COPY-PASTE TREES AND THEIR GROWTH RATES

JOSEPH PREVITE, MICHELLE PREVITE, AND MARY VANDERSCHOOT

Abstract. In this paper, we describe a copy-and-paste method for constructing a class of infinite self-similar trees. A copy-paste tree is constructed by repeatedly attaching copies of a finite tree (called a generator) to certain designated attachment vertices. We show that each generator has an associated nonnegative matrix which can be used to determine a formula for the growth function of the copy-paste tree. In our main theorem, we use results from Perron-Frobenius theory to show that every copy-paste tree has exponential growth with growth rate equal to the spectral radius of its associated matrix.

1. Introduction and Motivation

This work was motivated by the study of the growth rate of self-similar graphs that arise from a repetitive construction process called a vertex replacement rule, where certain vertices of a graph are systematically deleted and replaced by copies of certain finite graphs (see [7]). The copy-paste construction method described in this current paper is a special case of a vertex replacement rule applied to trees. Vertex replacement rules themselves were motivated by studying the horospheres of the geodesic flow of a two-dimensional singular branching space having non-positive curvature (see [1]).

Many other recursive construction schemes of graphs have been studied, see [9] and [3] and the references therein. These constructions, when applied to trees, often yield trees having finitely many cones (i.e., there are finitely many automorphism classes of cones, where a cone is obtained by taking a vertex and all its descendants). See [4]. The copy-paste method described in this paper also produces trees that have finitely many cones, but not all such trees can be obtained from the copy-paste procedure. The reader should be aware that unlike most constructions, we allow for edges of any integer length (not all edges have length one). Lastly, it should be mentioned that the results of this paper can be adjusted to compute the growth rates of trees that do not have finitely many cones, as is demonstrated in the last example of the last section.

The topic of this paper is a class of self-similar infinite rooted trees, which we call copy-paste trees, that are constructed by an iterative process of attaching copies of a given tree (called a generator) to designated vertices.
(called attachment vertices). In our main result (Theorem 2.6), we show that
every copy-paste tree has exponential growth with growth rate equal to the
spectral radius of a matrix whose entries are determined by the number
of the generator’s attachment vertices and their respective distances to the
root.

2. Definitions and Examples

A tree is a connected graph with no circuits. The trees that we consider
in this paper are equipped with more structure than a combinatorial object.
In particular, if one considers a tree as a simplicial complex, we consider
its realization. We impose a metric on each tree by assuming each edge
is isometric to an open interval, making any tree a metric space by using
the associated intrinsic metric. In other words, the distance $d(x, y)$ between
two points $x$ and $y$ (not necessarily vertices) is realized by the minimal
length of a (simple) path connecting $x$ to $y$. In particular, if $x$ and $y$ are
vertices then $d(x, y)$ is the sum of the lengths of the unique edges which
are part of a simple path from $x$ to $y$. Furthermore, we assume that every
edge has integer length and allow for different edges to have different integer
lengths.

Let $T$ be a rooted tree (i.e., a tree with one distinguished vertex $r$ called
the root). Denote the set of vertices as $V(T)$ and the set of edges as $E(T)$.
Given points $x$ and $y$ of $T$ (not necessarily vertices), if $x$ lies on a simple
path from $y$ to the root, then $x$ is an ancestor of $y$, and $y$ is a descendant
of $x$. We define the height of $T$ by

$$h(T) = \sup_{v \in V(T)} d(v, r).$$

A rooted tree is called infinite if its height is unbounded. For integers $i \geq 0$,
define the vertex count function of $T$ by

$$c_i(T) = |\{v \in V(T) \mid d(v, r) = i\}|,$$

where $| \cdot |$ denotes cardinality.

**Definition 2.1.** Let $T$ be an infinite rooted tree. The growth function of
$T$ is given by $f(T, n) = \sum_{i=0}^{n} c_i(T)$.

**Definition 2.2.** An infinite rooted tree $T$ has exponential growth if there
exists $\lambda > 1$ and positive constants $\alpha_1$ and $\alpha_2$ such that

$$\alpha_1 \lambda^n \leq f(T, n) \leq \alpha_2 \lambda^n$$

for all $n \in \mathbb{N}$. The constant $\lambda$ is called the growth rate of $T$. Observe that
$T$ has exponential growth with growth rate $\lambda$ if, and only if,

$$\lim_{n \to \infty} \frac{\ln f(T, n)}{n} = \ln \lambda.$$
The topic of this paper is a class of self-similar infinite rooted trees called 
**copy-paste trees** that are constructed according to rules described below in Definition 2.3.

Define a **generator** to be a finite rooted tree $H$, having root $r$, with a subset $V_{att}(H)$ of $V(H) \setminus \{r\}$, called the set of **attachment vertices** of $H$, such that $|V_{att}(H)| > 1$.

**Definition 2.3.** An infinite tree $T_H$ is a **copy-paste tree** if it is obtained by the following inductive procedure:

1. **Step 1.** Start with a single generator $H$. (The root $r$ of this first copy of $H$ will be the root of $T_H$.)
2. **Step 2.** To each attachment vertex of $H$ glue another copy of $H$ by identifying the root of the copy to the attachment vertex.
3. **Step 3.** To each attachment vertex of a copy of $H$ that appeared in Step 2, attach another copy of $H$ by identifying the root of this latest copy to this attachment vertex.
4. **Step 4.** Proceed inductively as follows: For each attachment vertex on a copy of $H$ that appeared at the previous stage, attach another copy of $H$ by identifying the root of this latest copy to this attachment vertex.

The limit of this procedure yields an infinite rooted tree $T_H$. We define the set of **attachment vertices** of $T_H$, denoted $V_{att}(T_H)$, to be the set of all vertices of $T_H$ that were attachment vertices at any stage of the construction of $T_H$.

**Example 2.4.** Let $H_1$ and $H_2$ be the generators shown in Figures 1 and 2, respectively. Suppose $V_{att}(H_1) = \{v_1, v_2\}$, $V_{att}(H_2) = \{w_1, w_2, w_3, w_4\}$ and all edges have length one. Figure 1 shows the first three stages in the construction of $T_{H_1}$. Note that $T_{H_1} = T_{H_2}$, the infinite full binary tree, but $V_{att}(T_{H_1}) \neq V_{att}(T_{H_2})$.

![Figure 1](image1.png)

**Figure 1.** First stages in the construction of $T_{H_1}$.

**Example 2.5.** Let $H$ be the generator shown in Figure 3. Suppose $V_{att}(H) = \{v_1, v_2\}$. Suppose also that edge $(r, w)$ has length two, and all other edges of $H$ have length one. Figure 3 also shows $T_H$.
Figure 2. $H_2$ is another generator of the infinite full binary tree.

