mathbb{P}(\sum_{i=1}^{n} X_i > (1 + \delta)\mu) \leq e^{-\frac{\delta^2 \mu}{3}}.$$

Proposition 2.3 ([11], Corollary 2.3 and Theorem 2.10)**.** *Suppose X has binomial or hypergeometric distribution and $0 < a < 3/2$. Then $\mathbb{P}(|X - \mathbb{E}X| \geq a\mathbb{E}X) \leq 2e^{-\frac{a^2}{3}\mathbb{E}X}$.*

2.3. **Regularity and Robust Outexpanders.** To prove Theorem 1.4 we shall make use of a directed version of Szemerédi's Regularity lemma. For this, we make the following definitions. If G is an undirected bipartite graph with vertex classes X and Y, then the density of G is defined as

$$d(X, Y) := \frac{e(X, Y)}{|X||Y|}.$$

Now, for any $\varepsilon > 0$, we say that G is *ε-regular* if for any $X' \subseteq X$ and $Y' \subseteq Y$ with $|X'| \geq \varepsilon|X|$ and $|Y'| \geq \varepsilon|Y|$ we have $|d(X', Y') - d(X, Y)| < \varepsilon$.

Given disjoint vertex sets X and Y in a digraph G, we use $G[X \to Y]$ to denote the edges of G directed from X to Y. We say $G[X \to Y]$ is *ε-regular with density d* if the underlying bipartite graph of $G[X \to Y]$ is ε-regular and has density d. Next we state the degree form of the regularity lemma for digraphs. A regularity lemma for digraphs was proven by Alon and Shapira [2]. The degree form follows from this in the same way as the undirected version (see [14] for a sketch of the latter).

Lemma 2.4 (Regularity Lemma for digraphs)**.** *For any ε, M' there exist M, n_0 such that if G is a digraph on $n \geq n_0$ vertices and $d \in [0, 1]$, then there exists a partition of $V(G)$ into $V_0, \dots, V_k$ and a spanning subgraph G' of G such that*

1. *$M' \leq k \leq M$,*
2. *$|V_0| \leq \varepsilon n$,*
3. *$|V_1| = \cdots = |V_k| =: m$,*
4. *$d^+_{G'}(x) > d^+_G(x) - (d + \varepsilon)n$ for all vertices $x \in V(G)$,*
5. *$d^-_{G'}(x) > d^-_G(x) - (d + \varepsilon)n$ for all vertices $x \in V(G)$,*
6. *for all $i \in [k]$ the digraph $G'[V_i]$ is empty,*
7. *for all $1 \leq i, j \leq k$ with $i \neq j$ the pair $G'[V_i \to V_j]$ is ε-regular and either has density 0 or density at least d.*

We refer to $V_1, \dots, V_k$ as *clusters*. Given a graph G on n vertices, we form the *reduced digraph R of G with parameters ε, d and M'* by applying the regularity lemma with these parameters to obtain $V_0, \dots, V_k$. R is then the digraph on vertex set $\{1, \dots, k\}$, with $i \to j$ an edge precisely when $G'[V_i \to V_j]$ is ε-regular with density at least d.

One particular regular structure will appear frequently in Section 3 and Section 4. We say that a digraph G is an *ε-regular d-dense cycle of cluster tournaments* if $V(G) = V_1 \cup \dots \cup V_k$, where the sets V_i are pairwise disjoint and of equal size, and for each i, $G[V_i]$ is a tournament and $G[V_i \to V_{i+1}]$ is ε-regular with density at least d (where here and throughout this paper addition and subtraction on the indices of clusters is to be taken modulo k). We shall often refer to the sets V_i as clusters, as we will obtain them by an application of the regularity lemma.

Now, let $V_1, \dots, V_k$ be disjoint sets of m vertices, and let G be a digraph on vertex set $V_1 \cup \dots \cup V_k$. Let S be a subset of some cluster V_i. Then we say that S is *(c, γ)-good* if for any $V'_{i-1} \subseteq V_{i-1}$ and $V'_{i+1} \subseteq V_{i+1}$ with $|V'_{i-1}| \geq cm$ and $|V'_{i+1}| \geq cm$, S contains at least $\gamma\sqrt{m}$ vertices which each have at least γm inneighbours in V'_{i-1} and at least γm outneighbours in V'_{i+1}. Our main tool in the use of regularity will be the next lemma, which states that if G is a regular and dense cycle of cluster tournaments, then any subset V'_i of any cluster V_i with $|V'_i| \geq \gamma m/2$ contains a (c, γ)-good subset S of size at most $\sqrt{m}$.

Lemma 2.5. *Suppose that $1/m \ll \varepsilon \ll \gamma \ll c, d$. Let G be an ε-regular d-dense cycle of cluster tournaments on clusters $V_1, \dots, V_k$, each of size m. Then for any i and for any $V'_i \subseteq V_i$ of size $|V'_i| = \gamma m/2$, there exists a (c, γ)-good set $S \subseteq V'_i$ with $|S| \leq \sqrt{m}$.*

Proof. Given $V'_i \subseteq V_i$ of size $|V'_i| = \gamma m/2$, choose $S \subseteq V'_i$ at random by including each vertex of V'_i in S with probability $1/\gamma\sqrt{m}$, independently of the outcome for each other vertex. Then by Proposition 2.3, with probability $1 - o(1)$, $|S| \leq \sqrt{m}$.

Now, $G[V_{i-1} \to V'_i]$ and $G[V'_i \to V_{i+1}]$ are each $(2\varepsilon/\gamma)$-regular with density at least $d/2$. So all but at most $2\varepsilon m/\gamma$ vertices $v_{i-1} \in V_{i-1}$ have at least $\gamma dm/5$ outneighbours in V'_i. Fix any such $v_{i-1} \in V_{i-1}$. Then $G[V'_i \cap N^+(v_{i-1}) \to V_{i+1}]$ is $(5\varepsilon/\gamma d)$-regular with density at least $d/2$. So all but at most $5\varepsilon m/\gamma d$ vertices $v_{i+1} \in V_{i+1}$ have at least $\gamma d^2 m/20$ inneighbours in $V'_i \cap N^+(v_{i-1})$. We therefore conclude that all but at most $7\varepsilon m^2/\gamma d$ pairs (v_{i-1}, v_{i+1}) with $v_{i-1} \in V_{i-1}$, $v_{i+1} \in V_{i+1}$ have at least $\gamma d^2 m/20$ common neighbours in V'_i.

By Proposition 2.3, for each such pair (v_{i-1}, v_{i+1}) the probability that (v_{i-1}, v_{i+1}) has fewer than $d^2\sqrt{m}/25$ common neighbours in S decreases exponentially with m, whilst the number of such pairs is quadratic in m. Thus with probability $1 - o(1)$ our randomly selected S will have the property that all but at most $7\varepsilon m^2/\gamma d$ pairs (v_{i-1}, v_{i+1}) with $v_{i-1} \in V_{i-1}$, $v_{i+1} \in V_{i+1}$ have at least $d^2\sqrt{m}/25$ common neighbours in S. We may therefore fix an outcome of our random choice of S such that both of these events of probability $1 - o(1)$ occur.

So if $|V'_{i-1}| \geq cm$ and $|V'_{i+1}| \geq cm$, then we know that at least $c^2m^2/2$ pairs (v_{i-1}, v_{i+1}) with $v_{i-1} \in V'_{i-1}$, $v_{i+1} \in V'_{i+1}$ have at least $d^2\sqrt{m}/25$ common neighbours $s \in S$. Thus there are at least $c^2d^2m^{5/2}/50$ triples of such vertices (v_{i-1}, s, v_{i+1}), so at least $c^2d^2\sqrt{m}/100 \geq \gamma\sqrt{m}$ vertices in S must lie in the common neighbourhood of at least $c^2d^2m^2/100$ such pairs (v_{i-1}, v_{i+1}). (Otherwise there would be fewer than $|V'_{i-1}||V'_{i+1}|c^2d^2\sqrt{m}/100 + |S|c^2d^2m^2/100 \leq c^2d^2m^{5/2}/50$ such triples (v_{i-1}, s, v_{i+1}).) Each of these vertices therefore has at least $c^2d^2m/100 \geq \gamma m$ neighbours in each of V'_{i-1} and V'_{i+1}, as required. □

We will also make use of the following well known observation, which says that if G is a regular and dense cycle of cluster tournaments on clusters $V_1, \dots, V_k$, and we select subsets $U_1 \subseteq V_1, \dots, U_k \subseteq V_k$ uniformly at random, then with high probability the restriction of G to these subsets is also regular and dense. This follows from a lemma of Alon et al. [1] showing that ε-regularity is equivalent to almost all vertices having the expected degree and almost all pairs of vertices having the expected common neighbourhood size. We include the proof for completeness.

Lemma 2.6. *Suppose that $1/m \ll 1/k \ll \varepsilon \ll \varepsilon' \ll d$ and that $m^{1/3} \leq m' \leq m$. Let G be an ε-regular d-dense cycle of cluster tournaments on clusters $V_1, \dots, V_k$, each of size m. For each $i \in [k]$, choose $U_i \subseteq V_i$ of size m' uniformly at random, and independently of all other choices. Then with probability $1 - o(1)$, $G[U_1 \cup \dots \cup U_k]$ is an ε'-regular $d/2$-dense cycle of cluster tournaments.*

Proof. We need to show that with high probability, $G[U_i \to U_{i+1}]$ is ε'-regular with density at least $d/2$ for each i. So fix some $i \in [k]$, and let $d_i \geq d$ be the density of $G[V_i \to V_{i+1}]$. Also, let B_i be the set of vertices $v \in V_i$ for which $|N^+(v) \cap V_{i+1}| \neq (d_i \pm \varepsilon)m$, and let D_i be the set of pairs $v_1 \neq v_2$ of vertices of V_i for which $|N^+(v_1) \cap N^+(v_2) \cap V_{i+1}| \neq (d_i^2 \pm 3\varepsilon)m$. Then since $G[V_i \to V_{i+1}]$ is ε-regular, $|B_i| \leq 2\varepsilon m$. Also, there are at most $2\varepsilon m^2$ pairs in D_i which contain a vertex of B_i, and each $v \in V_i \setminus B_i$ lies in at most $2\varepsilon m$ pairs in D_i, so $|D_i| \leq 4\varepsilon m^2$. So let $B'_i = B_i \cap U_i$ and similarly let D'_i consist of the pairs in D_i for which both vertices of the pair are in U_i. Then by Proposition 2.3, the probability that either $|B'_i| > 4\varepsilon m'$ or $|D'_i| > 8\varepsilon(m')^2$ declines exponentially with m.

Now, for each of the at most m' vertices $v \in U_i \setminus B_i$, by Proposition 2.3 the probability that $|N^+(v) \cap U_{i+1}| \neq (d_i \pm 2\varepsilon)m'$ decreases exponentially with m. Also, for each of the at most $\binom{m'}{2}$ pairs $v_1 \neq v_2$ with $v_1, v_2 \in U_i \setminus D_i$, the probability that $|N^+(v_1) \cap N^+(v_2) \cap U_{i+1}| \neq (d_i^2 \pm 4\varepsilon)m'$ decreases exponentially with m. So with probability $1 - o(1)$, for each i none of these events of exponentially declining probability will hold.

Fix such an outcome of our random choices. Then for each i at least $(1 - 4\varepsilon)m'$ vertices $v \in U_i$ have $|N^+(v_1) \cap U_{i+1}| = (d_i \pm 2\varepsilon)m'$ and at least $\binom{m'}{2} - 8\varepsilon(m')^2$ pairs $v_1, v_2 \in U_i$ have $|N^+(v_1) \cap N^+(v_2) \cap U_{i+1}| = (d_i^2 \pm 4\varepsilon)m'$. It then immediately follows from Lemma 3.2 of [1] that for each i, $G[U_i \to U_{i+1}]$ is ε'-regular (and it is clear that this has density at least $d_i/2 \geq d/2$), as desired. □

We now turn to the concept of a robust outexpander. Let $\mu > 0$, let G be a digraph on n vertices, and let $S \subseteq V(G)$. Then the *robust μ-outneighbourhood* of S, denoted $RN^+_\mu(S)$, is defined to be the set of vertices of G with at least μn inneighbours in S. For constants $0 < \mu \leq \nu < 1$, we say that a digraph G on n vertices is a *robust (μ, ν)-outexpander* if $|RN^+_\mu(S)| \geq |S| + \mu n$ for all $S \subseteq V(G)$ with $\nu n < |S| < (1 - \nu)n$. A recent result from [16] (which in turn relies on results from [13, 12]) states that every robust outexpander with linear minimum semidegree contains a Hamilton cycle. We shall make use of this to prove the next lemma, which states that if a tournament G is a robust outexpander then G contains a regular and dense cycle of cluster tournaments which covers almost all of the vertices of G. We will use this structure when we embed a tree T in a tournament G which is a robust outexpander.

Lemma 2.7. *Suppose that $1/n \ll 1/M \ll 1/M' \ll \varepsilon \ll d \ll \mu \ll \nu \ll \eta < 1$. Let G be a tournament on n vertices which is a robust (μ, ν)-outexpander with $\delta^0(G) \geq \eta n$. Then G contains an ε-regular d-dense cycle of cluster tournaments on clusters $V_1, \dots, V_k$, where $|\bigcup_{i=1}^k V_i| > (1 - \varepsilon)n$, and $M' \leq k \leq M$.*

Proof. Let R be the reduced digraph of G with parameters ε, d and M' obtained by applying Lemma 2.4, and let $k = |R|$, so $M' \leq k \leq M$. Then by Lemma 12 of [16], $\delta^0(R) \geq \eta|R|/2$, and R is a robust $(\mu/2, 2\nu)$-outexpander. Then by Theorem 14 of [16], which states that any robust outexpander of linear minimum semidegree contains a Hamilton cycle, we know that R contains a Hamilton cycle. Let $V_1, \dots, V_k$ be the clusters of R in the order of the cycle. Then $|\bigcup_{i=1}^k V_i| > (1 - \varepsilon)n$, $G[V_i]$ is a tournament for each i and since $V_i \to V_{i+1}$ is an edge of R for each i, $G'[V_i \to V_{i+1}]$ is ε-regular with density at least d. (Here G' is the spanning subgraph of G obtained by Lemma 2.4.) □

Of course, we will sometimes need to embed a tree T in a tournament G which is not a robust outexpander. The next lemma will be a useful tool in this situation; it states that if a tournament G is not a robust outexpander then $V(G)$ can be partitioned into two sets so that most edges between the two sets have the same direction.

Lemma 2.8. *Suppose that $1/n \ll \mu \ll \nu$, that G is a tournament on n vertices and that G is not a robust (μ, ν)-outexpander. Then we can partition $V(G)$ into sets S and S' such that $\nu n < |S|, |S'| < (1-\nu)n$ and $e(G[S \to S']) \leq 4\mu n^2$.*

Proof. Since G is not a robust (μ, ν)-outexpander there exists $S \subseteq V(G)$ such that $|RN^+_\mu(S)| < |S| + \mu n$ and $\nu n < |S| < (1-\nu)n$. Choose such an S, and let $S' = V(G) \setminus S$, so $\nu n < |S'| < (1-\nu)n$ also.

Since G is a tournament, at most $2\mu n + 1$ vertices $v \in S$ have $d^-_{G[S]}(v) < \mu n$, and so at most $2\mu n + 1$ vertices $v \in S$ have $v \notin RN^+_\mu(S)$. So $|RN^+_\mu(S) \setminus S| \leq 3\mu n + 1$, and so the number of edges from S to S' is at most $(3\mu n + 1)|S| + \mu n|S'| \leq 4\mu n^2$. □

2.4. **Basic Tree Properties.** In this section, we shall prove several lemmas which we shall make use of in proving Theorem 1.4. The first two of these will enable us to split a tree into several pieces with properties that will be useful for the analysis of the randomised embedding algorithm used in Section 3.

Lemma 2.9. *Let T be a tree on $n \geq 3$ vertices. Then there exist subtrees T' and T'' of T such that T' and T'' intersect in precisely one vertex of T, every edge of T lies in precisely one of T' and T'', and $e(T'), e(T'') \geq e(T)/3$.*

Proof. We begin by showing that T must contain a vertex v such that every edge e incident to v has $w_e(v) \leq n/2$. Recall that if $e = uv$, then $w_e(u) + w_e(v) = n$, and so at most one of $w_e(u) > n/2$ and $w_e(v) > n/2$ can hold. Since T contains n vertices and $n-1$ edges, by the pigeonhole principle T contains a vertex v so that no edge e incident to v has $w_e(v) > n/2$.

Now, choose such a vertex v in T, and let $v_1, \dots, v_r$ be the neighbours of v in T. For each i, let S_i be the set of vertices x of T such that v_i lies on the path from v to x. Then every vertex of T other than v lies in precisely one set S_i. Now, for each i, let T_i be the tree $T[S_i \cup \{v\}]$. Then each T_i is a subtree of T and every edge of T is contained in precisely one T_i. So $\{e(T_i) : i \in [r]\}$ is a set of positive integers, none greater than $2(n-1)/3$, which sum to $n-1$. Thus there exists $A \subseteq [r]$ such that the sum of elements of $\{e(T_i) : i \in A\}$ lies between $(n-1)/3$ and $2(n-1)/3$. Then if we take $T' = \bigcup_{i \in A} T_i$ and $T'' = \bigcup_{i \notin A} T_i$ then T' and T'' satisfy the conditions of the lemma (in particular, $T' \cap T'' = \{v\}$). □

Lemma 2.10. *Suppose that $1/n \ll 1/\Delta, \varepsilon, 1/k$. Let T be a tree on n vertices satisfying $\Delta(T) \leq \Delta$ and rooted at t_1. Then there exist pairwise disjoint subsets $F_1, \dots, F_r$ of $V(T)$, and vertices $v_1, \dots, v_r$ (not necessarily distinct) of T such that:*

1. *$|\bigcup_{i \in [r]} F_i| \geq (1-\varepsilon)n$.*
2. *$|F_i| \leq n^{2/3}$ for each i.*
3. *For any $i \in [r]$, let $x \in \{t_1\} \cup \bigcup_{j<i} F_j$, and let $y \in F_i$. Then the path from x to y in T includes the vertex v_i.*
4. *For any $y \in F_i$ we have $d_T(v_i, y) \geq k^3$.*

Proof. We begin by splitting T into a family $\mathcal{F}$ of subtrees of T by repeated use of Lemma 2.9. So initially let $\mathcal{F} = \{T\}$, and then we repeat the following step. Let T_{large} be the largest member of $\mathcal{F}$. Use Lemma 2.9 to split T_{large} into subtrees T' and T'' which intersect in a single vertex, partition the edges of T_{large}, and satisfy $e(T'), e(T'') \geq e(T_{large})/3$. Then remove T_{large} from $\mathcal{F}$, and replace it by the two smaller trees T' and T''. After at most $3n^{1/3}$ steps we must have that $|T^*| \leq n^{2/3}$ for every $T^* \in \mathcal{F}$. At this point we terminate the process.