Figure 3. A generator $H$ for the copy-paste tree $T_H$.

Our main result about copy-paste trees is the following theorem:

**Theorem 2.6.** Let $T_H$ be a copy-paste tree with generator $H$ and let

$$A_H = \begin{bmatrix} 0_{N-1,1} & I_{N-1,N-1} & \cdots & b_2 & b_1 \\ b_N & b_{N-1} & \cdots & b_2 & b_1 \end{bmatrix},$$

where $b_i(H) = |\{v \in V_{att}(H) : d(v, r) = i\}|$ and $I_{N-1,N-1}$ is the size $N-1$ identity matrix. Then $T_H$ has exponential growth with growth rate $\rho(A_H)$, where $\rho(A_H)$ is the spectral radius of $A_H$ (i.e., $\rho(A_H)$ is the maximum magnitude of the eigenvalues of $A_H$).

Note that Theorem 2.6 requires that the generator $H$ has at least two attachment vertices, i.e., $|V_{att}(H)| > 1$. Applying the copy-and-paste method to a rooted tree with only one attachment vertex results in linear growth.

A key step in the proof of Theorem 2.6 is determining a formula for a copy-paste tree’s growth function using powers of the nonnegative matrix $A_H$. We conclude this section with some notation, definitions, and results on nonnegative matrices.

We denote the $(i, j)$ entry of $A$ by $a_{i,j}$, and the $(i, j)$ entry of $A^n$ by $(A^n)_{i,j}$. A real matrix $A$ is **nonnegative** (positive) if $a_{i,j} \geq 0$ ($a_{i,j} > 0$) for every $(i, j)$. We write $A \geq 0$ if $A$ is nonnegative, and $A > 0$ if $A$ is
positive. A nonnegative square matrix \( A \) is called **irreducible** if for any \((i, j)\) there exists \( k \) such that \((A^k)_{i,j} > 0\). A nonnegative square matrix \( A \) is called **primitive** if \( A^k > 0 \) for some \( k \).

The following theorems can be found in [2], [5], and [8].

**Theorem 2.7. (Perron-Frobenius)** Let \( A \) be an irreducible matrix.

(a) \( A \) has a positive real eigenvalue \( r = \rho(A) \) such that \( r \geq |\lambda_i| \) for any eigenvalue \( \lambda_i \) of \( A \). (The eigenvalue \( r \) is called the **Perron eigenvalue** of \( A \).)

(b) Furthermore, \( r \) has algebraic and geometric multiplicity one and has a positive eigenvector \( x \).

(c) Any nonnegative eigenvector of \( A \) is a multiple of \( x \).

(d) \( \min_i \sum_j a_{i,j} \leq r \leq \max_i \sum_j a_{i,j} \).

(e) If \( A \) is primitive, then \( r > |\lambda_i| \) for any eigenvalue \( \lambda_i \) of \( A \), \( \lambda_i \neq r \).

**Theorem 2.8.** Let \( A \) be a primitive matrix with Perron eigenvalue \( r \). Then there exists a matrix \( L \) with positive entries so that
\[
\lim_{k \to \infty} \left( \frac{A}{r} \right)^k = L.
\]

The associated directed graph of an \( n \times n \) nonnegative matrix \( A \), denoted \( G(A) \), consists of \( n \) vertices \( v_1, \ldots, v_n \), where there exists a directed edge from \( v_i \) to \( v_j \), denoted \((v_i, v_j)\), if, and only if, \( a_{i,j} \neq 0 \). A graph is **strongly connected** if for any two vertices \( v_i \) and \( v_j \) there is a directed path from \( v_i \) to \( v_j \). The following theorem gives necessary and sufficient conditions for a matrix to be irreducible or primitive.

**Theorem 2.9.** Let \( A \) be a nonnegative matrix.

1. Then \( A \) is irreducible if, and only if, \( G(A) \) is strongly connected.
2. Suppose also that \( A \) is irreducible. Then \( A \) is primitive if, and only if, \( G(A) \) has two cycles of relatively prime lengths.

3. Computing the Growth Function of a Copy-Paste Tree

Throughout this section, let \( H \) be a generator of height \( N \). Recall from Theorem 2.6 that for \( i = 1, \ldots, N \), \( b_i(H) = |\{v \in V_{att}(H) : d(v, r) = i\}| \). The following definitions will be used to construct a partition of the vertices of \( T_H \) that leads to a recursive formula for \( c_i(T_H) \). (If only one generator \( H \) and its corresponding copy-paste tree \( T_H \) are under consideration, we will often write \( T \) for \( T_H \), \( b_i \) for \( b_i(H) \), and \( c_i \) for \( c_i(T_H) \).)

**Definition 3.1.** Define the closed ball of radius \( n \) about any point (not necessarily a vertex) \( x \in T \), denoted \( B_n(T, x) \), by
\[
B_n(T, x) = \{z \in T : d(z, x) \leq n\}.
\]
Definition 3.2. For \( i \geq 1 \), define the \( i \)-shell, denoted \( S_i \), by
\[
S_i = B_i(T, r) \setminus B_{i-1}(T, r).
\]

For example, Figure 8 shows the shells \( S_1, \ldots, S_6 \) for the copy-paste tree of Example 2.5 (depicted in Figure 3). Note that in this example not all points of distance \( i \) to the root are vertices in \( S_i \). (Vertices are denoted in Figure 8 with filled in dots.) Also note that \( |V(S_n)| = c_n(T) \).

Figure 4. First shells of the copy-paste tree of Example 2.5.

An immediate consequence of Definitions 2.3 and 3.2 is the following:

Observation 3.3. Let \( T \) be a copy-paste tree and let \( b \in V_{\text{att}}(T) \). Consider \( D(T, b) \), which consists of \( b \) together with all of its descendants. Then for all \( i \geq 1 \), \( (B_i(T, b) \setminus B_{i-1}(T, b)) \cap D(T, b) \) is isometric to \( S_i \).

The next lemma is technical but important.

Lemma 3.4. Let \( T \) be a copy-paste tree with a generator \( H \) of height \( N \). For all \( n > N \), there exists a partition \( P_n \) of \( S_n \) such that each element of \( P_n \) is isometric to one of the shells \( S_1, \ldots, S_N \).

For \( 1 \leq i \leq N \) and \( 1 \leq n \leq N \), let \( t_i(S_n) = \delta_{i,n} \), where \( \delta \) is the Kronecker delta. For \( 1 \leq i \leq N \) and \( n > N \), let \( t_i(S_n) \) be the number of isometric copies of \( S_i \) in the partition \( P_n \) of \( S_n \). Then the following formulas hold:

(a) \( c_n(T) = t_1(S_n)c_1(T) + t_2(S_n)c_2(T) + \cdots + t_N(S_n)c_N(T) \).