Observe that if $T', T'' \in \mathcal{F}$, then T' and T'' intersect in at most one vertex. Now, form a graph $\mathcal{G}_\mathcal{F}$ with vertex set $\mathcal{F}$ and with an edge between $T', T'' \in \mathcal{F}$ if and only if T' and T'' have a common vertex. Then $\mathcal{G}_\mathcal{F}$ is connected, and so contains a spanning tree $\mathcal{T}_\mathcal{F}$. Choose T_0 to be a member of $\mathcal{F}$ containing the root t_1 of T, and let $T_0, T_1, \dots, T_r$ be an ancestral ordering of the members of $\mathcal{F}$ (thought of as vertices of the tree $\mathcal{T}_\mathcal{F}$). Now, for each $1 \leq i \leq r$ let v_i be the common vertex of T_i and its parent in $\mathcal{T}_\mathcal{F}$. Then define F_i for each $i \in [r]$ by

$$F_i = V(T_i) \setminus \{x \in T : d_T(v_i, x) < k^3\}.$$

It remains to show that $F_1, \dots, F_r$ and $v_1, \dots, v_r$ satisfy the required properties. (4) is immediate from the definition of F_i, and (2) holds since each T_i contained at most $n^{2/3}$ vertices. For (1), observe that every vertex of T was contained in at least one of the subtrees T_i, and that in forming the sets F_i, we deleted at most Δ^{k^3} vertices from each of the at most $3n^{1/3}$ sets $V(T_i)$, so in total at most $3n^{1/3}\Delta^{k^3} \leq \varepsilon n$ vertices of T are not contained in any of the sets F_i.

For condition (3), suppose that $T'_1T'_2T'_3$ is a path in $\mathcal{T}_\mathcal{F}$, and some vertex v lies in $T'_1 \cap T'_3$, but $v \notin T'_2$. Let $v' \in T'_1 \cap T'_2$ and let $v'' \in T'_2 \cap T'_3$. Then $v' \neq v''$, as otherwise T'_1 and T'_3 would have a common vertex other than v. So there is a path from v' to v'' in T which does not contain v, so T contains a cycle, giving a contradiction. Similarly it follows that for any path $T_{i_1} \dots T_{i_j}$ in $\mathcal{T}_\mathcal{F}$, if T_{i_1} and T_{i_j} have a common vertex v, then v lies in each of $T_{i_1}, \dots, T_{i_j}$, and so if $T_{i_{j-1}}$ is the parent of T_{i_j} in $\mathcal{T}_\mathcal{F}$ then $v = v_{i_j}$. Now, for any $i \in [r]$, if $x \in \{t_1\} \cup \bigcup_{j<i} F_j$ and $y \in F_i$, then $x \in T_j$ for some $0 \leq j < i$ and $y \in T_i$. Let $T_jT'_1 \dots T'_sT_i$ be the path from T_j to T_i in $\mathcal{T}_\mathcal{F}$, then T'_s is the parent of T_i in $\mathcal{T}_\mathcal{F}$. So $T_j \cup T'_1 \cup \dots \cup T'_s$ contains a path P_1 from x to v_i and T_i contains a path P_2 from v_i to y. But the property we have proved before implies that P_1 and P_2 only intersect in v_i. Thus $P_1 \cup P_2$ is the path in T from x to y, and $v_i \in P_1 \cup P_2$, as required. It also follows that the sets F_i are pairwise disjoint. □

Recall that in Section 3, we will describe a randomised algorithm for embedding the vertices of a tree T in a digraph G. Whenever this algorithm embeds a vertex t of T in G, it will reserve a set of vertices of G in which to embed the children of t. No other vertices may be embedded in this set until all the children of t have been embedded. For this to work, we need to ensure that there will not be too many of these reserved sets at any point. This motivates the following definition. If T is a rooted tree on n vertices, then we say that an ancestral ordering of the vertices of T is *tidy* if it has the property that for any initial segment $\mathcal{I}$ of the order, at most $\log_2 n$ vertices in $\mathcal{I}$ have a child not in $\mathcal{I}$. The following lemma shows that such an order exists for any tree T.

Lemma 2.11. *Let T be a tree on n vertices rooted at some $t_0 \in T$. Then there exists a tidy ancestral ordering of the vertices of T.*

Proof. We shall prove that for any r, the vertices of any rooted tree T on fewer than 2^r vertices can be given an ancestral ordering so that fewer than r vertices in any initial segment $\mathcal{I}$ have neighbours outside $\mathcal{I}$. Indeed, suppose that this statement is false, and let T rooted at t_0 be a counterexample of minimal order, say of order n. Let r be minimal such that $n < 2^r$. Then let $T_1, \dots, T_s$ be the components of $T - t_0$, ordered in increasing size, and let t_i be the neighbour of t_0 in T_i. We shall think of t_i as the root of the tree T_i. Then $|T_i| < 2^{r-1}$ for $i \leq s-1$, and $T_s < 2^r$. So since T was a minimal counterexample, we can find an ancestral ordering of the vertices of each T_i so that for any $i \leq s-1$, any initial segment of the order of the vertices of T_i contains fewer than $r-1$ vertices with children outside the initial segment, and any initial segment of the order of the vertices of T_s contains fewer than r vertices with children outside the initial segment. Now, we order the vertices of T as follows. Begin with t_0, then add the vertices of T_1 in their order. Next, add the vertices of T_2 in their order, and continue in this fashion. Since the order of the vertices of each T_i was ancestral, this order is also ancestral. Also, any initial segment $\mathcal{I}$ of this order contains fewer than r vertices with children outside $\mathcal{I}$, contradicting the choice of T, and therefore proving the lemma. □

3. Embedding trees of bounded maximum degree in a robust outexpander.

3.1. **Introduction.**

Our aim in this section is the following lemma on embedding trees of bounded maximum degree in robust outexpander tournaments.

Lemma 3.1. *Suppose that $1/n \ll \mu \ll \nu \ll \eta \ll \alpha, 1/\Delta$, that G is a tournament on $(1+\alpha)n$ vertices which is a robust (μ,ν)-outexpander with $\delta^0(G) \geq \eta n$ and that T is a directed tree on n vertices with $\Delta(T) \leq \Delta$. Then G contains a copy of T.*

The proof of this lemma shows that we could actually put $1/\Delta$ lower down in the hierarchy, but this is how will shall apply this lemma later on. To prove this lemma, we begin by applying Lemma 2.7 to find a regular and dense cycle of cluster tournaments in G, containing almost all of the vertices of G. We will then use Lemma 3.2 to find a copy of T within this structure. This lemma is stated separately, and in a stronger form than necessary, as we shall also make use of it in Section 4.

Lemma 3.2. *Suppose that $1/n \ll 1/k, 1/\Delta \ll \varepsilon \ll d \ll \alpha \leq 2$, and that $m = n/k$. Let G be an ε-regular d-dense cycle of cluster tournaments on clusters $V_1, \dots, V_k$ of equal size $(1+\alpha)m$. Let v^* be a vertex of V_1 with at least d^2m inneighbours in V_k and at least d^2m outneighbours in V_2. Finally, let T be a directed tree on n vertices, rooted at t_1 and with $\Delta(T) \leq \Delta$. Then G contains a copy of T, where the vertex t_1 of T corresponds to the vertex v^* of G.*

The main problem in achieving this is to allocate the vertices of T to the clusters V_i in such a way that we can then use the ε-regularity of each $G[V_i \to V_{i+1}]$ to embed the vertices of T in G. When we say we allocate v to V_i this means that v will be embedded to a vertex of V_i, but this embedding has not been fixed yet. We wish to allocate each vertex of T to a cluster V_i so that for most edges $u \to v$ of T, if u is allocated to V_i then v is allocated to V_{i+1}. So if u is allocated to a cluster V_i and $u \to v$ then we say that the *canonical allocation* of v is to the cluster V_{i+1}, whereas if $u \leftarrow v$ then we say that the canonical allocation of v is to the cluster V_{i-1}. If we allocate v canonically, then we say that the edge between u and v has been allocated canonically. One way of allocating the vertices of T to the clusters V_i would be to begin by allocating the root t_1 to V_1, and then to allocate all remaining vertices canonically. However, to successfully embed the vertices of T within the clusters to which they are allocated we will need the vertices of T to be approximately evenly distributed amongst the k clusters. This method will usually not achieve this, for example if T is an anti-directed path.

To obtain an 'even distribution' for any sufficiently large tree of bounded maximum degree, we modify the method so that some vertices (selected randomly) are allocated to the same cluster as their parent, rather than being allocated canonically. However, having large components of vertices which are allocated to the same cluster may prevent a successful embedding of these vertices within this cluster, and so we shall also require that such components are small. This is the motivation behind the Vertex Allocation Algorithm given in the next subsection, which we shall use to allocate the vertices of T.

3.2. **Allocating the vertices of T.** We shall use the following random process to allocate the vertices of T to the clusters V_i.

Vertex Allocation Algorithm:

Input: A directed tree T on n vertices, a root vertex $t_1 \in T$, and clusters $V_1, \dots, V_k$.

Initialisation: Choose an ancestral ordering $t_1, \dots, t_n$ of the vertices of T.

Procedure: At time $\tau = 1$, allocate t_1 to V_1. At time $\tau \geq 1$, we shall allocate t_τ. Let t_σ be the parent of t_τ, which must have appeared before t_τ in the ordering and has therefore already been allocated. Then:

- If $d_T(t_\tau, t_1)$ is odd, then allocate t_τ canonically.
- If $d_T(t_\tau, t_1)$ is even, then allocate t_τ to the same cluster as t_σ with probability $1/2$, and allocate t_τ canonically with probability $1/2$ (where these choices are made independently for each vertex).

Termination: Terminate when every vertex of T has been processed and therefore allocated to some cluster V_j.

Note that the cluster to which a vertex t is allocated by this algorithm depends only on the cluster to which its parent vertex was allocated and the outcome of the random choice when embedding t (if $d(t, t_1)$ is even). Since these choices were independent, the probability of any possible outcome does not depend on which ancestral order of the vertices was chosen in the initialisation step. Now, we say that an edge of T is *allocated within a cluster* if both of its endvertices are allocated to the same cluster. Then we say that an allocation of the vertices of a directed tree T to clusters $V_1, \dots, V_k$ is *semi-canonical* if

(i) every edge of T is either allocated canonically or is allocated within a cluster,
(ii) every edge of T incident to t_1 is allocated canonically, and
(iii) every component of the subgraph of T formed by all edges allocated within a cluster contains at most $\Delta(T)$ vertices.

The next lemma shows that if we allocate the vertices of a directed tree T to clusters $V_1, \dots, V_k$ by applying the Vertex Allocation Algorithm, then the allocation obtained will be semi-canonical, and also that if vertices t and t' are far apart in T then the cluster to which t is allocated is almost independent of the cluster to which t' is allocated. As a consequence, if T is sufficiently large and has bounded maximum degree, each cluster will have approximately equally many vertices of T allocated to it. These properties will allow us to embed the vertices of such a T into a regular and dense cycle of cluster tournaments G in the next subsection.

Lemma 3.3. *Let T be a directed tree on n vertices rooted at t_1. Allocate the vertices of T to clusters $V_1, \dots, V_k$ by applying the Vertex Allocation Algorithm. Then the following properties hold.*

(a) *The allocation obtained will be semi-canonical.*
(b) *Suppose that $1/k \ll \delta$. Let u and v be vertices of T such that u lies on the path from t_1 to v, and $d_T(u, v) \geq k^3$. Then for any $i, j \in [k]$,*

$$\mathbb{P}(v \textit{ is allocated to } V_i \mid u \textit{ is allocated to } V_j) = \frac{1 \pm \delta/4}{k}.$$

(c) *Now suppose also that* $1/n \ll 1/\Delta, 1/k \ll \delta$*, and that* $\Delta(T) \leq \Delta$*. Then with probability* $1 - o(1)$*, each of the* k *clusters* V_i *has at most* $(1+\delta)m$ *vertices of* T *allocated to it, where* $m = n/k$*.*

Proof. (a) The Vertex Allocation Algorithm allocates every vertex either canonically or to the same cluster as its parent, so every edge will be allocated canonically or within a cluster. Furthermore, a vertex t can only be allocated to the same cluster as its parent if $d(t_1, t)$ is even, and so each edge incident to t_1 is allocated canonically. Finally, since edges allocated within a cluster can only be formed when we allocate t_i such that $d(t_i, t_1)$ is even, any such component is a star formed by some t_j and some of the children of t_j.

(b) Since the order in which the vertices are allocated is ancestral, at the stage in our algorithm when we have just allocated u, no other vertex on the path $P(u, v)$ in T from u to v has yet been allocated. So suppose that we have just allocated u to cluster V_j, let ℓ be the length of $P(u, v)$, so $\ell \geq k^3$, and let $u = v_0, v_1, \ldots, v_\ell = v$ be the vertices of $P(u, v)$. Then let $E = \{i \geq 1 \colon d(v_i, t_1) \text{ is even}\}$, so E indicates the vertices with a random element in their allocation, and let $O = [\ell] \backslash E$, so O indicates the vertices which are allocated deterministically. We then split the edges of $P(u, v)$ into four classes:

$$\begin{aligned} F_{\text{canon}} &= \{v_{i-1} \to v_i : i \in O\} \\ B_{\text{canon}} &= \{v_{i-1} \leftarrow v_i : i \in O\} \\ F_{\text{random}} &= \{v_{i-1} \to v_i : i \in E\} \\ B_{\text{random}} &= \{v_{i-1} \leftarrow v_i : i \in E\}. \end{aligned}$$

Then every edge of $P(u, v)$ lies in one of these 4 sets, and so $|F_{\text{canon}}| + |B_{\text{canon}}| + |F_{\text{random}}| + |B_{\text{random}}| = \ell$. Furthermore, each edge in F_{canon} will be allocated canonically, and hence from some V_i to V_{i+1}. Similarly, edges in B_{canon} will be allocated from some V_i to V_{i-1}. Meanwhile, edges in F_{random} or B_{random} will be allocated from some V_i to V_{i+1} or V_{i-1} respectively with probability $1/2$, and within some V_i with probability $1/2$. So let R be the sum of the number of edges from F_{random} which are allocated canonically and the number of edges from B_{random} which are *not* allocated canonically. Since the outcome of the random experiment for each edge is independent of the outcome for any other edge, R has distribution $\mathcal{B}(|E|, 1/2)$. Now, u was allocated to cluster V_j, and so v will be allocated to cluster V_i, where

$$i \equiv j + |F_{\text{canon}}| - |B_{\text{canon}}| + R - |B_{\text{random}}| \mod k.$$

But since $|E| \geq \lfloor \ell/2 \rfloor \geq k^3/3$, Lemma 2.1 applied with $X = R$ and $n = |E|$ implies that for any $r \in [k]$, the probability that $i = r$ is $\frac{1 \pm \delta/4}{k}$, as desired.

(c) Use Lemma 2.10 to choose pairwise disjoint subsets $F_1, F_2, \ldots, F_r$ of $V(T)$ and vertices $v_1, \ldots, v_r \in V(T)$ such that $|\bigcup_{i \in [r]} F_i| \geq (1 - \delta/2k)n$ and $|F_i| \leq n^{2/3}$ for each i, also such that if $j < i$, then any path from t_1 or any vertex of F_j to any vertex of F_i passes through the vertex v_i, and finally such that $d(v_i, F_i) \geq k^3$. We shall prove that (†) with probability $1 - o(1)$, the total number of vertices from any of the sets F_i allocated to cluster V_j is at most $(1 + \delta/2)m$. This will prove the lemma, as the number of vertices of T not contained in any of the sets F_i is at most $\delta m/2$, and so in total at most $(1+\delta)m$ vertices of T are allocated to any cluster V_j.

To prove (†), define random variables X_i^j for each $i \in [r]$, $j \in [k]$ by

$$X_i^j = \frac{\#\text{ of vertices of } F_i \text{ allocated to cluster } V_j}{n^{2/3}},$$

so that each X_i^j lies in the range $[0, 1]$. Then since the cluster to which a vertex t of T is allocated is dependent only on the cluster to which the parent of t is allocated and on the outcome of the random choice made when allocating t, $\mathbb{E}(X_i^j \mid X_{i-1}^j, \dots, X_1^j, v_i \in V_s) = \mathbb{E}(X_i^j \mid v_i \in V_s)$ for all $s \in [k]$. Here we write $v_i \in V_s$ to denote the event that v_i is allocated to V_s. So for any i and j,

$$\begin{aligned}\mathbb{E}(X_i^j \mid X_{i-1}^j, \dots, X_1^j) &\le \max_{s\in[k]} \mathbb{E}(X_i^j \mid X_{i-1}^j, \dots, X_1^j, v_i \in V_s) = \max_{s\in[k]} \mathbb{E}(X_i^j \mid v_i \in V_s)\\ &= \max_{s\in[k]} \frac{\sum_{x\in F_i} \mathbb{P}(x \in V_j \mid v_i \in V_s)}{n^{2/3}} \le \frac{(1+\delta/4)|F_i|}{kn^{2/3}}.\end{aligned}$$

using (b). So, by Lemma 2.2, with probability $1 - o(1)$ we have that for each j,

$$\sum_{i\in[r]} X_i^j \le \frac{(1+\delta/2)m}{n^{2/3}}$$

and so for each j, the total number of vertices from any of the sets F_i allocated to cluster V_j is at most $(1 + \delta/2)m$, proving (†). □

3.3. **Embedding the vertices of T.** Suppose that we have applied the Vertex Allocation Algorithm to find an approximately uniform allocation of the vertices of T to the clusters of G. We now wish to embed T in G so that each vertex is embedded in the cluster to which it is allocated. In principle we could use the blow–up lemma for this. However numerous complications arise, for instance because we embed some edges within clusters and because we allow Δ to be comparatively large in Section 4. Instead, we embed the vertices of T as follows.