(b) \( t_i(S_{N+1}) = b_{N+1-i} \) for \( i = 1, \ldots, N \).

(c) For \( n > N \),
\[
t_j(S_{n+1}) = \begin{cases} t_N(S_n)t_1(S_{n+1}) & j = 1 \\ t_{j-1}(S_n) + t_N(S_n)t_j(S_{n+1}) & 2 \leq j \leq N \end{cases}
\]

Proof. Let \( n > N \). For each \( x \in S_n \) (not necessarily a vertex), let \( a(x) \in T \) be the unique ancestor of \( x \) that is exactly \( N \) closer to the root than \( x \). Since \( h(H) = N \), any path from \( x \) to \( a(x) \) must contain at least one attachment
vertex. If \( a(x) \) is itself an attachment vertex, let \( k(x) = a(x) \). If \( a(x) \) is not an attachment vertex, then \( a(x) \) lies in a unique copy of \( H \) that was used in the construction of \( T \), which we will denote \( H(a(x)) \). In this case, let \( k(x) \) be the unique attachment vertex of \( H(a(x)) \) that is an ancestor of \( x \). Consult Example 3.5 for an illustration.

Define an equivalence relation on \( S_n \) by \( xRy \) if \( k(x) = k(y) \). (In Figure 5, \( yRx \).) Let \( \mathcal{P}_n \) denote the partition of \( S_n \) formed by the equivalence classes of \( R \), and let \( \mathcal{P}_n(x) \) denote the partition element of \( \mathcal{P}_n \) containing \( x \). Observe that \( \mathcal{P}_n(x) = (\mathcal{B}_i(T, k(x)) \setminus \mathcal{B}_{i-1}(T, k(x))) \cap D(T, k(x)) \) for some \( 1 \leq i \leq N \) which, by Observation 3.3, is isometric to \( S_i \) for some \( 1 \leq i \leq N \). The formula in part (a) immediately follows.

We next use the partition to find formulas for counting the number of isometric copies of \( S_1, \ldots, S_N \) in the \( S_{N+1} \) shell. Let \( x \in S_{N+1} \). Clearly, \( d(a(x), r) \leq 1 \) so \( a(x) \) is in the initial copy of \( H \) (the one with \( r \)). By part (a), \( \mathcal{P}_n(x) \) is isometric to \( S_i \) for some \( 1 \leq i \leq N \). Therefore \( k(x) \) is an attachment vertex of the original copy of \( H \) that is \( N + 1 - i \) away from \( r \). Finally, there is a one-to-one correspondence between the number of partition elements in \( S_{N+1} \) and the attachment vertices of \( H \) by the bijection \( \mathcal{P}_n(x) \leftrightarrow k(x) \). Thus the formula in part (b) holds.

Now we establish the formula in part (c). Let \( n > N \). Both \( S_n \) and \( S_{n+1} \) are partitioned into elements isometric to \( S_1, \ldots, S_N \) by part (a). For every partition element of \( S_n \) there are two cases.

Case 1: If a partition element \( P \) of \( S_n \) is isometric to \( S_i \) for \( 1 \leq i \leq N - 1 \), then such an element will uniquely correspond to a partition element \( Q \) of \( S_{n+1} \) that is isometric to \( S_{i+1} \) for \( 1 \leq i \leq N - 1 \). This unique partition element of \( S_{n+1} \) corresponds to descendants of \( P \).

Case 2: If a partition element \( P \) of \( S_n \) is isometric to \( S_N \), then its descendants in \( S_{n+1} \) will consist of an isometric copy of \( S_{N+1} \) which in turn can be partitioned into \( t_i(S_{N+1}) \) elements for each \( i \) with \( 1 \leq i \leq N \).

Note that those partition elements of \( S_{n+1} \) that are isometric to \( S_1 \) can only arise in the second case. So \( t_1(S_{n+1}) = t_N(S_n)t_1(S_{N+1}) \) since there are \( t_N(S_n) \) distinct isometric copies of \( S_N \) in \( S_n \). Those partition elements of \( S_{n+1} \) that are isometric to \( S_j \) for \( 2 \leq j \leq N \) can arise in either case. Therefore, the total number of partition elements in \( S_{n+1} \) isometric to \( S_j \) for \( 2 \leq j \leq N \) is

\[
t_j(S_{n+1}) = t_{j-1}(S_n) + t_N(S_n)t_j(S_{N+1}).
\]

\[ \blacksquare \]

**Example 3.5.** Figure 5 shows the copy-paste tree of Example 2.5 (depicted in Figure 3). Recall that \( V_{ad}(H) = \{v_1, v_2\} \) and \( N = 4 \). For \( w, x \) and \( y \) as indicated in Figure 5, \( a(w), a(x), a(y) \) are in the first copy of \( H \) used in the construction of \( T_H \). Since \( a(w) = k(w) = v_1 \), \( w \) is in an isometric copy of \( S_4 \). Note that \( k(w) \neq k(y) \). Hence \( w \) and \( y \) are not related and are not in the same partition element of \( S_5 \). Since \( k(x) = k(y) = v_2 \), \( x \) and \( y \) are
related and are in the same isometric copy of $S_1$. The partition elements of $S_5$, $S_6$ and $S_7$ are circled in Figure 5.

**Figure 5.** Partition elements of $S_5$, $S_6$ and $S_7$.

**Definition 3.6.** Define the **copy-paste matrix** $A_H$ associated to a generator $H$ of height $N$ to be the $N \times N$ nonnegative matrix

$$A_H = \begin{bmatrix}
t_1(S_2) & t_2(S_2) & t_3(S_2) & \cdots & t_N(S_2) \\
t_1(S_3) & t_2(S_3) & t_3(S_3) & \cdots & t_N(S_3) \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
t_1(S_{N+1}) & t_2(S_{N+1}) & t_3(S_{N+1}) & \cdots & t_N(S_{N+1})
\end{bmatrix}. \quad (1)$$

Recall that for $1 \leq i \leq N$ and $1 \leq j \leq N$, $t_i(S_j) = \delta_{i,j}$, where $\delta$ is the Kronecker delta. Using this fact together with Lemma 3.4 part (b), the copy-paste matrix in (1) can be simplified to the block matrix

$$A_H = \begin{bmatrix}
0_{N-1,1} & I_{N-1,N-1} & \cdots & b_2 & b_1 \\
b_N & b_{N-1} & \cdots & b_2 & b_1
\end{bmatrix}, \quad (2)$$

where $I_{N-1,N-1}$ is the size $N-1$ identity matrix.