Firstly, to deal with the problem of edges which are allocated within a cluster, we shall embed components of T formed by such edges at the same time, using Theorem 1.2 (it would also be easy to do this directly). To do this we make the following definition. Let T be a tree on n vertices with root t_1, and let the vertices of T be allocated to clusters $V_1, \dots, V_k$ by a semi-canonical allocation. Then the *canonical tree* T_{canon} of T is formed by contracting to a single vertex each component of the subgraph of T formed of edges which are allocated within a cluster. Since the allocation is semi-canonical, each such component contains at most Δ vertices – we say that these vertices *correspond* to that contracted vertex in T_{canon}. Note also that no edge incident to t_1 is contracted; we let the root of T_{canon} be the vertex corresponding to t_1. We shall proceed through all of the vertices of T_{canon} in turn using a tidy ancestral order, and at time τ we will embed all of the vertices of T which correspond to the vertex τ of T_{canon} in one step.

Secondly, we must ensure that at each time τ it is possible to carry out this embedding. To do this, each time we embed a vertex $t \in T$ to a vertex $v \in G$, we will use Lemma 2.5 to select sets A_t^+ and A_t^- of outneighbours and inneighbours of v in the clusters succeeding and preceding that of v, each of size at most $2\sqrt{m}$, which are reserved until all of the children of t have been embedded. Indeed, while these sets are reserved, no vertices may be embedded in them other than

(i) the children of t, and
(ii) those vertices of T which correspond to the same vertex of T_{canon} as a child of t.

We shall refer to these vertices as the *canonical children* of t; observe that there are at most Δ^2 such vertices. Since we are proceeding through the vertices of T_{canon} in a tidy ancestral order, this means that at any time τ not too many such sets will be reserved, and so only a small proportion of the vertices of any cluster will be reserved. When we later come to embed a child t' of t for which the edge tt' was allocated canonically, we embed t' in A_t^+ (if $t \to t'$) or A_t^- (if $t \leftarrow t'$) in such a way that we can choose $A_{t'}^+$ and $A_{t'}^-$ as desired.

When reading the next algorithm, one should bear in mind that often it is not apparent that a choice can be made as required by the algorithm. Indeed, if such a choice is not possible then the algorithm terminates with failure. Lemma 3.4 will show that under certain conditions on G, it will always be possible to make such choices, and so we can be sure that the algorithm will succeed.

Vertex Embedding Algorithm

Input:

- A tree T rooted at t_1.
- A constant α and a positive integer m.
- A digraph G on vertex set $V = V_1 \cup \dots \cup V_k$, where each V_i has size $(1+\alpha)m$, and a semi-canonical allocation of the vertices of T to the clusters V_i, with t_1 allocated to V_1.
- Finally, a vertex $v^* \in V_1$ to which t_1 should be embedded, and constants c and γ.

Initialisation: Form the canonical tree T_{canon} of T as explained above, and choose a tidy ancestral ordering $1, 2, \dots, n'$ of the vertices of T_{canon}. Let $t_1, \dots, t_n$ be a corresponding order of the vertices of T (so if $t_i \in T$ corresponds to $i \in T_{\text{canon}}$ and $t_j \in T$ corresponds to $j \in T_{\text{canon}}$ then t_i appears before t_j if and only if $i < j$.)

Procedure: At time τ we shall embed the vertices $t_r, \dots, t_{r+s-1}$ of T corresponding to vertex τ of T_{canon}. Each vertex t_i will be embedded to a vertex v_i of G, where $v_1 = v^*$. Then, for each t_i we will reserve sets $A^+_{t_i}$ and $A^-_{t_i}$ of vertices of G for the canonical children of t_i. To do this, at each time τ with $1 \leq \tau \leq n'$, take the following steps.

(1) We say that a vertex t_i of T is *open* at time τ if t_i has been embedded but some child of t_i has not yet been embedded. Define the set B^τ of vertices of G unavailable for use at time τ to consist of the vertices already occupied and the sets reserved for the canonical children of open vertices, so

$$B^\tau = \{v_1, \dots, v_{r-1}\} \cup \bigcup_{t_i \colon t_i \text{ is open}} (A^+_{t_i} \cup A^-_{t_i}).$$

For each cluster V_j, let $V^\tau_j = V_j \setminus B^\tau$, so V^τ_j is the set of available vertices of V_j.

(2) If $\tau = 1$ embed t_1 to v_1. Alternatively, if $\tau > 1$:

(2.1) Precisely one of the vertices $t_r, \dots, t_{r+s-1}$ of T corresponding to vertex τ of T_{canon} has a parent already embedded; we may assume this vertex is t_r. Let t_p be the already-embedded parent (so $p < r$, and when t_p was embedded sets $A^+_{t_p}$ and $A^-_{t_p}$ were chosen). Let V_j be the cluster to which t_p is embedded.

(2.2) If $t_p \to t_r$, choose a set S of $3s$ vertices of $A^+_{t_p} \subseteq V_{j+1}$ such that for each $v \in S$

$$|N^+(v) \cap V^\tau_{j+2}| \geq \gamma m \text{ and } |N^-(v) \cap V^\tau_j| \geq \gamma m.$$

If $t_p \leftarrow t_r$, choose a set S of $3s$ vertices of $A^-_{t_p} \subseteq V_{j-1}$ so for each $v \in S$

$$|N^+(v) \cap V^\tau_j| \geq \gamma m \text{ and } |N^-(v) \cap V^\tau_{j-2}| \geq \gamma m.$$

(2.3) Then choose a copy of $T[t_r, \dots, t_{r+s-1}]$ in $G[S]$, and embed each vertex t_i to the corresponding vertex v_i in this copy.

(3) In step (2), we embedded each of $t_r, \dots, t_{r+s-1}$ in the same cluster; let V_q be this cluster. For each $r \leq i \leq r+s-1$, choose sets

$$A^+_{t_i} \subseteq N^+(v_i) \cap V^\tau_{q+1} \text{ and } A^-_{t_i} \subseteq N^-(v_i) \cap V^\tau_{q-1}$$

such that the sets $A^+_{t_i}$ and $A^-_{t_i}$ are all pairwise disjoint, each $A^+_{t_i}$ and each $A^-_{t_i}$ is (c,γ)-good, and $|A^+_{t_i}|, |A^-_{t_i}| \leq 2\sqrt{m}$ for each i.

Whenever there are several choices (for example if there are several possibilities for S in (2.2)), take the lexicographically first of these. This ensures that for each input, the output is uniquely defined (i.e. we can view the algorithm as being deterministic).

Termination: If at any point it is not possible to make the choice required, terminate with failure. Otherwise, terminate after every vertex of T_{canon} has been processed, at which point $\psi(t_i) = v_i$ for each $t_i \in T$ is an embedding ψ of T into G, by construction.

Lemma 3.4. *Suppose that $1/n \ll 1/\Delta, 1/k \ll \varepsilon \ll \gamma \ll c \ll d \ll \alpha \leq 2$, and let $m = n/k$.*

(1) *Let T be a directed tree on at most n vertices with root t_1 and $\Delta(T) \leq \Delta$.*
(2) *Let G be an ε-regular d-dense cycle of cluster tournaments on clusters $V_1, \dots, V_k$, each of size $(1+\alpha)m$, and let $v^* \in V_1$ have at least γm inneighbours in V_k and at least γm outneighbours in V_2.*
(3) *Let the vertices of T be allocated to the clusters $V_1, \dots, V_k$ so that at most $(1+\alpha/2)m$ vertices are allocated to any one cluster V_i, and so that the allocation is semi-canonical.*

Then if we apply the Vertex Embedding Algorithm to T and G with this allocation and constants c and γ, then it will successfully embed T into G with t_1 embedded to v^*.

Proof. The Vertex Embedding Algorithm will only fail if at some point it is not possible to make the required choice. So to demonstrate that the algorithm will succeed, it is enough to show that it is always possible to make the required choices.

In the initialisation we are required to choose a tidy ancestral ordering of the vertices of the rooted tree T_{canon}; the existence of such a choice is guaranteed by Lemma 2.11. Now, consider the set of unavailable vertices B^τ at some time τ. Since the Vertex Embedding Algorithm embeds each vertex in the cluster to which it was allocated, we know that at most $(1+\alpha/2)m$ vertices of each V_j are already occupied. Furthermore, suppose that vertex t_i of T is open at time τ. Then t_i must correspond to a vertex $\tau' < \tau$ of T_{canon}, such that τ' has a child $\tau'' \geq \tau$. Since we are processing the vertices of T_{canon} in a tidy order, there can be at most $\log_2 n' \leq \log_2 n$ such vertices of T_{canon}. As each vertex of T_{canon} corresponds to at most Δ vertices of T, at most $\Delta \log_2 n$ vertices of T are open at time τ. Therefore, at any time τ, the total number of vertices in reserved sets $A^+_{t_i}$ and $A^-_{t_i}$ is at most $4\Delta\sqrt{m}\log_2 n \leq \alpha m/4$. So for any cluster V_j, at any time τ at most $(1+\alpha/2)m + \alpha m/4$ vertices of V_j are unavailable, and so $|V_j^\tau| \geq \alpha m/4$.

We can now demonstrate that it is possible to make the other choices that the algorithm asks for. Indeed, in step (2.2), if $t_p \to t_r$ with t_p embedded into V_j, then the algorithm has to choose a set S of $3s \leq 3\Delta$ vertices of $A^+_{t_p}$ such that each $v \in S$ has $|N^+(v) \cap V^\tau_{j+2}| \geq \gamma m$ and $|N^-(v) \cap V^\tau_j| \geq \gamma m$. But when $A^+_{t_p}$ was chosen at an earlier time τ', it was chosen to be (c,γ)-good. Since the vertex v_p to which t_p was embedded is in cluster V_j, $A^+_{t_p} \subseteq V_{j+1}$. Moreover, since $|V^\tau_j| \geq \alpha m/4 \geq (1+\alpha)cm$ and $|V^\tau_{j+2}| \geq \alpha m/4 \geq (1+\alpha)cm$, $A^+_{t_p}$ must contain at least $\gamma\sqrt{m}$ vertices v such that $|N^+(v) \cap V^\tau_{j+2}| \geq \gamma m$ and $|N^-(v) \cap V^\tau_j| \geq \gamma m$. Furthermore, since t_r is a child of t_p, t_p has been open since its embedding, and so only canonical children of t_p (of which there are at most Δ^2) can have been embedded in $A^+_{t_p}$. So it is indeed possible to select such a set S of $3s$ vertices as required. The argument for the case when $t_p \leftarrow t_r$ is similar. As for (2.3), observe that $G[S]$ is a tournament on $3s$ vertices, and that $T[t_r, \dots, t_{r+s-1}]$ is a directed tree on s vertices. So by Theorem 1.2, $G[S]$ contains a copy of $T[t_r, \dots, t_{r+s-1}]$, so we may choose such a copy.

Finally we come to step (3). In this step we have just embedded at most Δ vertices $t_r, \dots, t_{r+s-1}$ in some cluster V_q, and we wish to choose sets $A^+_{t_i}$ and $A^-_{t_i}$ for each such vertex t_i. When embedding these vertices we ensured that for each i the vertex v_i to which t_i was embedded satisfied $|N^+(v_i) \cap V^\tau_{q+1}| \geq \gamma m$ (for $\tau = 1$ this holds instead by the condition on the outneighbours of $v^* = v_1$). So suppose we have chosen $A^+_{t_r}, A^+_{t_{r+1}}, \dots, A^+_{t_{r+\ell-1}}$ and we now wish to choose $A^+_{t_\ell}$. Then the previously chosen $A^+_{t_i}$ contain at most $2\Delta\sqrt{m}$ vertices between them, and so at least $3\gamma m/4 \geq (1+\alpha)\gamma m/2$ vertices of $N^+(v_\ell) \cap V^\tau_{q+1}$ have not been used in these previous sets. So by Lemma 2.5, we may choose a (c,γ)-good set $A^+_{t_\ell} \subseteq N^+(v_\ell) \cap V^\tau_{q+1}$ of size at most $2\sqrt{m}$ which is disjoint from all of the previously chosen $A^+_{t_i}$. Do this for each vertex t_i in turn; the choice of the sets $A^-_{t_i}$ is similar. □

We can now give the proof of the main lemmas of this section, beginning with the proof of Lemma 3.2.

Proof of Lemma 3.2. Apply the Vertex Allocation Algorithm to allocate the vertices of T to the clusters $V_1, \dots, V_k$. Then by Lemma 3.3(a) this allocation is semi-canonical, and by Lemma 3.3(c) at most $(1+\alpha/2)m$ vertices are allocated to each of the k clusters V_i. Next, apply the Vertex Embedding Algorithm to T and G, giving this allocation as input. By Lemma 3.4, this will successfully embed T in G with t_1 embedded to v^*. □

Proof of Lemma 3.1. If $\alpha > 2$ then G contains a copy of T by Theorem 1.2. So we may assume that $\alpha \leq 2$. We begin by introducing new constants $1/n \ll 1/M \ll 1/M' \ll \varepsilon \ll \varepsilon' \ll d \ll \mu$. Then by Lemma 2.7, G contains an ε-regular d-dense cycle of cluster tournaments G' on clusters $V_1, \dots, V_k$, where $M' \leq k \leq M$, and $|V_1| = \dots = |V_k| \geq (1-\varepsilon)(1+\alpha)n/k \geq (1+\alpha/2)n/k$. For each i choose $V'_i \subseteq V_i$ of size $|V'_i| = (1+\alpha/2)n/k$ uniformly at random. By Lemma 2.6 we may fix an outcome of these choices so that $G'' = G'[V'_1 \cup \dots \cup V'_k]$ is a ε'-regular $d/2$-dense cycle of cluster tournaments. So by Lemma 3.2 G'' contains a copy of T, so G contains T also. □

We finish this section with an analogous result to Lemma 3.2 for small trees (i.e. the result does not demand that $|T|$ is large compared to $|G|$).

Lemma 3.5. *Suppose that $1/m \ll 1/k, 1/\Delta \ll \varepsilon \ll d \ll \alpha \leq 2$, and that $1/k \ll \delta$. Let G be an ε-regular d-dense cycle of cluster tournaments on clusters $V_1, \dots, V_k$, each of size $(1+\alpha)m$, and let $v^* \in V_1$ have at least d^2m inneighbours in V_k and at least d^2m outneighbours in V_2. Let T be a directed tree on at most m vertices, rooted at t_1 and with $\Delta(T) \leq \Delta$, and let T^{far} be the subgraph of T induced by the vertices $x \in T$ with $d(t_1, x) \geq k^3$. Let $\mathcal{G}_T$ denote the set of copies of T in G for which the vertex t_1 of T corresponds to vertex v^* of G. Then $\mathcal{G}_T$ is non-empty. Furthermore, there exists a probability distribution on $\mathcal{G}_T$ such that if a member of $\mathcal{G}_T$ is selected at random according to this distribution, then for each i,*

$$\mathbb{E}(\#\text{ vertices of } T^{far} \text{ embedded in } V_i) \leq \frac{(1+\delta)|T^{far}|}{k}.$$

The probability distribution will actually be, for each member of $\mathcal{G}_T$, the probability that applying first the Vertex Allocation Algorithm and then the Vertex Embedding Algorithm gives this copy of T in G (recall that actually the Vertex Embedding Algorithm is purely deterministic).

Proof. Apply the Vertex Allocation Algorithm to allocate the vertices of T to the clusters V_i. Since $|T| \leq m$, at most m vertices can be allocated to any cluster, and the allocation obtained is semi-canonical by Lemma 3.3(a). Next, introduce constants $\varepsilon \ll \gamma \ll c \ll d$, and apply the Vertex Embedding Algorithm to embed T in G. By Lemma 3.4, this will successfully embed T in G, with t_1 embedded to v^*, and every vertex of T embedded in the cluster to which it was allocated. So it remains only to show that for each i, the expected number of vertices of T^{far} allocated to V_i is at most $(1+\delta)|T^{\text{far}}|/k$. But since for any $x \in T^{\text{far}}$ we have $d(x, t_1) \geq k^3$, by Lemma 3.3(b) applied with $u = t_1$,

$$\mathbb{P}(x \text{ is allocated to } V_i) = \frac{(1 \pm \delta)}{k}$$

for each i, and the result follows. □

4. Embedding trees of unbounded maximum degree in a robust outexpander.

4.1. **Section outline.** Having proved the desired result for trees of bounded maximum degree, we now move onto proving a similar result for trees with no such bound, with a constant of 2 rather than 1 in the condition on the order of G. This is the following lemma.

Lemma 4.1. *Suppose that $1/n \ll \mu \ll \nu \ll \eta \ll \alpha$, that G is a tournament on $2(1+\alpha)n$ vertices which is a robust (μ, ν)-outexpander with $\delta^0(G) \geq \eta n$ and that T is a directed tree on n vertices. Then G contains a copy of T.*

To prove this, we shall begin with a definition. In Section 4.2 we shall define the core tree T_c of a tree T. This is a subtree of T which has bounded maximum degree, and the property that all components of $T - T_c$ are small. Then in Section 4.4 we will show that T_c can be extended to an 'extended tree' T_{ext} which also has bounded maximum degree, and also has the property that few vertices of T_{ext} have neighbours outside T_{ext}. We will embed the extended tree T_{ext} by a similar method to that of the previous section. We will need to do this so that the small number of vertices of T_{ext} with neighbours outside T_{ext} are embedded to vertices of G with large in- and outdegree in G. In Section 4.5 we will use our results from Section 3 to prove Lemma 4.6 on embedding trees of bounded maximum degree. This is similar to Lemma 3.2, but allows us also to demand that a small subset $H \subseteq V(T)$ of the vertices of T, satisfying certain conditions, should be embedded in a small subset U of the vertices of G. This will allow us to embed T_{ext} in G in the desired manner. Finally, in Section 4.6 we will complete the proof of Lemma 4.1 by first using Lemma 4.6 to embed T_{ext} into G as described and then embedding each component of $T - T_{\text{ext}}$ in the unoccupied vertices of G.

4.2. **The core tree.** Let T be a tree on n vertices, and let $\Delta \geq 2$ be fixed. Then we say that a vertex v of T is Δ-*core* if every edge incident to v has $w_e(v) \leq (1 - 1/\Delta)n$. We call the subgraph of T induced by Δ-core vertices of T the *core tree of T with parameter* Δ, and denote it by T_c (the value of Δ will always be clear from the context). With this definition, for any tree T, the core tree of T with parameter Δ is the same as the Δ-heart of T considered by Häggkvist and Thomason in [6]. The statements of the next proposition are also noted in Section 3 of [6], but we include the proof for completeness.

Proposition 4.2. *Let T be a tree on n vertices, let $\Delta \geq 2$ and let T_c be the core tree of T with parameter Δ. Then:*

(i) *T_c is a tree containing at least one vertex.*
(ii) *$w_e(x) \geq n/\Delta$ if $e = xy$ is an edge of T_c.*
(iii) *$\Delta(T_c) \leq \Delta$.*
(iv) *Every component subtree T' of $T - T_c$ has $|T'| \leq n/\Delta$.*

Proof. For (i), note that since $\Delta \geq 2$, for any edge $e = uv$ of T at most one of $w_e(u) > (1 - 1/\Delta)n$ and $w_e(v) > (1 - 1/\Delta)n$ holds. Since T has more vertices than edges, there must therefore be some vertex $v \in T$ such that $w_e(v) \leq (1 - 1/\Delta)n$ for every edge e incident to v, and so $v \in T_c$. It remains to show that T_c is connected. Observe that if u, v, w are distinct vertices of T such that there is an edge between u and v and an edge between v and w, then

$w_{uv}(u) > w_{vw}(v)$. Now, suppose $x, y \in T_c$, and let $x = v_1, v_2, \dots, v_r = y$ be the vertices of the path from x to y in T (in order). Suppose for a contradiction that some v_i is not in T_c. Then for some neighbour z of v_i, $w_{v_i z}(v_i) > (1-1/\Delta)n$. If $z \neq v_{i+1}$, then for each $i \leq j \leq r-1$ we have $w_{v_j v_{j+1}}(v_{j+1}) > (1-1/\Delta)n$, and so $y \notin T_c$, giving a contradiction. On the other hand, if $z = v_{i+1}$, then for each $2 \leq j \leq i$, $w_{v_{j-1} v_j}(v_{j-1}) > (1-1/\Delta)n$, and so $x \notin T_c$, again giving a contradiction.

Now, (ii) is immediate from the fact that if $e = xy$ is an edge of T then $w_e(x) + w_e(y) = n$. Then (iii) follows directly from (ii), as the sum of $w_e(v)$ over all edges incident to v is $n-1$.

Finally, for (iv), observe that for any such T' there is $u \in T'$, $v \in T_c$ with $e = uv$ an edge of T. Suppose that $|T'| > n/\Delta$. Then $w_e(v) \geq |T'| > n/\Delta$, and so $w_e(u) \leq (1-1/\Delta)n$. But since $w_{e'}(u) < w_e(v) \leq (1-1/\Delta)n$ for every other edge e' incident to u, this means that $u \in T_c$, giving a contradiction. □

Note that T_c is an undirected tree obtained from an undirected tree T. However we will often refer to the core tree of a directed tree T; this means the directed tree formed by taking the core tree T_c of the underlying graph T_{under} (an undirected tree) and directing each edge of T_c as it is directed in T.

The idea behind this definition is that the core tree is a bounded degree tree. The general technique we shall use to work with a tree T of unbounded maximum degree (in both this and later sections) is to first consider the core tree T_c, and then consider separately each component of $T - T_c$, making use of the fact that each such component is small.

4.3. **Leading paths.** Let T be a tree on n vertices, rooted at t_1, let $H \subseteq V(T)$, and let k be a positive integer. For any vertex $x \in T$, there is a unique path in T from x to t_1; let P_x denote the set of the first k vertices of this path, starting from x. Let $H^1 = \bigcup_{x \in H} P_x$, and then for each $i \geq 1$ let H^{i+1} be formed from H^i by adding the vertices of P_x for any $x \in H^i$ with at least two children in H^i. After at most n steps we must have $H^i = H^{i+1}$, when we terminate the process. We refer to this final H^i as *H with leading paths included*, denoted $\mathcal{P}_k(H)$. So $H \subseteq \mathcal{P}_k(H) \subseteq V(T)$. Note that $\mathcal{P}_k(H)$ depends on both the value of k and the root t_1 of T.

Next we shall prove two results which will enable us to make use of this definition. The first shows that if H is small then $\mathcal{P}_k(H)$ is small, and the second shows that if H is small then it is possible to embed any component T' of $T[\mathcal{P}_k(H)]$ in a regular and dense cycle of cluster tournaments such that the vertices of $V(T') \cap H$ are embedded in the first cluster and the 'root' of H is embedded in a given cluster.

Proposition 4.3. *Let k be any positive integer, let T be a tree on n vertices, rooted at some $t_1 \in T$, and let $H \subseteq V(T)$. Then $|\mathcal{P}_k(H)| \leq 3k|H|$.*

Proof. Consider any component T' of $T[\mathcal{P}_k(H)]$, and let t'_1 be the unique vertex of T' with minimal $d(t_1, t'_1)$. Then every vertex of T' lies on the path from some vertex of H to t_1, and so T' is precisely the set of vertices in paths between t'_1 and vertices of $H \cap V(T')$. Thus only t'_1 and vertices of H can be leaves of T'. It follows that $T[\mathcal{P}_k(H)]$ has at most $2|H|$ leaves. Since $T[\mathcal{P}_k(H)]$ is a forest, it follows that the number of vertices of $T[\mathcal{P}_k(H)]$ with at least two children in $T[\mathcal{P}_k(H)]$ is also at most $2|H|$. Furthermore, any vertex $x \in T$ for which the vertices of P_x were added to $\mathcal{P}_k(H)$ at any stage is either a member of H or has at least two children in $\mathcal{P}_k(H)$. This is true for at most $3|H|$ vertices x, and for each such vertex at most k vertices were added. □

Lemma 4.4. *Suppose that $1/m \ll 1/k \ll \varepsilon \ll d$. Let T be a directed tree rooted at some $t_1 \in T$. Let $H \subseteq V(T)$ be of size $|H| \leq m/10k$, let T' be a component of $T[\mathcal{P}_k(H)]$ which* *does not contain* t_1*, and let* t'_1 *be the unique vertex of* T' *with minimal* $d(t'_1, t_1)$*. Let* G *be an* ε*-regular* d*-dense cycle of cluster tournaments on clusters* $V_1, \dots, V_k$*, each of size* m*. Then for any* $j \in [k]$*,* G *contains a copy of* T' *with the vertex* t'_1 *corresponding to some vertex of* V_j*, and every vertex in* $V(T') \cap H$ *corresponding to some vertex of* V_1*.*

Proof. Informally, from the perspective of t'_1, T' begins with a path of length $k-1$ (from t'_1 to t, say) before possibly branching out. So we shall find a copy of T' in G by first embedding the vertices of this path so that t'_1 is embedded in V_j and t is embedded in V_1. We then embed all of the remaining vertices of T' in V_1.

More formally, note that for each $0 \leq s \leq k-1$ there is precisely one vertex x_s of T' with $d(t'_1, x_s) = s$ (so $x_0 = t'_1$, and $x_i \notin H$ for any $i < k-1$). Let $F \subseteq [k-1]$ be the set of those s such that $x_{s-1} \to x_s$, and let $B \subseteq [k-1]$ be the set of s such that $x_{s-1} \leftarrow x_s$. Then $|F| + |B| = k-1$, so either $|F| > k-j$ or $|B| \geq j-1$. Suppose first that $|B| \geq j-1$. Then choose $B' \subseteq B$ of size $j-1$. We shall allocate the vertices of T' to the clusters $V_1, \dots, V_j$. Begin by allocating x_0 to V_j. Then for each $s \in [k-1]$ in turn, let V_i be the cluster to which x_{s-1} was allocated, and allocate x_s to V_i if $s \notin B'$, or to V_{i-1} if $s \in B'$. Then since $|B'| = j-1$, x_{k-1} will be assigned to V_1. Finally, allocate all other vertices of T' to V_1. Then every edge of T' is allocated either canonically or within a cluster.

Next we shall embed T' in G so that every vertex is embedded within the cluster to which it is allocated. To begin, by a standard regularity argument we may choose for each i a set $V'_i \subseteq V_i$ so that $|V'_i| \geq 9m/10$ and every vertex $v \in V'_i$ has at least $dm/2$ outneighbours in V'_{i+1}. Let $G' = G[V'_1 \cup \dots \cup V'_k]$. Now, for each i, let S_i be the set of vertices of T' allocated to V_i. So $|S_2|, \dots, |S_k| \leq k-1$ and $|S_1| \leq |T'|$. Then by Proposition 4.3, $3|S_1| \leq 3|T'| \leq 9k|H| \leq |V'_1|$. So by Theorem 1.2 we may embed $T'[S_1]$ in $G[V'_1]$. Now suppose that we have successfully embedded $T'[S_1 \cup \dots \cup S_{i-1}]$ in $G[V'_1 \cup \dots \cup V'_{i-1}]$ for some $i \leq j$. Then precisely one vertex $t \in S_i$ has a neighbour $t' \in S_{i-1}$, and t' has already been embedded to some $v' \in V'_{i-1}$. Now v' has at least $dm/2 \geq 3|S_i|$ outneighbours in V'_i, and so by Theorem 1.2 we may embed $T'[S_i]$ among these outneighbours. Let v be the vertex to which t is embedded; then since v is an outneighbour of v', we have extended our embedding to an embedding of $T'[S_1 \cup \dots \cup S_i]$ in $G[V'_1 \cup \dots \cup V'_i]$. Continuing in this manner we obtain an embedding of T' in G, with t'_1 embedded in V_j and $V(T') \setminus \{x_0, \dots, x_{k-2}\} \supseteq V(T') \cap H$ embedded into V_1, as desired. A similar argument will achieve this if $|F| > k-j$. □

4.4. **The extended tree.** The next lemma combines the ideas of the core tree and leading paths to give the structure within a tree T which we shall use to prove Lemma 4.1. It shows that given a tree T we may extend the core tree T_c of T with parameter Δ to an 'extended tree' T_{ext} which, like T_c, has bounded maximum degree (although this bound is now much larger than Δ). T_{ext} will also have the property that only a small subset H of the vertices of T_{ext} have neighbours outside T_{ext}, and that few vertices of T_{ext} are close to a vertex of $\mathcal{P}_k(H)$.

Lemma 4.5. *Suppose that* $1/n, 1/\Delta^* \ll 1/\Delta, 1/k, \omega \ll 1$*. Let* T *be a tree on* n *vertices, and let* T_c *be the core tree of* T *with parameter* Δ*. Choose any vertex* $t_1 \in T_c$ *as the root of* T*. Then there exists a subtree* T_{ext} *of* T *and a subset* $H \subseteq V(T_{ext})$ *which satisfy the following properties.*

(i) $T_c \subseteq T_{ext}$.
(ii) $\Delta(T_{ext}) \leq \Delta^*$.
(iii) *For any edge* e *between* $V(T - T_{ext})$ *and* $V(T_{ext})$*, the endvertex of* e *in* $V(T_{ext})$ *lies in* H*.*
(iv) *The number of vertices* $v \in T_{ext}$ *which satisfy* $1 \leq d(v, \mathcal{P}_k(H)) \leq k^3$ *is at most* ωn*.*

(v) $|H| \leq n/\Delta^{k^{1/\omega}}$.

Proof. We consider the subgraph $T - E(T_c)$ of T obtained by deleting the edges (but not the vertices) of T_c from T. Each vertex $v \in T_c$ lies in a separate component of $T - E(T_c)$; we denote the component containing v by T_v. Then the trees T_c and $\{T_v : v \in T_c\}$ partition the edges of T, and the trees $\{T_v : v \in T_c\}$ partition the vertices of T.

We say that a vertex $v \in T_c$ is *i-heavy* if $|T_v| \geq \Delta_i := \Delta^{k^i}$. For any integer i, let H_i denote the set of i-heavy vertices in T_c. So $|H_i| \leq n/\Delta_i$, and so by Proposition 4.3 we have $|\mathcal{P}_k(H_i)| \leq 3kn/\Delta_i$ for each i. We wish to choose a large integer t so that few vertices of T lie in trees T_v for which v is not in H_t but is close to a member of $\mathcal{P}_k(H_t)$. The next claim shows that this is possible.

Claim. For some natural number $1/\omega \leq t \leq 3/\omega$ we have

$$|\bigcup_{\substack{v \in V(T_c)\setminus H_t \\ d(v, \mathcal{P}_k(H_t)) \leq k^3}} T_v| \leq \omega n. \tag{1}$$

Proof of Claim. Observe that for each integer i with $1/\omega \leq i \leq 3/\omega$, if $v \in V(T_c) \setminus H_{i-1}$ then $|T_v| < \Delta_{i-1}$, and so

$$|\bigcup_{\substack{v \in V(T_c)\setminus H_{i-1} \\ d(v, \mathcal{P}_k(H_i)) \leq k^3}} T_v| < |\mathcal{P}_k(H_i)|\Delta^{k^3+1}\Delta_{i-1} \leq \frac{3k\Delta^{k^3+1}\Delta^{k^{i-1}}n}{\Delta^{k^i}} \leq \frac{3kn}{\Delta^{k^i/2}} \leq \omega n/3.$$

Now let

$$B_i := \bigcup_{\substack{v \in H_{i-1}\setminus H_i \\ d(v, \mathcal{P}_k(H_i)) \leq k^3}} T_v.$$

Then the sets B_i are pairwise disjoint subsets of $V(T)$. If the claim is false, then $|B_i| > 2\omega n/3$ for every integer i with $1/\omega \leq i \leq 3/\omega$, and so $|\bigcup_{1/\omega \leq i \leq 3/\omega} B_i| > n$, giving a contradiction. □

Fix such a value of t, and let $H = H_t$. We define the extended tree T_{ext} by $T_{\text{ext}} := T_c \cup \bigcup_{v \in V(T_c)\setminus H} T_v$. Then T_{ext} is a subtree of T with $T_c \subseteq T_{\text{ext}}$, so (i) is satisfied. Since $H \subseteq V(T_c)$, we have $H \subseteq V(T_{\text{ext}})$ as desired. Also (ii) holds since any vertex $u \in T_{\text{ext}}$ has at most Δ neighbours in T_c and at most Δ_t neighbours in the single tree T_v with $v \in T_c$ which contains u. So $\Delta(T_{\text{ext}}) \leq \Delta + \Delta_t \leq \Delta + \Delta^{k^{3/\omega}} \leq \Delta^*$. For (iii), observe that if $u \notin T_{\text{ext}}$, then u must lie in some T_v with $v \in H$. But then if u has a neighbour in T_{ext} this neighbour must be v. For (iv), consider any $u \in T_{\text{ext}}$ satisfying $1 \leq d(u, \mathcal{P}_k(H)) \leq k^3$. Since $d(u, \mathcal{P}_k(H)) \geq 1$ we know that $u \notin H_t$, so if $u \in T_c$ then u is counted in (1). If $u \notin T_c$ then there exists v such that $u \in T_v$ and $v \in V(T_c) \setminus H$. Note that $\mathcal{P}_k(H) \subseteq V(T_c)$ (since $t_1 \in T_c$). This in turn implies that $d(v, \mathcal{P}_k(H)) < d(u, \mathcal{P}_k(H)) \leq k^3$. So u is also counted in (1). Finally, for (v), recall that $|H| \leq n/\Delta_t \leq n/\Delta^{k^{1/\omega}}$. □

4.5. **Embedding trees of bounded maximum degree with restrictions.** In this section we shall prove the following lemma, which is similar to Lemma 3.2, but which allows us to restrict some vertices of T to a subset of $V(G)$.

Lemma 4.6. *Suppose that $1/n \ll 1/\Delta, 1/k \ll \varepsilon \ll d \ll \alpha, \lambda \leq 1/2$, that $m = n/k$, that $\lambda \leq \alpha/4$ and that $\delta = d\lambda/8k$. Let T be a directed tree on n vertices rooted at t_1 and with $\Delta(T) \leq \Delta$. Let $H \subseteq V(T)$ be such that $|H| \leq \delta n/7k$ and $|\{x \in T \colon 1 \leq d(x, \mathcal{P}_k(H)) \leq k^3\}| \leq \delta n$. Let G be an ε-regular d-dense cycle of cluster tournaments on clusters $V_1, \dots, V_k$, each of size $(1+\alpha)m$, and let $U \subseteq V_1 \cup \dots \cup V_k$ have size $|U| \geq \lambda n$. Then T can be embedded in G so that each vertex $t \in H$ is embedded to some $u \in U$.*

Proof. We may assume without loss of generality that $|U \cap V_1| \geq \lambda m$. If $t_1 \notin H$, then add t_1 to H, so now we have $|H| \leq \delta n/6k$. Moreover, the new $\mathcal{P}_k(H)$ is the union of the old $\mathcal{P}_k(H)$ and $\{t_1\}$. So now

$$|\{x \in T \colon 1 \leq d(x, \mathcal{P}_k(H)) \leq k^3\}| \leq \delta n + \Delta^{k^3+1} \leq 3\delta n/2. \tag{2}$$

Also, introduce a new constant ε' with $\varepsilon \ll \varepsilon' \ll d$. To begin, for each i choose disjoint sets $X_i, Y_i \subseteq V_i$ such that

- $|X_i| = (1 + \alpha/2)m$ and $|Y_i| = 3\lambda m/4 \leq \alpha m/4$,
- every vertex of $X_i \cup Y_i$ has at least $d\lambda m/2$ inneighbours in Y_{i-1} and at least $d\lambda m/2$ outneighbours in Y_{i+1}, and
- $Y_1 \subseteq U \cap V_1$.