The next lemma lists several properties of copy-paste matrices.

**Lemma 3.7.** Let $A$ be a copy-paste matrix of the form (2) with spectral radius $\rho(A)$.

(a) If $\lambda$ is an eigenvalue of $A$, then $[1 \ \lambda \ \cdots \ \lambda^{N-1}]^t$ is an eigenvector of $A$ corresponding to $\lambda$ (where $t$ denotes transpose).
(b) For all $N \geq 2$, the determinant of $A - I$ is given by

$$|A - I| = (-1)^N \left( 1 - \sum_{i=1}^{N} b_i \right),$$

(c) 1 is not an eigenvalue of $A$.

(d) If $b_N \neq 0$, then $A$ is irreducible.

(e) If $b_N = b_{N-1} = \cdots = b_{N-j} = 0$, $b_{N-(j+1)} \neq 0$ and

$$A' = \begin{bmatrix}
  0 & I_{N-(j+2),N-(j+2)} & \cdots & b_2 & b_1 \\
  b_{N-(j+1)} & b_{N-(j+2)} & \cdots & b_2 & b_1
\end{bmatrix}, \quad (3)$$

then

$$|A - \lambda I| = (-\lambda)^{j+1}|A' - \lambda I| \text{ and } \rho(A) = \rho(A'),$$

where $I$ is the appropriate sized identity matrix.

Proof. Part (a): Let $v = \begin{bmatrix} v_1 & v_2 & \cdots & v_N \end{bmatrix}^t$ be an eigenvector of $A$ associated to $\lambda$. Since $Av = \lambda v$ we have that

$$v_i = \lambda v_{i-1} \text{ for } i = 2, \ldots, N \quad (4)$$

$$b_N v_1 + b_{N-1} v_2 + \cdots + b_1 v_N = \lambda v_N \quad (5)$$

Since $v$ is an eigenvector of $A$, then $v_1 \neq 0$. For if $v_1 = 0$, then $v_2 = \cdots = v_N = 0$ by Equation (4). Without loss of generality, $v_1 = 1$. From Equation (4) it follows that

$$v_i = \lambda^{i-1}, \text{ for } i = 2, \ldots, N. \quad (6)$$

Substituting Equation (6) into Equation (5) yields

$$b_N + b_{N-1} \lambda + b_{N-2} \lambda^2 + \cdots + b_1 \lambda^{N-1} = \lambda^N. \quad (7)$$

Using Equation (7), it is now straightforward to verify

$$A \begin{bmatrix} 1 & \lambda & \lambda^2 & \cdots & \lambda^{N-1} \end{bmatrix}^t = \lambda \begin{bmatrix} 1 & \lambda & \lambda^2 & \cdots & \lambda^{N-1} \end{bmatrix}^t.$$

Part (b): We will argue by induction on $N$. When $N = 2$, we have that

$$|A - I| = \begin{vmatrix} -1 & 1 \\
  b_2 & b_1 - 1 \end{vmatrix} = 1 - b_1 - b_2.$$

So the base case holds. Next assume the determinant formula is valid for any $(k-1) \times (k-1)$ matrix of the form given by Equation (2), where $k \geq 3$.

Let $A = \begin{bmatrix} 0_{N-1,1} & I_{N-1,N-1} & \cdots & b_2 \\
  b_k & b_{k-1} & \cdots & b_1
\end{bmatrix}$. Using a cofactor expansion along the first column of $A - I$ we have that
\[
|A - I| = -1
\]

\[
\begin{pmatrix}
-1 & 1 & 0 & \cdots & 0 \\
0 & -1 & 1 & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & \cdots & 1 \\
b_{k-1} & b_{k-2} & b_{k-3} & \cdots & b_1 - 1
\end{pmatrix}
\begin{pmatrix}
1 & 0 & 0 & \cdots & 0 \\
-1 & 1 & 0 & \cdots & 0 \\
0 & -1 & 1 & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & -1 & 1
\end{pmatrix},
\]

By applying the induction hypothesis and properties of triangular matrices, we see

\[
|A - I| = (-1)^{k-1}
\left(1 - \sum_{i=1}^{k-1} b_i\right) + (-1)^{k+1} b_k = (-1)^k \left(1 - \sum_{i=1}^{k} b_i\right),
\]

which is the desired result.

Part (c): Since \(\sum_{i=1}^{N} b_i \geq 2\), it follows by part (b) that \(|A - I|\) is not zero and hence 1 is not an eigenvalue of \(A\).

Part (d): By Theorem 2.9, it suffices to show that \(G(A)\), the associated directed graph of \(A\), is strongly connected. Let \(x_1, \ldots, x_N\) denote the vertices of \(G(A)\). Observe that for \(i = 1, \ldots, N - 1\), there is a directed edge in \(G(A)\) from \(x_i\) to \(x_{i+1}\). Since \(a_{N,1} = b_N \neq 0\), there is also a directed edge in \(G(A)\) from \(x_N\) to \(x_1\). Therefore \(G(A)\) has a cycle of length \(N\), namely \(x_N, x_1, x_2, \ldots, x_{N-1}, x_N\), that includes every vertex of \(G(A)\). Hence \(G(A)\) is strongly connected.

Part (e): Since \(A\) is irreducible, it follows from the Perron-Frobenius Theorem (Theorem 2.7) that \(\rho(A) \geq 1\). The result now follows by part (c).

Part (f): It is straightforward to check that \(|A - \lambda I| = (-\lambda)^{j+1}|A' - \lambda I|\) by computing the determinant of \((A - \lambda I)\) by a sequence of first column cofactor expansions.

\[\square\]

**Theorem 3.8.** Let \(T\) be a copy-paste tree with generator \(H\) of height \(N\), and let \(A\) be the copy-paste matrix associated to \(H\).

(a) Then \((A^n)_{i,j} = t_j(S_{n+1})\) for all \(n \geq 1\).

(b) Let \(u = [1 \ 1 \ \cdots \ 1]\) and \(c(n) = [c_{n+1} \ c_{n+2} \ \cdots \ c_{n+N}]^t\). Then the growth function of \(T\) satisfies

\[
f(T, kN) = 1 + u \left[I + A_N + \cdots + A^{(k-1)N}\right] c(0).
\]

(c) If \(A\) is primitive, then \(T\) has exponential growth with growth rate \(\rho(A)\).