The existence of such sets can be shown by a standard regularity argument. Indeed, choose disjoint sets $X'_i, Y'_i \subseteq V_i$ such that $|X'_i| = (1+\alpha/2+d^2)m$, $|Y'_i| = (3\lambda/4+d^2)m$ and $Y'_1 \subseteq U \cap V_1$. Then both $G[Y'_{i-1} \to X'_i \cup Y'_i]$ and $G[X'_i \cup Y'_i \to Y'_{i+1}]$ are $2\varepsilon/\lambda$-regular with density at least $3d/4$. So all but at most $9\varepsilon m/\lambda \leq d^2 m$ vertices in $X'_i \cup Y'_i$ have at least $9d\lambda m/16$ inneighbours in Y'_{i-1} and at least $9d\lambda m/16$ outneighbours in Y'_{i+1}. Delete $d^2 m$ vertices from X'_i and $d^2 m$ vertices from Y'_i including these $d^2 m$ vertices of small degree (for each $i \in [k]$). Then the sets X_i and Y_i thus obtained from X'_i and Y'_i are as desired.

Each vertex of $\mathcal{P}_k(H)$, and every child of any such vertex, will be embedded in the sets Y_i, whilst the remaining vertices of T will be embedded in the sets X_i. Observe that by Proposition 4.3, $|\mathcal{P}_k(H)| \leq 3k|H| \leq \delta n/2$. Moreover, (2) implies that there are at most $3\delta n/2$ children of vertices of $\mathcal{P}_k(H)$ outside $\mathcal{P}_k(H)$. So at most $2\delta n = d\lambda m/4$ vertices will be embedded in the sets Y_i.

Next, let $T_1, \dots, T_r$ be the component subtrees of $T[\mathcal{P}_k(H)]$ and $T - \mathcal{P}_k(H)$. So each vertex of T lies in precisely one of the T_i. Let T^{con} be the tree obtained by contracting each T_i to a single vertex i. We may assume the T_i were labelled so that $t_1 \in T_1$ and $1, 2, \dots, r$ is an ancestral order of the vertices of T^{con}. Then let

$$\begin{aligned} J &= \{i : T_i \text{ is a component subtree of } T[\mathcal{P}_k(H)]\}, \\ L &= \{i : T_i \text{ is a component subtree of } T - \mathcal{P}_k(H) \text{ and } |T_i| \geq \sqrt{n}\}, \\ Q &= \{i : T_i \text{ is a component subtree of } T - \mathcal{P}_k(H) \text{ and } |T_i| < \sqrt{n}\}. \end{aligned}$$

Note that each vertex of H lies in some T_i such that $i \in J$. For each $i > 1$, T_i contains precisely one vertex with a neighbour in some T_j with $j < i$. (Furthermore, if $i \in L \cup Q$ then this j must belong to J.) Let t_i be this vertex, then the children of vertices of $\mathcal{P}_k(H)$ which are not in $\mathcal{P}_k(H)$ are precisely the vertices t_i for $i \in L \cup Q$. For each i let T_i^{far} be the set of vertices $x \in T_i$ with $d(t_i, x) \geq k^3$. Then

$$\sum_{i \in L \cup Q} |V(T_i) \setminus T_i^{\mathrm{far}}| \leq 3\delta n/2 \tag{3}$$

by (2). Finally, for each i let $T_i^{\leq} = T[V(T_1) \cup \dots \cup V(T_i)]$, so $T_i^{\leq}$ is the graph formed from the union of $T_1, \dots, T_i$ by also adding the edges between $T_1, \dots, T_i$.

We shall use a randomised algorithm to embed the vertices of T in G. At each time τ this algorithm will embed the vertices of T_τ. Indeed, if $\tau \in J$, we will use Lemma 4.4 to embed T_τ in the sets Y_i so that the vertices of $H \cap V(T_\tau)$ are embedded in $Y_1 \subseteq U$. If $\tau \in L$, we will use Lemma 3.2 to embed T_τ in the sets X_i (except for the vertex t_τ, which will be embedded in some Y_i) so that approximately equally many vertices of T_τ are embedded in each set X_i. Finally, if $\tau \in Q$ we will use Lemma 3.5 to randomly embed T_τ in the sets X_i (again with the exception of the vertex t_τ, which will be embedded in some Y_i) so that the expected number of vertices of T_τ^{far} embedded in each set X_i is approximately equal. Together the embeddings of each T_i in G will form an embedding of T in G such that every vertex of H is embedded in U, as desired. At any time τ we will be able to choose the desired embedding of T_τ unless there are insufficient vertices remaining unoccupied in one of the sets X_i. We shall show that this is unlikely to happen for any i, and hence that with positive probability the algorithm will find a copy of T in G, proving the lemma.

Tree Embedding Algorithm.

At time $\tau = 1$, we wish to embed T_1 in G. Recall that we ensured that $t_1 \in H$, so $1 \in J$. We shall embed T_1 in Y_1. Indeed, $|Y_1| = 3\lambda m/4$, and $|T_1| \le |\mathcal{P}_k(H)| \le \delta n/2 = d\lambda m/16$, and so Y_1 contains a copy of T_1 by Theorem 1.2. Choose (deterministically) such a copy, and embed each vertex of T_1 to the corresponding vertex in this copy.

So after completing the first step, the algorithm will have obtained an embedding of $T_1 = T_1^{\le}$ in G such that any vertex of $H \cap V(T_1^{\le})$ is embedded in Y_1, and only vertices of $\mathcal{P}_k(H)$ and their children have been embedded in the sets Y_i.

At a given time $\tau > 1$ we may therefore suppose that the algorithm has found an embedding of $T_{\tau-1}^{\le}$ in G so that each vertex of $H \cap V(T_{\tau-1}^{\le})$ is embedded in Y_1, and only vertices of $\mathcal{P}_k(H)$ and their children have been embedded in the sets Y_i. (Recall that this implies that at most $d\lambda m/4$ vertices are embedded in the sets Y_i.) We wish to extend this embedding to include T_τ, and we do this by the following steps.

- For each i let X_i^τ and Y_i^τ consist of the unoccupied vertices of X_i and Y_i respectively. If $|X_i^\tau| < |T_\tau|/k + \alpha m/4$ for some i, then terminate the algorithm with failure. So we may assume that $|X_i^\tau| \ge |T_\tau|/k + \alpha m/4$ for each i. Also, since at most $d\lambda m/4$ vertices have been embedded in the sets Y_i, every vertex of $X_i \cup Y_i$ must have at least $d\lambda m/4$ inneighbours in Y_{i-1}^τ and at least $d\lambda m/4$ outneighbours in Y_{i+1}^τ.
- By definition, t_τ is the unique vertex of T_τ with a neighbour which has already been embedded. Let t'_τ be this neighbour, and let v'_τ be the vertex to which t'_τ was embedded. Also let V_j be the cluster into which t_τ should be embedded so that the edge between t_τ and t'_τ is embedded canonically. Then v'_τ has at least $d\lambda m/4$ neighbours in Y_j^τ, and so by a standard regularity argument, we may choose some such neighbour $v_\tau \in Y_j^\tau$ which has at least $\alpha dm/8$ outneighbours in X_{j+1}^τ and at least $\alpha dm/8$ inneighbours in X_{j-1}^τ.
- Now, if $\tau \in L$, for each i consider a set $Z_i^\tau \subseteq X_i^\tau$ of size $(1+\alpha/8)|T_\tau|/k$ chosen uniformly at random and independently of all other choices. We can do this since $(1+\alpha/8)|T_\tau|/k \le |T_\tau|/k + \alpha m/8 \le |X_i^\tau|$ for each $i \in [k]$. Then since $G[X_1^\tau \cup \dots X_k^\tau]$ is a $(16\varepsilon/\alpha)$-regular $d/2$-dense cycle of cluster tournaments, by Lemma 2.6 $G[Z_1^\tau, \dots, Z_k^\tau]$ is an ε'-regular $d/4$-dense cycle of cluster tournaments with probability $1 - o(1)$. Also with probability $1 - o(1)$, v_τ has at least $\alpha d|T_\tau|/16k$ outneighbours in Z_{j+1}^τ and at least $\alpha d|T_\tau|/16k$ inneighbours in Z_{j-1}^τ. So we may choose (deterministically) sets Z_i^τ satisfying these two properties. Now delete a single vertex (chosen arbitrarily) from Z_j^τ, and replace it by v_τ, and let G^τ be the restriction of G to the new $Z_1^\tau, \dots, Z_k^\tau$. Then G^τ is a $(2\varepsilon')$-regular $(d/8)$-dense cycle of cluster tournaments with clusters

of size $(1+\alpha/8)|T_\tau|/k$. So by Lemma 3.2 G^τ contains a copy of T_τ with at most $(1+\alpha/8)|T_\tau|/k$ vertices of T_τ embedded in each X_i, and with t_τ embedded to v_τ. Embed each vertex of T_τ to the corresponding vertex in this copy.

- If instead $\tau \in Q$, then arbitrarily choose $Z^\tau_j \subseteq X^\tau_j \cup \{v_\tau\}$ of size $\alpha m/8$ with $v_\tau \in Z^\tau_j$, and for each $i \neq j$ choose $Z^\tau_i \subseteq X^\tau_i$ of size $\alpha m/8$ uniformly at random and independently of all other choices. Then $G^\tau := G[Z^\tau_1, \dots, Z^\tau_k]$ is a $(16\varepsilon/\alpha)$-regular $d/2$-dense cycle of cluster tournaments. Also, with probability $1-o(1)$, v_τ has at least $\alpha^2 dm/128$ outneighbours in Z^τ_{j+1} and at least $\alpha^2 dm/128$ inneighbours in Z^τ_{j-1}, so we may fix (deterministically) our choices of the Z^τ_i such that this event holds. Then by Lemma 3.5 the set of copies of T_τ in G^τ such that t_τ is embedded to v_τ is non-empty, and furthermore there exists a probability distribution on this set so that if a copy is chosen according to this distribution, then the expected number of vertices of T^{far}_τ embedded in each Z^τ_i is at most $(1+\sqrt{\varepsilon})|T^{\text{far}}_\tau|/k$. Choose (deterministically) such a distribution, and choose randomly such a copy according to this distribution. Embed each vertex of T_τ to the corresponding vertex in this copy.
- Finally, if $\tau \in J$, then since v'_τ has at least $d\lambda m/4$ neighbours in Y^τ_j, we may choose sets $Z^\tau_1 \subseteq Y^\tau_1, \dots, Z^\tau_k \subseteq Y^\tau_k$, each of size $d\lambda m/4$, so that every vertex of Z^τ_j is a neighbour of v'_τ. Let G^τ be the restriction of G to the sets Z^τ_i; then G^τ is a $(8\varepsilon/d\lambda)$-regular $(d/2)$-dense cycle of cluster tournaments. Since $|H| \leq \delta n/6k = d\lambda m/48k$, by Lemma 4.4, G^τ contains a copy of T_τ, with vertex t_τ embedded in Y^τ_j, and with every vertex of $H \cap V(T_\tau)$ corresponding to a vertex of Y^τ_1. Embed each vertex of T_τ to the corresponding vertex in this copy.
- In either case, we have extended the embedding of $T^{\leq}_{\tau-1}$ in G to an embedding of $T^{\leq}_\tau$ in G, such that every vertex of $H \cap V(T^{\leq}_\tau)$ is embedded in $Y_1 \subseteq U$, and only vertices of $\mathcal{P}_k(H)$ and their children have been embedded in the sets Y_i.

Since $T^{\leq}_r = T$, if the algorithm does not terminate with failure then at time r, after embedding T_r it will have obtained an embedding of T in G so that every vertex of H is embedded in U, as desired. At this point the algorithm terminates with success.

It remains to show that with positive probability this algorithm will not terminate with failure before embedding T_r. Suppose first that $\sum_{j\in Q}|T_j| < \alpha m/8$. Then for any $i \in [k]$ and at any time τ, the number of vertices embedded in X_i is at most

$$\frac{1+\alpha/8}{k}\sum_{\substack{j\in L\\ j<\tau}}|T_j| + \sum_{\substack{j\in Q\\ j<\tau}}|T_j| \leq \frac{(1+\alpha/8)(n-|T_\tau|)}{k} + \frac{\alpha m}{8} < \left(1+\frac{\alpha}{4}\right)m - \frac{|T_\tau|}{k}$$

and so $|X^\tau_i| \geq |T_\tau|/k + \alpha m/4$. Therefore the algorithm cannot terminate with failure at any point. So we may assume that $\sum_{j\in Q}|T_j| \geq \alpha m/8$.

Let OUT be the set of all possible courses of the algorithm until termination. Since the only random choices made by the algorithm are the choices of where to embed the T_i for each $i \in Q$, any possible course of the algorithm $\mathcal{C} \in OUT$ can be uniquely described by the embeddings f_i of T_i into G for each $i \in Q$ such that the algorithm does not terminate before embedding T_i. So we may define a probability space with sample space OUT where for any $\mathcal{C} \in OUT$, $\mathbb{P}(\mathcal{C})$ is defined to be the probability that the algorithm takes course $\mathcal{C}$. So

$$\mathbb{P}(\mathcal{C}) = \prod_{j\in Q}\mathbb{P}(F_j \mid F_i\colon i<j, i\in Q).$$

where F_j denotes the event that f_j is the embedding of T_j into G, if T_j is embedded at some point during $\mathcal{C}$, and is taken to be true otherwise.

Now, we define the random variable W_j^i in this probability space as follows. For any $\mathcal{C} \in OUT, j \in Q$ and $i \in [k]$, let

$$W_j^i(\mathcal{C}) = \begin{cases} \frac{\#\text{ of vertices from } T_j^{\text{far}} \text{ embedded in } X_i}{\sqrt{n}} & \text{if } T_j \text{ is embedded during } \mathcal{C}, \\ \frac{|T_j^{\text{far}}|}{k\sqrt{n}} & \text{otherwise.} \end{cases}$$

Since $|T_j^{\text{far}}| \leq |T_j| < \sqrt{n}$ for each $j \in Q$, W_j^i is a well-defined function from OUT to $[0,1]$, and so is a well-defined random variable in our probability space.

For any $j \in Q$ and $\mathcal{C}_a, \mathcal{C}_b \in OUT$, let $\mathcal{C}_a \sim_j \mathcal{C}_b$ if and only if $\mathcal{C}_a$ and $\mathcal{C}_b$ share the same course before time j (i.e. they embed $T_1, \dots, T_{j-1}$ identically) or T_j is not embedded at any point in either $\mathcal{C}_a$ or $\mathcal{C}_b$. Then $\sim_j$ is an equivalence relation on OUT (since if two courses agree up to time $j-1$, then at time j either they both terminate with failure or they both successfully embed T_j). For any equivalence class $\mathcal{C}^*$ of $\sim_j$ other than the class of $\mathcal{C}$ for which T_j is not embedded, every $\mathcal{C} \in \mathcal{C}^*$ shares the same course before time j. So for each $\mathcal{C} \in \mathcal{C}^*$, the same probability distribution on the set of copies of T_j will have been chosen at time j, and a copy will then have been chosen according to this distribution. So further partition $\mathcal{C}^*$ into $\mathcal{C}_1^*, \dots, \mathcal{C}_a^*$ by this choice, so courses $\mathcal{C}, \mathcal{C}' \in \mathcal{C}^*$ are in the same $\mathcal{C}_s^*$ if and only if T_j is embedded identically in $\mathcal{C}$ and $\mathcal{C}'$. Now

$$\mathbb{E}(W_j^i \mid \mathcal{C}^*) = \sum_s \mathbb{E}(W_j^i \mid \mathcal{C}_s^*)\mathbb{P}(\mathcal{C}_s^* \mid \mathcal{C}^*),$$

but every member of $\mathcal{C}_s^*$ embeds T_j identically, so $\mathbb{E}(W_j^i \mid \mathcal{C}_s^*)$ is simply the number of vertices of T_j^{far} embedded in X_i in this common embedding, divided by $\sqrt{n}$. Also, $\mathbb{P}(\mathcal{C}_s^* \mid \mathcal{C}^*)$ is the probability that this embedding of T_j is chosen when the random choice of the embedding of T_j is made. So by our (deterministic) choice of the probability distribution on the copies of T_j in G,

$$\mathbb{E}(W_j^i \mid \mathcal{C}^*) \leq \frac{(1+\sqrt{\varepsilon})|T_j^{\text{far}}|}{k\sqrt{n}}. \tag{4}$$

If instead $\mathcal{C}^*$ is the class of all $\mathcal{C}$ such that T_j is not embedded in $\mathcal{C}$, then $\mathbb{E}(W_j^i \mid \mathcal{C}^*) = |T_j^{\text{far}}|/k\sqrt{n}$ by definition, and so (4) holds in this case also.

Now, for any equivalence class $\mathcal{C}^*$ other than the class in which T_j is not embedded, the embeddings of $T_1, \dots, T_{j-1}$ are identical amongst the members of $\mathcal{C}^*$, and so

$$\mathbb{E}(W_j^i \mid \mathcal{C}^*, W_s^i \colon s \in Q, s < j) = \mathbb{E}(W_j^i \mid \mathcal{C}^*).$$

Clearly this equality also holds for the class $\mathcal{C}^*$ in which T_j is not embedded, and so for any $i \in [k]$,

$$\mathbb{E}(W_j^i \mid W_s^i \colon s \in Q, s < j) \leq \max_{\mathcal{C}^*} \mathbb{E}(W_j^i \mid \mathcal{C}^*, W_s^i \colon s \in Q, s < j) \leq \frac{(1+\sqrt{\varepsilon})|T_j^{\text{far}}|}{k\sqrt{n}}.$$

Since $\sum_{j \in Q} |T_j| \geq \alpha m/8$, by Lemma 2.2, for any i the probability that

$$\sum_{j \in Q} W_j^i \leq \frac{(1+\alpha/8)\sum_{j \in Q}|T_j|}{k\sqrt{n}} \tag{5}$$

does not hold decreases exponentially with n. So with probability $1-o(1)$, (5) holds for each $i \in [k]$.

To finish the proof, we show that if (5) holds for each $i \in [k]$, then the algorithm cannot terminate with failure, and will therefore successfully embed T in G as desired. Indeed, the algorithm will only terminate with failure if at some time τ we have $|X_i^\tau| < |T_\tau|/k + \alpha m/4$ for some i. But for any $i \in [k]$ and any time τ, only vertices from subtrees T_s such that $s \in L \cup Q$ and $s < \tau$ have been embedded in X_i before time τ. So the number of vertices embedded in X_i before time τ is at most

$$\frac{(1+\alpha/8)}{k} \sum_{s \in L \setminus \{\tau\}} |T_s| + \sum_{s \in Q \setminus \{\tau\}} |V(T_s) \setminus T_s^{\mathrm{far}}| + \sqrt{n} \sum_{s \in Q \setminus \{\tau\}} W_s^i$$
$$\overset{(3)}{\leq} \frac{(1+\alpha/8)}{k} \sum_{s \in L} |T_s| + \frac{3\delta n}{2} + \sqrt{n} \sum_{s \in Q} W_s^i + \frac{\delta n}{2} - \frac{|T_\tau|}{k}$$
$$\overset{(5)}{\leq} \frac{(1+\alpha/8)}{k} \sum_{s \in L \cup Q} |T_s| + 2\delta n - \frac{|T_\tau|}{k} \leq (1 + \frac{\alpha}{4})m - \frac{|T_\tau|}{k}.$$

To see that the second line holds, note that $|T_\tau|/k < \sqrt{n}/k < \delta n/2$ whenever $\tau \in Q$ and $|T_\tau|/k < |\mathcal{P}_k(H)| < \delta n/2$ whenever $\tau \in J$. So if (5) holds, then at any time τ and for any $i \in [k]$, $|X_i^\tau| \geq |T_\tau|/k + \alpha m/4$, and so the algorithm succeeds. This completes the proof of Lemma 4.6. □

4.6. **Proof of Lemma 4.1.** We can now give the proof of Lemma 4.1, which will proceed as follows. We shall apply Lemma 4.5 to find a subtree T_{ext} of T and a subset $H \subseteq V(T_{\mathrm{ext}})$. Then we shall find a cluster cycle $\mathcal{C}$ in G such that $|\mathcal{C}|$ is slightly larger than $|T_{\mathrm{ext}}|$. We then embed T_{ext} into $\mathcal{C}$ using Lemma 4.6, restricting H to a set U of vertices of $\mathcal{C}$ which have many inneighbours and outneighbours outside $\mathcal{C}$. Finally we shall use this property of U to embed the vertices of $T - T_{\mathrm{ext}}$ in $V(G) \setminus V(\mathcal{C})$ and thereby complete the embedding.

If $\alpha \geq 1/2$, then G contains a copy of T by Theorem 1.2. So we may assume that $\alpha < 1/2$. We begin by introducing new constants $\Delta^*, M, M', \delta, \varepsilon, d$ and Δ with

$$1/n \ll 1/\Delta^* \ll 1/M \ll 1/M' \ll \varepsilon \ll d \ll \mu \ll \nu \ll \eta \ll 1/\Delta \ll \alpha.$$

Then Lemma 2.7 implies that G contains an ε-regular d-dense cycle of cluster tournaments on clusters $V_1, \dots, V_k$, where $M' \leq k \leq M$ and each cluster has equal size between $(2+\alpha)m$ and $(2+2\alpha)m$, where $m = n/k$. Also let

$$\delta := d\alpha^2/16000k.$$

Remove vertices from each V_i to obtain a 2ε-regular $d/2$-dense cycle of cluster tournaments G' on clusters $V_1', \dots, V_k'$ each of size $(2+\alpha)m$.

Let T_c be the core tree of T with parameter Δ, and choose any vertex $t_1 \in T_c$ as the root of T. Then by Lemma 4.5 (applied with $\omega = \delta\alpha/50$), we may choose a subtree T_{ext} of T and a subset $H \subseteq V(T_{\mathrm{ext}})$ satisfying the following properties.

(i) $T_c \subseteq T_{\mathrm{ext}}$.
(ii) $\Delta(T_{\mathrm{ext}}) \leq \Delta^*$.
(iii) For any edge e between $V(T - T_{\mathrm{ext}})$ and $V(T_{\mathrm{ext}})$, the endvertex of e in $V(T_{\mathrm{ext}})$ lies in H.
(iv) The number of vertices $v \in T_{\mathrm{ext}}$ which satisfy $1 \leq d(v, \mathcal{P}_k(H)) \leq k^3$ is at most $\delta\alpha n/50$.
(v) $|H| \leq n/\Delta^{k^{50/\delta\alpha}} \leq \delta\alpha n/350k$.

Let $T_1^+, \dots, T_r^+$ and $T_1^-, \dots, T_s^-$ be the component subtrees of $T - T_{\mathrm{ext}}$. Each T_i^+ and T_i^- will contain precisely one vertex, v_i^+ or v_i^- respectively, with a neighbour in T_{ext}. Label the $T_1^+, \dots, T_r^+$ and $T_1^-, \dots, T_s^-$ so that each T_i^+ contains v_i^+ with an inneighbour in T_{ext}, and each T_i^- contains v_i^- with an outneighbour in T_{ext}. By (i) and Proposition 4.2(iv) each T_i^+

and each T_i^- contains at most n/Δ vertices. Let $x = |T_{\text{ext}}|$, let $y = |T_1^+ \cup \dots \cup T_r^+|$ and let $z = |T_1^- \cup \dots \cup T_s^-|$, so $x + y + z = n$.

Then all but at most $2y+x+\alpha n/2$ vertices of G have at least $y+x/2+\alpha n/4$ outneighbours in G, and all but at most $2z+x+\alpha n/2$ vertices of G have at least $z+x/2+\alpha n/4$ inneighbours in G. So at least $2(1+\alpha)n - 2y - 2z - 2x - \alpha n = \alpha n$ vertices of G satisfy both of these conditions. Choose any $\alpha n/8$ of these vertices to form U_0. Then $|U_0| = \alpha n/8$, and each $v \in U_0$ has at least $y + x/2 + \alpha n/8$ outneighbours outside U_0 and at least $z + x/2 + \alpha n/8$ inneighbours outside U_0.

Suppose first that $x \geq \alpha n/50$. From each cluster V_i' of G' choose a set X_i of $x(1+\alpha/2)/k$ vertices uniformly at random, and let $X = X_1 \cup \dots \cup X_k$. Then $|X| = x(1+\alpha/2)$, and for any single vertex $u \in G'$, the probability that u is included in X is equal to $x/2n$. So by Proposition 2.3, with probability $1-o(1)$ the set $U := X \cap U_0$ satisfies $|U| \geq \alpha x/20 \geq \alpha^2 n/1000$. Also, for any vertex $v \in U$, the expected number of outneighbours of v outside X is at least

$$\left(y + \frac{x}{2} + \frac{\alpha n}{8}\right)\left(1 - \frac{x}{2n}\right) = y + \frac{x}{2} - \frac{xy}{2n} - \frac{x^2}{4n} + \left(1 - \frac{x}{2n}\right)\frac{\alpha n}{8}$$
$$\geq y + x\left(\frac{1}{2} - \frac{y+x}{2n}\right) + \frac{\alpha n}{16} \geq y + \frac{\alpha n}{16}.$$

A similar calculation shows that for each $v \in U$, the expected number of inneighbours of v outside X is at least $z + \alpha n/16$. So by Proposition 2.3 we find that with probability $1-o(1)$, every vertex $v \in U$ has at least $y + \alpha n/20$ outneighbours outside X and at least $z + \alpha n/20$ inneighbours outside X. Fix a choice of X such that both these events of probability $1-o(1)$ occur.

Since every vertex of U has either at least $(2(1+\alpha)n - |X|)/2 \geq y + z + x/2 + \alpha n/2 \geq y + z + \alpha n/2$ inneighbours outside X or at least $y + z + \alpha n/2$ outneighbours outside X, we may choose a set $U' \subseteq U$ of size $|U'| \geq |U|/2 \geq \alpha^2 n/2000$ such that either

(a) every $v \in U'$ has at least $y+\alpha n/20$ outneighbours outside X and at least $y+z+\alpha n/20$ inneighbours outside X, or
(b) every $v \in U'$ has at least $y+z+\alpha n/20$ outneighbours outside X and at least $z+\alpha n/20$ inneighbours outside X.

So $G'[X]$ is a $(150\varepsilon/\alpha)$-regular $(d/2)$-dense cycle of cluster tournaments on clusters $X_1, \dots, X_k$ each of size $(1+\alpha/2)x/k$, and $U' \subseteq X_1 \cup \dots X_k$ has size $|U'| \geq \alpha^2 n/2000 \geq \alpha^2 x/2000$. Also by (ii), (iv) and (v) we know that T_{ext} is a directed tree on x vertices rooted at t_1 and with $\Delta(T_{\text{ext}}) \leq \Delta^*$, and also that $H \subseteq V(T_{\text{ext}})$ satisfies $|H| \leq \delta\alpha n/350k \leq \delta x/7k$ and $|\{t \in T_{\text{ext}} : 1 \leq d(t, \mathcal{P}_k(H)) \leq k^3\}| \leq \delta\alpha n/50 \leq \delta x$. So by Lemma 4.6 (with Δ^* in place of Δ and $\alpha^2/2000$ in place of λ), we may embed T_{ext} in $G'[X]$ so that every vertex of H is embedded to a vertex of U'.

Now suppose instead that $x < \alpha n/50$. Then, since every vertex v of G has either $d^+(v) \geq (1+\alpha)n - 1 \geq y + z + \alpha n$ or $d^-(v) \geq (1+\alpha)n - 1 \geq y + z + \alpha n$, we can choose a set $U' \subseteq U_0$ of size $|U'| \geq \alpha n/16$ which satisfies either (a) or (b) (with $X := U'$). Since $|T_{\text{ext}}| = x < \alpha n/50 \leq |U'|/3$, and $G[U']$ is a tournament, by Theorem 1.2 we may embed T_{ext} in $G[U']$, so in particular every vertex of H is embedded to a vertex of U'.

In either case, let V_{ext} be the set of vertices of G to which T_{ext} is embedded. We may now complete the embedding of T in G. If U' satisfies (a), then we first proceed through the trees T_i^+ in turn. For each T_i^+, let u_i^+ be the inneighbour of v_i^+ in T_{ext} (so $u_i^+ \in H$ by (iii)). Then u_i^+ has been embedded to some vertex $v \in U'$. This $v \in U'$ has at least $y + \alpha n/20$ outneighbours outside V_{ext}, of which at most y have been used for embedding the trees T_j^+

for $j < i$. So there are at least $\alpha n/20$ outneighbours of v outside V_{ext} available to embed T_i^+, and so since $|T_i^+| \leq n/\Delta \leq \alpha n/60$, by Theorem 1.2 we can embed T_i^+ among these vertices. In this way we may embed each of the T_i^+. We then proceed through the T_i^- similarly. For each T_i^- let u_i^- be the inneighbour of v_i^- in T_{ext} (so $u_i^- \in H$ by (iii)). Then u_i^- has been embedded to some vertex $v \in U'$. This $v \in U'$ has at least $y + z + \alpha n/20$ inneighbours outside V_{ext}, of which at most $y + z$ have been used for embedding the trees $T_1^+, \dots, T_r^+$ and the trees T_j^- for $j < i$. So there are at least $\alpha n/20$ inneighbours of v outside V_{ext} available to embed T_i^-, and so since $|T_i^-| \leq n/\Delta \leq \alpha n/60$, again by Theorem 1.2 we can embed T_i^- among these vertices. If U' satisfies (b) we can embed T similarly, first embedding the T_i^-, and then the T_i^+. Either way we have completed the embedding of T in G. □

5. Embedding trees in an almost-transitive tournament.

A *transitive tournament* is a tournament in which the vertices can be given a total order so that every edge is directed towards the endvertex which is greater in this order. It is easy to show that any transitive tournament G on n vertices contains any directed tree T on n vertices, by first showing that the vertices of T can be given a total order so that every edge is directed towards the endvertex which is greater in this order, and then embedding each vertex of T to the vertex of G in the corresponding position (in the order of vertices of G).

In this section, we shall prove an approximate version of this result, namely that if a tournament on $(1 + \alpha)n$ vertices (for some small α) is sufficiently close to being transitive, then it contains any directed tree on n vertices. To state this lemma precisely, we say that a tournament G on n vertices is *ε-almost-transitive* if the vertices of G can be given an order $v_1, \dots, v_n$ so that at most εn^2 edges are directed against the ordering of the vertices, that is, they are directed from v_i to v_j where $i > j$.

The proof of this lemma is by a similar method to the proof of Theorem 1.4 in the next section. The approach is that if the lemma is false, then there is some $\alpha > 0$ for which the lemma does not hold, and so the infimum a_{inf} of all α for which the lemma does hold is greater than zero. We then choose α slightly less than a_{inf} and apply (to a smaller subtree) the fact that the lemma holds for any $\alpha' > a_{\text{inf}}$ to show that the lemma holds for α, giving a contradiction.

Lemma 5.1. *For all $\alpha > 0$ there exists $\varepsilon_0 > 0$ and $n_0 \in \mathbb{N}$ such that for any $\varepsilon \leq \varepsilon_0$ and any $n \geq n_0$, any ε-almost-transitive tournament G on at least $(1 + \alpha)n$ vertices contains any directed tree T on n vertices.*

Proof. We consider the set A of all positive values of α such that the lemma holds. More precisely, A is the set of all positive values of α such that there exist $\varepsilon_0 > 0$ and $n_0 \in \mathbb{N}$ so that for any $n \geq n_0$ and $\varepsilon \leq \varepsilon_0$, any ε-almost-transitive tournament G on at least $(1 + \alpha)n$ vertices contains a copy of any directed tree T on n vertices. So if $\alpha' \in A$ and $\alpha'' > \alpha'$ then $\alpha'' \in A$. Also $2 \in A$ by Theorem 1.2, and so we may define $a_{\text{inf}} = \inf A$, with $0 \leq a_{\text{inf}} \leq 2$. Then for any $\alpha' > a_{\text{inf}}$, $\alpha' \in A$. With this definition the lemma is equivalent to the statement that $a_{\text{inf}} = 0$, so suppose for a contradiction that $a_{\text{inf}} > 0$. Let

$$\gamma \ll 1/\Delta \ll a_{\text{inf}} \qquad \text{and} \qquad \alpha = a_{\text{inf}} - \gamma,$$

so we may assume that $1/\Delta \ll \alpha$. Then $\alpha + 2\gamma > a_{\text{inf}}$, so $\alpha + 2\gamma \in A$, and so by definition of A there exist $\varepsilon_0' > 0$ and $n_0' \in \mathbb{N}$ such that for any $\varepsilon' \leq \varepsilon_0'$ and $n' \geq n_0'$, any ε'-almost-transitive tournament G on at least $(1 + \alpha + 2\gamma)n'$ vertices contains a copy of any directed tree T on n' vertices. Moreover, we may assume that $\varepsilon_0' \ll \gamma$. Fix such an ε_0' and n_0', and let

$1/n_0 \ll 1/n_0', \gamma$ and $\varepsilon_0 \ll \varepsilon_0'$. We will show that for any $n \geq n_0$ and $\varepsilon \leq \varepsilon_0$, any ε-almost-transitive tournament G on at least $(1+\alpha)n$ vertices contains a copy of any directed tree T on n vertices. It then follows that $\alpha \in A$, yielding a contradiction and therefore proving the lemma.

So let $\varepsilon \leq \varepsilon_0$ and $n \geq n_0$, let G be an ε-almost-transitive tournament on at least $(1+\alpha)n$ vertices and let T be a directed tree on n vertices. If $|G| \geq 3n$, then G contains a a copy of T by Theorem 1.2, and so we may assume that $|G| < 3n$. Since G is ε-almost-transitive, we may order the vertices of G as $v_1, \dots, v_{|G|}$ so that at most $\varepsilon|G|^2 \leq 9\varepsilon n^2$ edges are directed from v_j to v_i where $i < j$. Now, at most $18\sqrt{\varepsilon}n$ vertices of G are incident to more than $\sqrt{\varepsilon}n$ such edges; let G' be the subgraph of G obtained by deleting these vertices from G, and let $v_1', v_2', \dots, v_{|G'|}'$ be the vertices of G' in the inherited order. Then G' is a tournament on at least $(1+\alpha-18\sqrt{\varepsilon})n$ vertices such that for any vertex v_i' there are at most $\sqrt{\varepsilon}n$ vertices v_j' for which the edge between v_i' and v_j' is directed towards $v_{\min\{i,j\}}'$.

Next, let T_c be the core tree of T with parameter Δ, as defined in Section 4.2. We consider three possibilities for T_c, in each case showing that T can be embedded in G'.