**Proof.** Part (a): We argue by induction on \(n\). When \(n = 1\), the result clearly holds. Next we assume the result holds for \(n = k\), where \(k \geq 1\). Note that multiplication in the order below results in a matrix whose first \(N - 1\) rows are just the rows of \(A^k\) shifted up by one row.
\[ A^{k+1} = AA^k = A \begin{bmatrix} t_1(S_{k+1}) & t_2(S_{k+1}) & \cdots & t_N(S_{k+1}) \\ t_1(S_{k+2}) & t_2(S_{k+2}) & \cdots & t_N(S_{k+2}) \\ t_1(S_{k+3}) & t_2(S_{k+3}) & \cdots & t_N(S_{k+3}) \\ \vdots & \vdots & \ddots & \vdots \\ t_1(S_{k+N}) & t_2(S_{k+N}) & \cdots & t_N(S_{k+N}) \end{bmatrix}. \]

Thus the entries in the first \(N - 1\) rows of \(A^{k+1}\) are the desired result. To show the last row of \(A^{k+1}\) is the desired result we write \(A^{k+1} = A^k A\) and use part (c) of Lemma 3.4.

Part (b): The following calculation uses part (a) of Lemma 3.4 to verify that \(c(n) = A^n c(0)\):

\[
A^n \begin{bmatrix} c_1 \\ \vdots \\ c_N \end{bmatrix} = \begin{bmatrix} t_1(S_{n+1}) & \cdots & t_N(S_{n+1}) \\ \vdots & \ddots & \vdots \\ t_1(S_{n+N}) & \cdots & t_N(S_{n+N}) \end{bmatrix} \begin{bmatrix} c_1 \\ \vdots \\ c_N \end{bmatrix} = \begin{bmatrix} c_1 t_1(S_{n+1}) + \cdots + c_N t_N(S_{n+1}) \\ \vdots \\ c_1 t_1(S_{n+N}) + \cdots + c_N t_N(S_{n+N}) \end{bmatrix} = \begin{bmatrix} c_{n+1} \\ \vdots \\ c_{N+n} \end{bmatrix}.
\]

Observe that \(u A^n c(0) = uc(n) = \sum_{i=1}^{N} c_{i+n}\). Therefore,

\[
f(T, kN) = \sum_{i=0}^{kN} c_i = 1 + u \left[ I + A^N + A^{2N} + \cdots + A^{(k-1)N} \right] c(0). \tag{8}\]

Part (c): Suppose \(A\) is primitive. Let \(r = \rho(A)\) and let \(L\) be the positive matrix (as described in Theorem 2.8) given by \(L = \lim_{k \to \infty} \left( A^k \right) / r^k\). It follows from Lemma 3.7 part (c) and Theorem 2.7 part (d) that \(r > 1\).

For every \(n \geq 0\) there exists an integer \(k_n \geq 0\) such that \(k_n N \leq n \leq (k_n + 1)N\). \tag{9}\]

Clearly,

\[
0 < f(T, k_n N) \leq f(T, n) \leq f(T, (k_n + 1)N). \tag{10}\]

So, for \(r > 1\) we have

\[
\frac{f(T, k_n N)}{r^{k_n N}} \leq \frac{f(T, n)}{r^n} \leq \frac{f(T, (k_n + 1)N)}{r^{k_n N}}. \tag{11}\]
Since 1 is not an eigenvalue of $A$ (Lemma 3.7), the matrix $A^N - I$ is invertible. Therefore Equation (8) of part (b) can be simplified as follows:

$$f(T, kN) = 1 + u \left[ I + A^N + \cdots + A^{(k-1)N} \right] c(0)$$

$$= 1 + u \left[ (A^N - I)^{-1}(A^{kN} - I) \right] c(0)$$

$$= 1 + u \left[ (A^N - I)^{-1} A^{kN} \right] c(0) - u \left[ (A^N - I)^{-1} \right] c(0).$$

From Inequality (11), Definition 2.2, and from convergent sequence results, it follows that $T$ has exponential growth with growth rate $r$ if there exists real numbers $\ell_1$ and $\ell_2$, $0 < \ell_1 \leq \ell_2$, such that

$$\lim_{k \to \infty} \frac{f(T, kN)}{r^{kN}} = \ell_1,$$

and

$$\lim_{k \to \infty} \frac{f(T, (k+1)N)}{r^{kN}} = \ell_2.$$

We will show that $\ell_1$ exists. By Equation (12),

$$\frac{f(T, kN)}{r^{kN}} = \frac{1}{r^{kN}} + \frac{u \left[ (A^N - I)^{-1} A^{kN} \right] c(0)}{r^{kN}} - \frac{u \left[ (A^N - I)^{-1} \right] c(0)}{r^{kN}}.$$

Since $r > 1$, the first and last terms of the right-hand side of Equation (13) vanish as $k \to \infty$. Since $\lim_{k \to \infty} A^{kN}/r^{kN} = L$, we have

$$\ell_1 = \lim_{k \to \infty} \frac{f(T, kN)}{r^{kN}} = \frac{u \left[ (A^N - I)^{-1} \right] Lc(0)}{r^{kN}}.$$

We next show that $\ell_1 > 0$. From Equation (8) we have that

$$\frac{f(T, kN)}{r^{kN}} = \frac{1 + u \left[ I + A^N + \cdots + A^{(k-1)N} \right] c(0)}{r^{kN}}.$$

Since $A$ is primitive, $r > 1$, and the entries of $u$ and $c(0)$ are positive, it immediately follows from Equation (14) that

$$\frac{f(T, kN)}{r^{kN}} \geq \frac{u A^{(k-1)N} c(0)}{r^{kN}}.$$

Since $\lim_{k \to \infty} \frac{u A^{(k-1)N} c(0)}{r^{(k-1)N} r^{2N}} = \frac{u Lc(0)}{r^{2N}} > 0$, it follows that $\ell_1 > 0$.

A similar argument establishes that $\ell_2$ also exists and, by Equation (11), $\ell_2 \geq \ell_1$.

4. Copy-Paste Trees with Exponential Growth

The purpose of this section is to show, using a bootstrapping argument, that every copy-paste tree has exponential growth. In particular, we consider the case where the associated copy-paste matrix is reducible. We then construct related copy-paste trees having the same growth rate whose associated copy-paste matrices are primitive.
4.1. Reducing to the Case of Irreducible Copy-Paste Trees. Let $H$ be a generator of height $N$ for a copy-paste tree $T_H$ and let $M$ be the maximum of the distances of all the attachment vertices in $H$ to the root. If $N > M$ then $b_N(H) = 0$ and hence the associated copy-paste matrix $A_H$ has a column of all zeros, making it reducible.