Case 1: Some vertex $t \in T_c$ has $d^+_{T_c}(t) \geq 2$. Then let F^- be the (possibly empty) forest consisting of each component subtree T' of $T - t$ such that the edge between T' and t is directed towards t. Similarly let the component subtrees T'' of $T - t$ such that the edge between T'' and t is directed away from t be partitioned into two forests, F_1^+ and F_2^+. Since $d^+_{T_c}(t) \geq 2$, by Proposition 4.2(ii) at least two such component subtrees each contain at least n/Δ vertices, and so we may choose F_1^+ and F_2^+ so that $|F_1^+|, |F_2^+| \geq n/\Delta$. Note that $|F^-| = w^-(t)$, and $|F_1^+| + |F_2^+| = w^+(t)$, so in particular $w^+(t) \geq 2n/\Delta$, and also recall that $w^+(t) + w^-(t) = n - 1$.

We first determine where to embed the vertex t. For this, let

$$p := \begin{cases} 3\gamma n + \sqrt{\varepsilon}n + 1 & \text{if } w^-(t) < \gamma n, \\ (1+\alpha+2\gamma)w^-(t) + \sqrt{\varepsilon}n + 1 & \text{if } w^-(t) \geq \gamma n. \end{cases}$$

and embed t to the vertex v_p' of G'. This can be done, as we shall see later that $p < |G'|$. We will embed F^- in the vertices preceding v_p' and F_1^+, F_2^+ in the vertices succeeding v_p' in the vertex ordering of G'. Embedding F^- will be possible because p is a little larger than one might expect, whereas embedding F_1^+ and F_2^+ can be done successively, which will give us enough room for both. Let $S^- = N^-(v_p') \cap \{v_1', \dots, v_{p-1}'\}$, and $S^+ = N^+(v_p') \cap \{v_{p+1}', \dots, v_{|G'|}'\}$. Then S^- and S^+ are disjoint, $|S^-| \geq p - \sqrt{\varepsilon}n - 1$ and $|S^+| \geq |G'| - p - \sqrt{\varepsilon}n$. Next we shall embed F^- in $G'[S^-]$. Indeed, if $w^-(t) < \gamma n$ then $|S^-| \geq 3\gamma n$, and so by Theorem 1.2 we can embed F^- in $G'[S^-]$. Alternatively, if $w^-(t) \geq \gamma n$, let $n' = w^-(t) \geq n_0'$ and $\varepsilon' = |G|^2\varepsilon/(n')^2 \leq \varepsilon_0'$, then F^- is a forest on n' vertices, and $G'[S^-]$ is an ε'-almost-transitive tournament on at least $(1+\alpha+2\gamma)n'$ vertices. So by the choice of ε_0' and n_0' we can embed F^- in $G'[S^-]$.

Finally we shall complete the embedding of T in G' by embedding F_1^+ and F_2^+ in $G'[S^+]$. Now,

$$\begin{aligned} |S^+| &\geq |G'| - p - \sqrt{\varepsilon}n \\ &\geq (1+\alpha-18\sqrt{\varepsilon})n - (3\gamma n + (1+\alpha+2\gamma)w^-(t) + \sqrt{\varepsilon}n + 1) - \sqrt{\varepsilon}n \\ &\geq (1+\alpha)w^+(t) - 5\gamma n - 20\sqrt{\varepsilon}n \geq (1+\alpha)w^+(t) - 6\gamma n. \end{aligned}$$

Let $n' = |F_1^+|$, so $n_0' \leq n/\Delta \leq n'$ and $n' \leq w^+(t) - n/\Delta$, and again let $\varepsilon' = |G|^2\varepsilon/(n')^2$, so $\varepsilon' \leq \varepsilon_0'$. Then $G'[S^+]$ is an ε'-almost-transitive tournament on $|S^+| \geq (1+\alpha)(n' + n/\Delta) -$

$6\gamma n \geq (1+\alpha+1/\Delta)n' + (\alpha/\Delta - 6\gamma)n \geq (1+\alpha+2\gamma)n'$ vertices, and so by our choice of n'_0 and ε'_0, we may embed F_1^+ in $G'[S^+]$.

Now, let $S^+_{\rm rem}$ consist of the vertices of S^+ not occupied by the vertices of F_1^+. We shall embed F_2^+ in $S^+_{\rm rem}$ in a similar manner. Indeed, we now let $n' = |F_2^+|$, so again $n'_0 \leq n/\Delta \leq n'$, and again let $\varepsilon' = |G|^2\varepsilon/(n')^2 \leq \varepsilon'_0$. Then

$$\begin{aligned}|S^+_{\rm rem}| = |S^+| - |F_1^+| &\geq (1+\alpha)w^+(t) - 6\gamma n - (w^+(t) - |F_2^+|)\\ &= (1+\alpha)n' + \alpha|F_1^+| - 6\gamma n \geq (1+\alpha+2\gamma)n',\end{aligned}$$

so $G'[S^+_{\rm rem}]$ is an ε'-almost-transitive tournament on at least $(1+\alpha+2\gamma)n'$ vertices, and so by our choice of n'_0 and ε'_0, we may embed F_2^+ in $G'[S^+_{\rm rem}]$.

Case 2: Some vertex $t \in T_c$ has $d^-_{T_c}(t) \geq 2$. Then we may embed T in G' by the same method as in Case 1, the main difference being that the roles of outdegrees and outneighbours are switched with those of indegrees and inneighbours.

Case 3: T_c is a directed path (possibly consisting of just a single vertex). Then let $w^+ = w^+(T_c)$ and $w^- = w^-(T_c)$ be as defined in Section 2, and partition the vertices of G' into three sets $S^- = \{v'_1, \dots, v'_{w^-+\alpha n/3}\}$, $S = \{v'_{w^-+\alpha n/3+1}, \dots, v'_{|G'|-w^+-\alpha n/3}\}$ and $S^+ = \{v'_{|G'|-w^+-\alpha n/3+1}, \dots, v'_{|G'|}\}$. Then since $w^+ + w^- + |T_c| = n$, we know that $|S| = |G'| - w^- - w^+ - 2\alpha n/3 \geq |T_c|$. Therefore by Theorem 1.3 we may embed T_c in $G'[S]$. Now, let $T_1^+, \dots, T_r^+$ be the component subtrees of $T - T_c$ such that the edge between T_i^+ and T_c is directed towards T_i^+, and for each i let $t_i^+ \in T_c$ be the vertex of T_c to which this edge is incident, and let v_i^+ be the vertex of G' to which t_i^+ was embedded. Similarly, let $T_1^-, \dots, T_s^-$ be the component subtrees of $T - T_c$ such that the edge between T_i^- and T_c is directed towards T_c, let t_i^- be the vertex of T_c to which this edge is incident, and let v_i^- be the vertex of G' to which t_i^- was embedded. Then every vertex of T lies in T_c or one of the T_i^+ or T_i^-. Furthermore $|T_i^+|, |T_j^-| \leq n/\Delta$ for each i and j by Proposition 4.2(iv).

We shall complete the embedding of T in G' by greedily embedding each T_i^+ in $N^+(v_i^+) \cap S^+$, and each T_i^- in $N^-(v_i^-) \cap S^-$. Indeed, suppose we have already embedded $T_1^+, \dots, T_{i-1}^+$, and we now wish to embed T_i^+. Then

$$|N^+(v_i^+) \cap S^+| \geq |S^+| - \sqrt{\varepsilon}n \geq w^+ + \alpha n/3 - \sqrt{\varepsilon}n \geq w^+ + \alpha n/4.$$

At most w^+ of these vertices have already been occupied by vertices of $T_1^+, \dots, T_{i-1}^+$, and so there remain at least $\alpha n/4$ available vertices in which to embed T_i^+. Since $|T_i^+| \leq n/\Delta \leq \alpha n/12$, we may embed T_i^+ in these available vertices by Theorem 1.2. Continuing in this way we may embed all of the T_i^+, and the T_i^- may be embedded similarly, to give us a copy of T in G'.

Any tree in which every vertex has at most one outneighbour and at most one inneighbour is a directed path. So T_c must fall into at least one of the three cases, and so we can find a copy of T in G', and hence in G, contradicting our assumption that $a_{\rm inf} > 0$. So we must have $a_{\rm inf} = 0$, and so the lemma holds. □

6. Proof of Theorem 1.4

Recall the statement of Theorem 1.4.

Theorem 1.4 *Let $\alpha > 0$. Then the following properties hold.*

(1) *There exists $n_0 \in \mathbb{N}$ such that for any $n \geq n_0$, any tournament G on at least $2(1+\alpha)n$ vertices contains any directed tree T on n vertices.*

(2) *Let* Δ *be any positive integer. Then there exists* $n_0 \in \mathbb{N}$ *such that for any* $n \geq n_0$*, any tournament* G *on at least* $(1+\alpha)n$ *vertices contains any directed tree* T *on* n *vertices with* $\Delta(T) \leq \Delta$.

The proofs of each of the two statements of the theorem are very similar, so to avoid repetition we shall prove the first statement, explaining in *italic* font where the proof of the second statement differs.

6.1. **Partitioning the vertices of** G**.** As in the last section, we consider the set A of all positive values of α such that the theorem holds. So $\alpha' \in A$ if and only if there exists n_0 such that for any $n \geq n_0$, any tournament on at least $2(1+\alpha')n$ vertices contains any tree on n vertices. So if $\alpha' \in A$ and $\alpha'' > \alpha'$ then $\alpha'' \in A$, and also $1/2 \in A$ by Theorem 1.2. Thus we may define $a_{\text{inf}} = \inf A$, and then the theorem is equivalent to the statement that $a_{\text{inf}} = 0$. So suppose $a_{\text{inf}} > 0$, and choose constants

$$1/n_0 \ll 1/n_0' \ll \mu \ll \nu \ll \eta \ll 1/\Delta' \ll \gamma \ll a_{\text{inf}}.$$

Let $\alpha = a_{\text{inf}} - \mu$, so $\alpha \leq 1/2$, and we may assume that $\gamma \ll \alpha$. Then $\alpha + 2\mu \in A$, and so for any $n' \geq n_0'$, any tournament on at least $2(1+\alpha+2\mu)n'$ vertices contains any tree on n' vertices. We shall prove that if $n \geq n_0$, any tournament G on at least $2(1+\alpha)n$ vertices contains any tree on n vertices. This proves that $\alpha \in A$, giving a contradiction to our assumption that $a_{\text{inf}} > 0$, and so proving the theorem.

(For the bounded degree case, fix any value of Δ*, and here* $A = A(\Delta)$ *is defined by* $\alpha' \in A$ *if and only if there exists* n_0 *such that for any* $n \geq n_0$*, any tournament on at least* $(1+\alpha')n$ *vertices contains any tree* T *on* n *vertices with* $\Delta(T) \leq \Delta$*. So if* $\alpha' \in A$ *and* $\alpha'' > \alpha'$ *then* $\alpha'' \in A$*, and also* $2 \in A$ *by Theorem 1.2. Thus we may define* $a_{inf} = \inf A$*; then the theorem is equivalent to the statement that* $a_{inf} = 0$*. So suppose* $a_{inf} > 0$*, and choose constants* $1/n_0 \ll 1/n_0' \ll \mu \ll \nu \ll \eta \ll 1/\Delta' \ll \gamma \ll 1/\Delta, a_{inf}$*. Let* $\alpha = a_{inf} - \mu$*, so* $\alpha < 2$*, and we may assume that* $\gamma \ll \alpha$*. Then* $\alpha + 2\mu \in A$*, so for any* $n' \geq n_0'$*, any tournament on at least* $(1+\alpha+2\mu)n'$ *vertices contains any tree* T *on* n' *vertices with* $\Delta(T) \leq \Delta$*. Using this, we shall prove that if* $n \geq n_0$*, any tournament* G *on at least* $(1+\alpha)n$ *vertices contains any tree* T *on* n *vertices with* $\Delta(T) \leq \Delta$*. This proves that* $\alpha \in A$*, giving a contradiction to our assumption that* $a_{inf} > 0$*, and so proving the theorem.)*

So let G be a tournament on at least $2(1+\alpha)n$ vertices. *(For the bounded degree case, instead let* G *be a tournament on at least* $(1+\alpha)n$ *vertices.)* If $|G| \geq 3n$ then by Theorem 1.2, G contains any directed tree T on n vertices. So we may assume $|G| < 3n$. We shall use an algorithm which keeps track of an ordered family $\mathcal{S}^\tau$ of disjoint subsets of $V(G)$, and a set B^τ of bad edges of G, at each time τ. Initially, let $\mathcal{S}^1 = (V(G))$, and let $B^1 = \emptyset$. Then at time $\tau \geq 1$, we have $\mathcal{S}^\tau = (S_1^\tau, \dots, S_\tau^\tau)$, and the algorithm proceeds as follows.

(1) Let S_ℓ^τ be a largest member of $\mathcal{S}^\tau$. If $|S_\ell^\tau| < \gamma n$, then terminate.
(2) If $G[S_\ell^\tau]$ is a robust (μ, ν)-outexpander with $\delta^0(G[S_\ell^\tau]) \geq \eta n$, then terminate.
(3) If some $v \in S_\ell^\tau$ has $d^+_{G[S_\ell^\tau]}(v) < \eta n$, then let

$$\mathcal{S}^{\tau+1} = (S_1^\tau, \dots, S_{\ell-1}^\tau, S_\ell^\tau \setminus \{v\}, \{v\}, S_{\ell+1}^\tau, \dots, S_\tau^\tau),$$

let $B^{\tau+1} = B^\tau \cup E(\{v\} \to S_\ell^\tau \setminus \{v\})$, and proceed to step (6).
(4) Similarly, if some $v \in S_\ell^\tau$ has $d^-_{G[S_\ell^\tau]}(v) < \eta n$, then let

$$\mathcal{S}^{\tau+1} = (S_1^\tau, \dots, S_{\ell-1}^\tau, \{v\}, S_\ell^\tau \setminus \{v\}, S_{\ell+1}^\tau, \dots, S_\tau^\tau),$$

let $B^{\tau+1} = B^\tau \cup E(S_\ell^\tau \setminus \{v\} \to \{v\})$, and proceed to step (6).

(5) If $G[S_\ell^\tau]$ is not a robust (μ,ν)-outexpander then apply Lemma 2.8 to partition the vertices of S_ℓ^τ into sets S' and S'' such that $\nu|S_\ell^\tau| \leq |S'|, |S''| \leq (1-\nu)|S_\ell^\tau|$ and at most $4\mu|S_\ell^\tau|^2$ edges of $G[S_\ell^\tau]$ are directed from S'' to S'. Then let
$$\mathcal{S}^{\tau+1} = (S_1^\tau, \dots, S_{\ell-1}^\tau, S', S'', S_{\ell+1}^\tau, \dots, S_\tau^\tau)$$
and let $B^{\tau+1} = B^\tau \cup E(S'' \to S')$.

(6) Finally, for each $i \in [\tau+1]$, delete from $S_i^{\tau+1}$ any vertex v which lies in more than $\sqrt{\eta}n$ edges of $B^{\tau+1}$.

At any step, if the algorithm does not terminate at step (1) or (2), then the condition of one of steps (3), (4) and (5) must hold. Therefore at each time τ, either the algorithm terminates or $|\mathcal{S}^\tau|$ increases from τ to $\tau+1$ (in forming $\mathcal{S}^{\tau+1}$) by reducing the size of the largest piece. Therefore the algorithm must terminate at some time $\tau_{\text{end}} \leq |G| \leq 3n$.

Now, at any time τ at which the algorithm does not terminate, the algorithm will split the set S_ℓ^τ in precisely one of steps (3), (4) and (5). We next show that the split in step (5) will occur for at most $3/\gamma\nu$ times $\tau < \tau_{\text{end}}$. This is because any set obtained by a split in step (5) must have size at least $\gamma\nu n$ (since $|S_\ell^\tau| \geq \gamma n$, and the sets S', S'' obtained have $|S'|, |S''| \geq \nu|S_\ell^\tau|$), and so at most $|G|/\gamma\nu n \leq 3/\gamma\nu$ such sets can be obtained.

Next, we show that when the algorithm terminates at time τ_{end}, most vertices lie in one of the sets S_i^τ, or equivalently that only a few vertices have been deleted. To do this, note that at each time $\tau \leq \tau_{\text{end}}$, the number of edges added to form $B^{\tau+1}$ from B^τ is at most ηn if the algorithm carried out the split in step (3) or (4), and at most $4\mu|G|^2 \leq 36\mu n^2$ if the algorithm carried out the split in step (5). Since $\tau_{\text{end}} \leq 3n$, and the split in step (5) is carried out in at most $3/\gamma\nu$ of these steps, the number of bad edges at time τ_{end} is at most $3\eta n^2 + 108\mu n^2/\nu\gamma \leq 4\eta n^2$. Since $B^1 \subseteq \dots \subseteq B^{\tau_{\text{end}}}$, any vertex of G which was ever deleted in step (6) must lie in at least $\sqrt{\eta}n$ edges of $B^{\tau_{\text{end}}}$, and so at most $8\sqrt{\eta}n$ vertices of G can have been deleted in step (6) over the entire course of the algorithm. Let G' be the restriction of G to the undeleted vertices at time τ_{end}, so $G' = G[\bigcup \mathcal{S}^{\tau_{\text{end}}}]$. Then G' is a tournament and
$$|G'| \geq |G| - 8\sqrt{\eta}n. \tag{6}$$

Our approach now depends on whether the algorithm terminated in step (1) or (2). If the algorithm terminated in step (1), then for each $i \in [\tau_{\text{end}}]$ we have $|S_i^{\tau_{\text{end}}}| < \gamma n$. We shall show that in this case G' is 2γ-almost-transitive. Indeed, order the vertices of G' as $v_1, v_2, \dots, v_{|G'|}$ in the same order as in $\mathcal{S}^{\tau_{\text{end}}}$, i.e. beginning with all the vertices of $S_1^{\tau_{\text{end}}}$, then the vertices of $S_2^{\tau_{\text{end}}}$, and so forth. Then any edge $v_j \to v_i$ where $j > i$ either lies in $B^{\tau_{\text{end}}}$ or has both endvertices in the same $S_i^{\tau_{\text{end}}}$. So the total number of such edges is at most
$$4\eta n^2 + \sum_{S \in \mathcal{S}^{\tau_{\text{end}}}} \binom{|S|}{2} \leq 4\eta n^2 + \sum_{S \in \mathcal{S}^{\tau_{\text{end}}}} \frac{\gamma n |S|}{2} \leq 4\eta n^2 + \frac{3\gamma n^2}{2} \leq 2\gamma n^2.$$
Since in both the unbounded degree case and the bounded degree case we have
$$|G'| \geq (1+\alpha/2)n,$$
by (6), G' is indeed 2γ-almost-transitive, and by Lemma 5.1 G' contains a copy of T, which is also a copy of T in G.