We build two related copy-paste trees with generators $H'$ and $\tilde{H}$ of height $M$ with associated $M \times M$ matrices

$$
\begin{bmatrix}
0_{M-1,1} & I_{M-1,M-1} & \cdots & b_2 & b_1 \\
b_M & b_{M-1} & \cdots & 0 & 0
\end{bmatrix}.
$$

(16)

This matrix satisfies the hypotheses of parts (d) and (e) of Lemma 3.7. We will construct $H'$ and $\tilde{H}$ in such a way so that

$$f(T_{H'}, n) \leq f(T_H, n) \leq f(T_{\tilde{H}}, n).$$

We will use this to show that $T_H$ has the same growth rate as $T_{H'}$ and $T_{\tilde{H}}$.

To build $H'$, start with a root $r'$ and for each attachment vertex $v \in V_{att}(H)$, create an attachment vertex $v' \in V_{att}(H')$ with one edge adjacent to both $r'$ and $v'$ having length $d(r, v)$ (i.e., obtain $H'$ from $H$ by keeping only the root and all attachment vertices and removing all interior vertices). Note that $H'$ will have exactly $|V_{att}(H)| + 1$ vertices, with $|V_{att}(H)|$ leaves (i.e., vertices adjacent to exactly one edge) and $|V_{att}(H)|$ edges all adjacent to $r'$.

To construct $\tilde{H}$, again start with a root $\tilde{r}$ and for each attachment vertex $v \in V_{att}(H)$, create an attachment vertex $\tilde{v} \in V_{att}(\tilde{H})$ which is a leaf. However, unlike $H'$, create $d(\tilde{r}, v)$ edges, each of length one, and $d(\tilde{r}, v) - 1$ vertices so that these new vertices and edges form a path from $\tilde{r}$ to $\tilde{v}$ having length $d(\tilde{r}, v)$. Additionally, for every vertex $w \in V(H)$, create a non-attachment leaf $\tilde{w}$ that is exactly $M$ away from the root together with $M$ edges of length one and $M - 1$ vertices so that these new vertices and edges form a path from $\tilde{r}$ to $\tilde{w}$. (Recall that $M$ is the maximum of the distances of all the attachment vertices in $H$ to the root.) Note that all vertices except for the leaves and the root are adjacent to exactly two edges. The following example illustrates the construction of $H'$ and $\tilde{H}$. Note that $A_{H'} = A_{\tilde{H}}$ are given by (16).

Example 4.1. Let $H$ be the generator shown below in Figure 6. Suppose edges $(x, y)$ and $(z, w)$ have length two, and all other edges have length one. Suppose also that $V_{att}(H) = \{v_1, v_2, v_3\}$. Note that $H$ has height $N = d(s, r) = 6$. Since the distances of the attachment vertices to the root are $d(v_1, r) = d(v_2, r) = 4$ and $d(v_3, r) = 2$, we have that $M = 4$.

The tree $H'$ has exactly four vertices: a root $r'$ and three attachment vertices: $v'_1$, $v'_2$ and $v'_3$. Edges $(r', v'_1)$ and $(r', v'_2)$ have length four, and edge $(r', v'_3)$ has length two. The tree $\tilde{H}$ has three attachment vertices: $\tilde{v}_1$, $\tilde{v}_2$ and $\tilde{v}_3$, and fifteen non-attachment leaves. All edges of $\tilde{H}$ have length one. (See Figure 6.)
Since \( b_M \neq 0 \) we have that \( A_{H'} \) is irreducible. Furthermore, it is clear that \( c_i(H') \leq c_i(H) \leq c_i(\tilde{H}) \), from which it follows that

\[
f(T_{H'}, n) \leq f(T_H, n) \leq f(T_{\tilde{H}}, n). \tag{17}
\]

If \( A_{H'} \) is primitive, then, by Theorem 3.8, the copy-paste trees \( T_{H'} \) and \( T_{\tilde{H}} \) have exponential growth with growth rate \( \rho(A_{H'}) \). Hence by (17), it follows that \( T_H \) has the same growth rate.

**4.2. Reducing to the Case of a Primitive Copy-Paste Tree.** Although the matrix \( A_{H'} \) is irreducible, it may not be primitive. Consider for instance the copy-paste matrix \( A_{H'} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 2 & 0 & 1 & 0 \end{bmatrix} \) of Example 4.1.

One can readily check that the directed graph of \( A_{H'} \) contains cycles of lengths two and four, and hence, by Theorem 2.9, \( A_{H'} \) is not primitive.

In this section, we find sufficient conditions for \( A_{H'} \) to be primitive, and then consider the case where \( A_{H'} \) is not primitive. In the case where \( A_{H'} \) is not primitive, we construct related copy-paste trees with matrices that are primitive which will be used to compute the growth rate of \( T_{H'} \) (which in turn has the same growth rate as \( T_H \)).

**Definition 4.2.** Let \( H \) be a generator and let \( m = \gcd(d(v_1, r), \ldots, d(v_n, r)) \), where each \( v_i \in V_{\text{att}}(H) \). We say that \( H \) is unscaleable if \( m = 1 \), and that \( H \) is scaleable if \( m > 1 \).

Recall that \( H_1 \) and \( H_2 \) of Example 2.4 are both generators for the infinite full binary tree. The generator \( H_1 \) is unscaleable, whereas \( H_2 \) is scaleable with \( m = 2 \).

**Lemma 4.3.** Let \( T_H \) be a copy-paste tree with unscaleable generator \( H \) of height \( N > 1 \). Then \( A_{H'} \) is primitive.

**Proof.** Let \( G(A_{H'}) \) be the associated directed graph of \( A_{H'} \), and let \( x_1, \ldots, x_M \) denote the vertices of \( G(A_{H'}) \). Recall that in the proof of Lemma 3.7 part
(d), it was shown that $b_M(H) \neq 0$ implies that $\mathcal{G}(A_H)$ has a cycle of length $M$ that contains every vertex, namely,

$$x_M, x_1, x_2, \ldots, x_{M-1}, x_M.$$ 

Observe that if $b_i(H) \neq 0$ for some $1 \leq i \leq M$, then $(x_M, x_{M-i+1})$ is a directed edge in $\mathcal{G}(A_H)$. Using the cycle described in the previous paragraph, it follows that

$$x_M, x_{M-i+1}, x_{M-i+2}, \ldots, x_{(M-i)+(i-1)}, x_M$$

is a cycle of length $i$ in $\mathcal{G}(A_H)$.

Since $H$ is unscaleable, there exist attachment vertices $v$ and $w$ of $H$ such that $\text{gcd}(d(v, r), d(w, r)) = 1$. Let $j = d(v, r)$ and $k = d(w, r)$. Then $b_j(H) \neq 0$, $b_k(H) \neq 0$ and $\text{gcd}(j, k) = 1$. Thus $\mathcal{G}(A_H)$ has a cycle of length $j$ and a cycle of length $k$. Since $k$ and $j$ are relatively prime, Theorem 2.9 implies $A_H$ is primitive. \(\square\)

If $H$ is scaleable, we will construct two additional trees $H'_{sc}$ and $\bar{H}_{sc}$. Construct $H'_{sc}$ from $H'$ by scaling each edge of $H'$ by a factor of $\frac{i}{m}$, where $m = \text{gcd}(d(v_1, r), \ldots, d(v_n, r))$ for each $v_i \in V_{\text{att}}(H)$. (Note that the distance between any two vertices in $H'_{sc}$ is still an integer.)

Construct $\bar{H}_{sc}$ as follows: For each attachment (respectively, non-attachment) leaf of $\bar{H}$, construct an attachment (respectively, non-attachment) leaf of $\bar{H}_{sc}$ by a factor of $\frac{M}{m}$ edges, each of length one, so that there are $\frac{M}{m} - 1$ interior vertices associated with this path. The following example illustrates these constructions.

**Example 4.4.** Let $H$ be the generator of Example 4.1 (depicted in Figure 6). As before, edges $(x, y)$ and $(z, w)$ have length two, and all other edges have length one. Since the distances of the attachment vertices of $H$ to the root are $d(v_1, r) = d(v_2, r) = 4$ and $d(v_3, r) = 2$, $H$ is scaleable with $m = 2$.

The tree $H'_{sc}$ has exactly four vertices: a root $r'$ and three attachment vertices: $v_1', v_2'$ and $v_3'$. Note that $d(r', v_1') = d(r', v_2') = 2$ and $d(r', v_3') = 1$. The tree $H_{sc}$ has three attachment vertices: $v_1$, $v_2$ and $v_3$, and fifteen non-attachment leaves $\tilde{w}_1$, $\tilde{w}_2$, $\tilde{w}_3$. All edges of $H_{sc}$ have length one, $d(\tilde{r}, \tilde{v}_1) = d(\tilde{r}, \tilde{v}_2) = 2$, $d(\tilde{r}, \tilde{v}_3) = 1$, and $d(\tilde{r}, \tilde{w}_i) = 2$ for all $i = 1, \ldots, 15$. Figure 7 shows the corresponding scaled trees $H'_{sc}$ and $\bar{H}_{sc}$.

**Observation 4.5.** For a scaleable $H$, the properties below follow immediately from the construction of $H'_{sc}$. Recall that $c_i$ is the number of vertices in a tree that are distance $i$ to the root.

(a) $H'_{sc}$ and $\bar{H}_{sc}$ have height $\frac{M}{m}$, and $A_{H'_{sc}} = A_{\bar{H}_{sc}}$.

(b) $b_m(H') = b_1(H'_{sc})$ and $c_m(H') = c_1(H'_{sc})$. Hence, $c_i(T_{H'_{sc}}) = c_{im}(T_{H'})$ for $i \geq 0$.

(c) $b_m(\bar{H}) = b_1(\bar{H}_{sc})$ and $c_m(\bar{H}) = c_1(\bar{H}_{sc})$. Hence, $c_i(T_{\bar{H}_{sc}}) = c_{im}(T_{\bar{H}})$ for $i \geq 0$. 


The next lemma describes the effects of scaling on the spectral radius of the copy-paste matrix and on the growth rate of the copy-paste tree.

**Lemma 4.6.** Let $H$ be a generator with associated generator $H'$ of height $M > 1$. Suppose $H'$ is scaleable with $m = \gcd(d(v_1, r), \ldots, d(v_n, r)) > 1$ and let $H'_{sc}$ be the scaled generator described above.

(a) $H'_{sc}$ and $H_{sc}$ are unscaleable which implies that $A_{H'_{sc}} = A_{H_{sc}}$ is primitive. Moreover, $T_{H'_{sc}}$ and $T_{H_{sc}}$ have growth rate $\rho(A_{H'_{sc}})$.

(b) If $s$ is an eigenvalue of $A_{H'}$, then $s^m$ is an eigenvalue of $A_{H'_{sc}}$. Conversely, if $s$ is an eigenvalue of $A_{H'_{sc}}$ then $\sqrt[m]{s}$ is an eigenvalue of $A_{H'}$. Moreover, $\rho(A_{H'}) = \sqrt[m]{\rho(A_{H'_{sc}})}$.

(c) The copy-paste trees $T_{H'}$ and $T_{H_{sc}}$ have exponential growth rate $\rho(A_{H'})$.

(d) $T_H$ has exponential growth with growth rate $\rho(A_{H'}) = \rho(A_{H'_{sc}})$.

**Proof.** Part (a): One can show that if $H'_{sc}$ (respectively $H_{sc}$) is scaleable, then $m$ is not the greatest common divisor of $\{d(v_1, r), \ldots, d(v_n, r)\}$. Therefore, by Lemma 4.3 we have that $A_{H'_{sc}}$ is primitive and, by Theorem 3.8, $T_{H'_{sc}}$ and $T_{H_{sc}}$ have exponential growth rate $\rho(A_{H'_{sc}})$.

Part (b): Let $s$ be an eigenvalue of $A_{H'}$. By Lemma 3.7 part (a), $v = [1 \ s \ s^2 \ \cdots \ s^{M-1}]^T$ is an eigenvector of $A_{H'}$ associated with $s$. Note that the last component in the vector equation $A_{H'}v = sv$ is

\[b_M + sb_{M-1} + s^2b_{M-2} + \cdots + s^{M-1}b_1 = s^M,\]  

(18)

where $b_i$ denotes $b_i(H)$.

Since $m|M$, there exists $k \in \mathbb{Z}^+$ such that $M = km$. Since $m > 1$, the only possible nonzero entries in the last row of $A_{H'}$ have the form $b_{mk}(H)$ for some $i \in \mathbb{Z}^+$. Thus Equation (18) can be written as

\[s^{mk} = b_{mk} + s^m b_{mk-1} + \cdots + s^{m(k-1)}b_m.\]  

(19)
Since \( b_i(H_{sc}') = b_{mi}(H') = b_{mi}(H) \) and \( H_{sc}' \) has height \( k \), the copy-paste matrix of \( H_{sc}' \) is given by

\[
A_{H_{sc}'} = \begin{bmatrix}
0_{k-1,1} & I_{k-1,k-1} & \cdots & b_{2m} & b_m
\end{bmatrix}.
\] (20)

Using Equation (19), a simple calculation verifies that \( s^m \) is an eigenvalue of \( A_{H_{sc}'} \).