6.2. **Partitioning the vertices of T.** We may therefore assume that the algorithm terminated in step (2) at some time τ_{end}; when for some $S_i^{\tau_{\text{end}}}$ with $|S_i^{\tau_{\text{end}}}| \geq \gamma n$, $G[S_i^{\tau_{\text{end}}}]$ is a (μ,ν)-robust outexpander with $\delta^0(G[S_i^{\tau_{\text{end}}}]) \geq \eta n$. For this i, let $S = S_i^{\tau_{\text{end}}}$, let $S^+ = \bigcup_{i<j\leq\tau_{\text{end}}} S_j^{\tau_{\text{end}}}$ and let $S^- = \bigcup_{1\leq j<i} S_j^{\tau_{\text{end}}}$. Then $|S^+ \cup S^- \cup S| = |G'|$. Also, if $u \in S^+$ and $v \in S \cup S^-$ then $u \in S_j^{\tau_{\text{end}}}$, $v \in S_\ell^{\tau_{\text{end}}}$ for some $j > \ell$, and so if $u \to v$ then this edge is in $B^{\tau_{\text{end}}}$. So any vertex

$u \in S^+$ has at most $\sqrt{\eta}n$ outneighbours in $S \cup S^-$, since u was not deleted at any stage of the algorithm. Similarly each vertex of S has at most $\sqrt{\eta}n$ outneighbours in S^- and inneighbours in S^+, and each vertex of S^- has at most $\sqrt{\eta}n$ inneighbours in $S^+ \cup S$. Define β, β^+, β^- by $|S| = \beta|G'|$, $|S^+| = \beta^+|G'|$, and $|S^-| = \beta^-|G'|$, so $\beta + \beta^+ + \beta^- = 1$ and $\beta \geq \gamma n/|G'| \geq \gamma/3$.

Suppose first that β^+ and β^- are both small. More precisely, $\beta^+, \beta^- \leq \alpha\beta^2/20$, and so $\beta \geq 1 - \alpha/10$. Then we shall find a copy of T in $G[S]$ (and therefore in G). Indeed, T is a tree on n vertices, and $G[S]$ is a (μ, ν)-robust outexpander with $\delta^0(G[S]) \geq \eta n$. Furthermore,

$$|S| = \beta|G'| \overset{(6)}{\geq} (2 + 2\alpha - 8\sqrt{\eta})\beta n \geq (2 + \alpha)(1 - \frac{\alpha}{10})n \geq 2(1 + \frac{\alpha}{4})n$$

and so by Lemma 4.1 $G[S]$ (and therefore G) contains a copy of T.

(For the bounded degree case, we have

$$|S| = \beta|G'| \overset{(6)}{\geq} (1 + \alpha - 8\sqrt{\eta})(1 - \alpha/10)n \geq (1 + \alpha/4)n,$$

and so $G[S]$ (and therefore G) contains a copy of T by Lemma 3.1.)

So we may assume that at least one of β^+ and β^- is greater than $\alpha\beta^2/20$, so in particular, $\beta \leq 1 - \alpha\beta^2/20$. We next split the vertices of T according to the values of β^+ and β^-.

Case 1: β^- is large but β^+ is small. More precisely, $\beta^+ \leq \alpha\beta^2/20$ and $\beta^- > \alpha\beta^2/20$. Then we partition the vertex set of T into T^- and T^0, where every edge of T between T^- and T^0 is directed from T^- to T^0, and $|T^-| = \beta^-(1 - \alpha\beta)n$. We can form T^0 greedily by successively removing a sink vertex from T and adding it to T^0. Since $\beta^+ + \beta + \beta^- = 1$,

$$|T^0| = n - |T^-| = \beta n(1 + \alpha - \alpha\beta) + (1 - \alpha\beta)\beta^+ n \leq \beta n(1 + \alpha - \alpha\beta) + \alpha\beta^2 n/20.$$

Case 2: β^+ is large but β^- is small. More precisely, $\beta^- \leq \alpha\beta^2/20$ and $\beta^+ > \alpha\beta^2/20$. Then we similarly partition the vertex set of T into T^0 and T^+, where every edge of T between T^0 and T^+ is directed from T^0 to T^+, and $|T^+| = \beta^+(1 - \alpha\beta)n$. Again $|T^0| = n - |T^+| \leq \beta n(1 + \alpha - \alpha\beta) + \alpha\beta^2 n/20$.

Case 3: β^+ and β^- are both large. More precisely, $\beta^+, \beta^- > \alpha\beta^2/20$. Then we partition the vertex set of T into pieces T^-, T^0 and T^+ such that all edges of T between T^- and T^0 are directed from T^- to T^0, all edges of T between T^0 and T^+ are directed from T^0 to T^+ and all edges of T between T^- and T^+ are directed from T^- to T^+. Also $|T^+| = \beta^+(1 - \alpha\beta)n$ and $|T^-| = \beta^-(1 - \alpha\beta)n$, so $|T^0| = \beta(1 + \alpha - \alpha\beta)n$.

Note that in each of the three cases T^0 satisfies $|T^0| \geq \beta(1 + \alpha - \alpha\beta)n$ and

$$|T^0| \leq \beta(1 + \alpha - \alpha\beta)n + \alpha\beta^2 n/20 \leq \beta(1 + \alpha)n - \frac{\alpha\beta^2 n}{2}. \tag{7}$$

6.3. **Embedding T in G.** Having partitioned the vertices of G' into three sets S, S^+ and S^-, and the tree T into three forests T^+, T^0, T^-, we now complete the proof by embedding T in G, with T^-, T^0 and T^+ embedded in $G[S^-], G[S]$ and $G[S^+]$ respectively. Indeed, the fact that $G[S]$ is a robust (μ, ν)-outexpander will enable us to embed slightly more vertices in $G[S]$ than the βn that would be embedded in $G[S]$ if the vertices of T were distributed proportionately amongst $G[S]$, $G[S^+]$ and $G[S^-]$. This gives us some leeway for embedding T^+ and T^- in $G[S^+]$ and $G[S^-]$ respectively, which by our choice of α is sufficient to successfully complete these embeddings.

So let $T_1^-, \dots, T_x^-$ be the component subtrees of T^-, let $T_1^+, \dots, T_y^+$ be the component subtrees of T^+, and let $T_1, \dots, T_z$ be the component subtrees of T^0. Let the *contracted tree* T_{con} be formed from T by contracting each T_i^+, T_i^- and T_i to a single vertex.

To begin the embedding, we embed into $G[S]$ every T_i satisfying $|T_i| \geq n/\Delta'$. Note that there are at most Δ' such T_i. Also, the union of all such T_i is a forest on at most $|T^0|$ vertices, and the tournament $G[S]$ is a robust (μ, ν)-outexpander on

$$\beta|G'| \overset{(6)}{\geq} \beta(2+2\alpha-8\sqrt{\eta})n \overset{(7)}{\geq} 2\left(1+\frac{\alpha\beta}{10}\right)|T^0| \geq 2(1+\gamma^2)|T^0|$$

vertices with $\delta^0(G[S]) \geq \eta n$, and hence $G[S]$ contains a copy of this forest by Lemma 4.1. *(For the bounded degree case, $|S| \geq (1+\gamma^2)|T^0|$ by a similar calculation, and so $G[S]$ contains a copy of this forest by Lemma 3.1.)*

Now, choose an order of the vertices of $T_{\rm con}$, beginning with the at most Δ' vertices corresponding to the T_i which we have just embedded, and such that any vertex of $T_{\rm con}$ has at most Δ' neighbours preceding it in this order. (To do this, choose one of the Δ' vertices corresponding to the T_i which have already been embedded, and then choose any ancestral ordering of the vertices of $T_{\rm con}$, beginning with the chosen vertex, so every vertex has at most one neighbour preceding it in this order. Now move the remaining $\Delta' - 1$ vertices corresponding to the T_i which have already been embedded to the front of this order; then every vertex gains at most $\Delta' - 1$ preceding neighbours.) We shall proceed through the remaining vertices of $T_{\rm con}$ in this order, at each step embedding the tree T_i, T_i^+ or T_i^- corresponding to the current vertex of $T_{\rm con}$ in the unoccupied vertices of the tournament $G[S], G[S^+]$ or $G[S^-]$ respectively.

So suppose first that the current vertex t^* of $T_{\rm con}$ corresponds to some T_i. Since T_i has not already been embedded, we know that $|T_i| \leq n/\Delta'$. Also, since t^* has at most Δ' neighbours preceding it in $T_{\rm con}$, the vertices of T_i have at most Δ' neighbours outside T_i which have already been embedded. Since T_i is a component of T^0, each of these neighbours of vertices in T_i lies either in T^- (in which case it is an inneighbour) or in T^+ (in which case it is an outneighbour). So let $t_1^-, \dots, t_p^-$ be the vertices in T^- which are inneighbours of some vertex in T_i and which have previously been embedded, and let $v_1^-, \dots, v_p^-$ be the vertices of $G'[S^-]$ to which $t_1^-, \dots, t_p^-$ were embedded. Similarly, let $t_1^+, \dots, t_q^+$ be the vertices in T^+ which are outneighbours of some vertex in T_i and which have previously been embedded, and let $v_1^+, \dots, v_q^+$ be the vertices of $G'[S^+]$ to which $t_1^+, \dots, t_q^+$ were embedded. Finally let S^* be the set of unoccupied vertices in $S \cap N^+(v_1^-, \dots, v_p^-) \cap N^-(v_1^+, \dots, v_q^+)$. Then we wish to embed T_i in S^*. For this, note that

$$\begin{aligned}|S^*| &\geq |S| - (p+q)\sqrt{\eta}n - |T^0| \overset{(7)}{\geq} \beta|G'| - \Delta'\sqrt{\eta}n - (\beta(1+\alpha)n - \alpha\beta^2 n/2)\\ &\overset{(6)}{\geq} \beta n(1+\alpha) - (8+\Delta')\sqrt{\eta}n - \beta(1+\alpha)n + \alpha\beta^2 n/2 \geq \alpha\beta^2 n/3 \geq 3n/\Delta' \geq 3|T_i|.\end{aligned}$$

Note that this calculation is valid for both the bounded degree case and the unbounded degree case, with plenty of room to spare in the unbounded case. So by Theorem 1.2, $G[S^*]$ contains a copy of T_i, to which we embed T_i.

Alternatively, if the current vertex of $T_{\rm con}$ corresponds to some T_i^-, then similarly the vertices of T_i^- have at most Δ' neighbours outside T_i^- which have already been embedded, all of which are outneighbours. As before we let $v_1, \dots, v_r$ be the vertices of $G'[S \cup S^+]$ to which these vertices have been embedded, and let S^* be the set of unoccupied vertices of $S^- \cap N^-(v_1, \dots, v_r)$. Note that at most $|T^-| - |T_i^-|$ vertices of T^- have already been embedded. Since some T_i^- exists we have

$$\begin{aligned}|S^*| &\geq |S^-| - r\sqrt{\eta}n - (|T^-| - |T_i^-|) \overset{(6)}{\geq} \beta^-(2+2\alpha)n - (8+\Delta')\sqrt{\eta}n - \beta^-(1-\alpha\beta)n + |T_i^-|\\ &\geq \beta^-(1+2\alpha+\alpha\beta/2)n + |T_i^-|. \qquad (8)\end{aligned}$$

In the final line we used the fact that $\beta^- \geq \alpha\beta^2/20$ and $\beta \geq \gamma/3$ (so $\eta, 1/\Delta' \ll \gamma, \beta, \beta^-$). So $|S^*| \geq 2(1+\alpha+2\mu)|T_i^-|$. Therefore if $|T_i^-| \geq \beta^- n/2$, then $|T_i^-| \geq \alpha\beta^2 n/40 \geq \alpha\gamma^2 n/360 \geq n_0'$, and so we can embed T_i^- in $G[S^*]$ by our choice of n_0'. On the other hand, if $|T_i^-| < \beta^- n/2$ then $|S^*| \geq 3|T_i^-|$ by (8), and so we can embed T_i^- in $G[S^*]$ by Theorem 1.2.

(For the bounded degree case, observe that

$$\begin{aligned} |S^*| &\geq |S^-| - r\sqrt{\eta}n - (|T^-| - |T_i^-|) \overset{(6)}{\geq} \beta^-(1+\alpha)n - (8+\Delta')\sqrt{\eta}n - \beta^-(1-\alpha\beta)n + |T_i^-| \\ &\geq \beta^-(\alpha + \alpha\beta/2)n + |T_i^-|. \end{aligned}$$

So $|S^*| \geq (1+\alpha+2\mu)|T_i^-|$. Therefore if $|T_i^-| \geq \beta^-\alpha n/2$, then $|T_i^-| \geq n_0'$, and so we can embed T_i^- in $G[S^*]$ by our choice of n_0'. On the other hand, if $|T_i^-| < \beta^-\alpha n/2$ then $|S^*| \geq 3|T_i^-|$, and so we can embed T_i^- in $G[S^*]$ by Theorem 1.2.)

Finally, if the current vertex of T_{con} corresponds to some T_i^+, we embed T_i^+ in the unoccupied vertices of S^+ by a similar method to the method used to embed some T_i^- in the unoccupied vertices of $G[S^-]$. We continue in this manner until we have embedded the T_i, T_i^+ or T_i^- corresponding to each vertex of T_{con}, at which point we will have obtained an embedding of T in G, completing the proof. At each stage in this proof we had 'room to spare' in our choices, and so the fact that the expressions for $|T_i|$, $|T_i^+|$ and $|T_i^-|$ and other such expressions may not be integers is not a problem. □

Acknowledgements

We would like to thank the anonymous referees for their many helpful suggestions and comments.

References

[1] N. Alon, R.A. Duke, H. Lefmann, V. Rödl and R. Yuster, The algorithmic aspects of the regularity lemma, *Journal of Algorithms* **16** (1994), 80–109.
[2] N. Alon and A. Shapira, Testing subgraphs in directed graphs, *Journal of Computer and System Sciences* **69** (2004), 354–382.
[3] B. Bollobás, Random graphs, Cambridge University Press, 2001.
[4] S. Céroi and F. Havet, Trees with three leaves are $(n+1)$-unavoidable, *Discrete Applied Mathematics* **141** (2004), 19–39.
[5] A. El Sahili, Trees in tournaments, *Journal of Combinatorial Theory, Series B* **92** (2004), 183–187.
[6] R. Häggkvist and A.G. Thomason, Trees in tournaments, *Combinatorica* **11** (1991), 123–130.
[7] F. Havet, Trees in tournaments, *Discrete Mathematics* **243** (2002), 121–134.
[8] F. Havet, On unavoidability of trees with k leaves, *Graphs and Combinatorics* **19** (2003), 101–110.
[9] F. Havet and S. Thomassé, Median orders of tournaments: a tool for the second neighbourhood problem and Sumner's conjecture, *Journal of Graph Theory* **35** (2000), 244–256.
[10] F. Havet and S. Thomassé, Oriented Hamiltonian paths in tournaments: a proof of Rosenfeld's conjecture, *J. Combin. Theory B* **78** (2000), 243–273.
[11] S. Janson, T. Łuczak and A. Ruciński, Random graphs, Wiley-Interscience, 2000.
[12] P. Keevash, D. Kühn and D. Osthus, An exact minimum degree condition for Hamilton cycles in oriented graphs, *Journal of the London Mathematical Society* **79** (2009), 144–166.
[13] L. Kelly, D. Kühn and D. Osthus, A Dirac type result for oriented graphs, *Combinatorics, Probability and Computing* **17** (2008), 689–709.
[14] D. Kühn and D. Osthus, Embedding large subgraphs into dense graphs, in Surveys in Combinatorics 2009, eds. S. Huczynska, J.D. Mitchell and C.M. Roney-Dougal, *London Math. Soc. Lecture Notes* **365**, Cambridge University Press, 2009, 137–168.
[15] D. Kühn, R. Mycroft and D. Osthus, A proof of Sumner's conjecture for large tournaments, *Proceedings of the London Mathematical Society*, to appear.

[16] D. Kühn, D. Osthus and A. Treglown, Hamiltonian degree sequences in digraphs, *Journal of Combinatorial Theory, Series B* **100** (2010), 367–380.

[17] C. McDiarmid, Concentration, in Probabilistic Methods for Algorithmic Discrete Mathematics, eds. M. Habib, C. McDiarmid, J. Ramirez-Alfonsin and B. Reed, Springer, 1998, 195–248.

[18] L. Redei, Ein kombinatorischer Satz, *Acta Lit. Szeged* **7** (1934), 39–43.

[19] K.B. Reid and N.C. Wormald, Embedding oriented n-trees in tournaments, *Studia Scientiarum Mathematicarum Hungarica* **18** (1983) 377–387.

[20] B. Sudakov and J. Vondrák, A randomized embedding algorithm for trees, *Combinatorica* **30** (2010), 445–470.

[21] A. Thomason, Paths and cycles in tournaments, *Transactions of the American Mathematical Society* **296** (1986), 167–180.

[22] N.C. Wormald, Subtrees of large tournaments, *Combinatorial Mathematics X, Springer Lecture Notes in Mathematics* **1036** (1983) 417–419.

Daniela Kühn, Deryk Osthus,
School of Mathematics,
University of Birmingham,
Birmingham,
B15 2TT,
United Kingdom,
{kuehn,osthus}@maths.bham.ac.uk

Richard Mycroft,
School of Mathematical Sciences,
Queen Mary,
University of London,
London,
E1 4NS,
United Kingdom,
r.mycroft@qmul.ac.uk