Similarly, if \( s \) is an eigenvalue of \( A_{H_{sc}'} \), then \( w = [1 \ s^2 \ \cdots \ s^{k-1}]^T \) is an eigenvector associated to \( s \) where \( km = M \). Since the only possible nonzero entries in the last row of \( A_{H'} \) have the form \( b_{mi}(H) \), that is, \( b_{M-j}(H) = b_{mk-j}(H) = 0 \) for \( j \neq im \), we have

\[
b_{mk} + b_{M-1}s^{1/m} + b_{M-2}s^{2/m} + \cdots + b_{M-(m-1)}s^{1-1/m} \\
b_{m(k-1)}s + b_{M-(m+1)}s^{1+1/m} + \cdots + b_{M-(2m-1)}s^{2-1/m} \\
\vdots \\
+ b_ms^{k-1} + \cdots + b_1s^{k-1/m} = s^k,
\] (21)
because the coefficients of all non-integer powers of \( s \) in the above expression are zero.

Another straightforward calculation verifies that

\[
\begin{bmatrix}
1 & s^{1/m} & s^{2/m} & \cdots & s & \cdots & s^2 & \cdots & s^{k-1/m}
\end{bmatrix}^T
\]
is an eigenvector of \( A_{H'} \) with corresponding eigenvalue \( s^{1/m} \), where Equation (21) gives the last entry in the matrix computation.

Part(c): We will only prove the result for \( T_{H_{sc}'} \). (The case of \( T_{H_{sc}} \) is similar.) Because \( T_{H_{sc}} \) has exponential growth with growth rate \( \lambda = \rho(A_{H_{sc}'}) \), there exist positive constants \( \alpha_1 \) and \( \alpha_2 \) such that

\[
\alpha_1 \lambda^n \leq f(T_{H_{sc}'}, n) \leq \alpha_2 \lambda^n.
\]

Thus, using part (b) of Observation 4.5,

\[
f(T_{H_{sc}'} , n) = \sum_{i=0}^{n} c_i(H_{sc}') = \sum_{i=0}^{n} c_{im}(T_{H'}) \leq \sum_{i=0}^{nm} c_i(T_{H'}) = f(T_{H'}, nm).
\]

By Observation 4.5 part (b), it also follows that

\[
f(T_{H'}, nm) = c_0(T_{H'}) \\
+ (c_1(T_{H'}) + \cdots + c_m(T_{H'})) \\
+ (c_{m+1}(T_{H'}) + \cdots + c_{2m}(T_{H'})) \\
\vdots \\
+ (c_{(n-1)m+1}(T_{H'}) + \cdots + c_{nm}(T_{H'})) \\
\leq m(c_0(T_{H'}) + c_m(T_{H'})) + c_{2m}(T_{H'}) + \cdots + c_{nm}(T_{H'})) \\
= m(c_0(T_{H_{sc}'}) + c_1(T_{H_{sc}'}) + \cdots + c_n(T_{H_{sc}'})) \\
= mf(T_{H_{sc}'}, n).
\] (22)
Therefore, 
\[ f(T_{H_w}, n) \leq f(T_{H'}, nm) \leq mf(T_{H_w}, n). \]
Hence, 
\[ \alpha_1 \lambda^n \leq f(T_{H'}, nm) \leq (m \alpha_2) \lambda^n, \]
and so 
\[ \alpha_1 (\lambda^{1/m})^{nm} \leq f(T_{H'}, nm) \leq (m \alpha_2) (\lambda^{1/m})^{nm}. \]
It now follows that \( T_{H'} \) has exponential growth with growth rate \( \sqrt[\lambda]{\lambda} \) which equals \( \rho(A_{H'}) \) by (b). Again, from a similar argument, it follows that \( T_{H} \) has the same growth rate \( \rho(A_{H'}). \)

Part (d): Since \( T_{H'} \) and \( T_{H} \) have the same growth rate, namely \( \rho(A_{H}) = \rho(A_{H'}) \), it follows from Equation (17) that \( T_{H} \) has growth rate \( \rho(A_{H}) = \rho(A_{H'}) \).

To summarize, we have proved our main result:

**Theorem 2.6.** Let \( T_{H} \) be a copy-paste tree with generator \( H \) and copy-paste matrix
\[
A_{H} = \begin{bmatrix}
0_{N-1,1} & I_{N-1,N-1} & \cdots & b_N \\
b_1 & b_2 & \cdots & b_N \\
\end{bmatrix}.
\]
Then \( T_{H} \) has exponential growth with growth rate \( \rho(A_{H}) \).

5. **Examples**

Next, we calculate the growth rates of the copy-paste trees considered throughout this paper.

**Example 5.1.** Let \( T \) be the infinite full binary tree (Example 2.4) with generators \( H_1 \) and \( H_2 \) shown in Figure 2. Note that \( A_{H_1} = \begin{bmatrix} 2 \end{bmatrix} \) and \( A_{H_2} = \begin{bmatrix} 0 & 1 \\ 4 & 0 \end{bmatrix} \). As expected, \( T \) has exponential growth with growth rate \( \rho(A_{H_1}) = \rho(A_{H_2}) = 2. \)

**Example 5.2.** Let \( H \) be the generator of the copy-paste tree \( T_{H} \) of Example 2.5 (see Figure 3). Note that \( A_{H} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix} \). By Theorem 2.6, \( T_{H} \) has exponential growth with growth rate \( \rho(A_{H}) \approx 1.3803. \)

**Example 5.3.** Let \( H \) be the generator of the copy-paste tree \( T_{H} \) of Example 4.1 (depicted in Figure 6). Note that \( N = 6, M = 4 \), and \( A_{H'} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 2 & 0 & 1 & 0 \end{bmatrix} \). (Since \( N > M \), it is convenient to compute the growth rate by using the smaller-sized matrix \( A_{H'} \).) Then \( T_{H} \) has exponential growth with growth rate \( \rho(A_{H'}) = \sqrt{2}. \)
Our next example illustrates a class of trees called **shuffled copy-paste trees**, which are obtained by permuting components of the shells of a given copy paste tree. Such trees obviously have identical growth functions to the original copy-paste tree.

**Example 5.4.** Consider the generator described in Example 2.5, however, to simplify the construction, we include a vertex in \( H \) at the midpoint of \((r, w)\). To build \( T'_H \), realize that the components of the shells of \( T_H \) can be permuted to obtain a new tree. In particular, at \( S_2 \), permute the edge on the right with the wedge shape on the left to obtain a new tree \( T'_H \) that is not self-similar.

![Figure 8. The permuted tree \( T'_H \) is not self-similar. It has the same growth rate as \( T_H \).](image)

Clearly, one could also create a shuffled copy-paste tree from \( T_H \) in Example 5.4 that does not have finitely many cones, by producing a tree having non-branching paths of any integer length having the same growth rate as \( T_H \).

**References**


