On the SIG dimension of trees under $L_\infty$ metric

L. Sunil Chandran\textsuperscript{1}, Rajesh Chitnis\textsuperscript{2}, and Ramanjit Kumar\textsuperscript{1}

\textsuperscript{1} Indian Institute of Science, Dept. of Computer Science and Automation, Bangalore–560012, India. email: ramanjikumar,sunil@csa.iisc.ernet.in
\textsuperscript{2} Chennai Mathematical Institute, Siruseri, Chennai–603103, India. email: rajesh@cmi.ac.in

Abstract. Let $P$, where $|P| \geq 2$, be a set of points in $d$-dimensional space with a given metric $\rho$. For a point $p \in P$, let $r_p$ be the distance of $p$ with respect to $\rho$ from its nearest neighbour in $P$. Let $B(p, r_p)$ be the open ball with respect to $\rho$ centered at $p$ and having the radius $r_p$. We define the sphere-of-influence graph (SIG) of $P$ as the intersection graph of the family of sets $\{B(p, r_p) \mid p \in P\}$. Given a graph $G$, a set of points $P_G$ in $d$-dimensional space with the metric $\rho$ is called a $d$-dimensional SIG representation of $G$, if $G$ is the SIG of $P_G$.

It is known that the absence of isolated vertices is a necessary and sufficient condition for a graph to have an SIG representation under $L_\infty$ metric in some space of finite dimension. The SIG dimension under $L_\infty$ metric of a graph $G$ without isolated vertices is defined to be the minimum positive integer $d$ such that $G$ has a $d$-dimensional SIG representation under the $L_\infty$ metric. It is denoted as $SIG_\infty(G)$.

We study the SIG dimension of trees under $L_\infty$ metric and answer an open problem posed by Michael and Quint (Discrete applied Mathematics: 127, pages 447-460, 2003). Let $T$ be a tree with at least two vertices. For each $v \in V(T)$, let leaf-degree($v$) denote the number of neighbours of $v$ that are leaves. We define the maximum leaf-degree as $\alpha(T) = \max_{v \in V(T)} \text{leaf-degree}(v)$. Let $S = \{v \in V(T) \mid \text{leaf-degree}(v) = \alpha\}$. If $|S| = 1$, we define $\beta(T) = \alpha(T) - 1$. Otherwise define $\beta(T) = \alpha(T)$. We show that for a tree $T$, $SIG_\infty(T) = \lceil \log_2(\beta + 2) \rceil$ where $\beta$ is not of the form $2^k - 1$, for some positive integer $k \geq 1$. If $\beta = 2^k - 1$, then $SIG_\infty(T) \in \{k, k+1\}$. We show that both values are possible.

Keywords: Sphere of Influence Graphs, Trees, $L_\infty$ norm, Intersection Graphs.

1 Introduction

Let $x$ and $y$ be two points in $d$-dimensional space. For any vector $z$, let $z[i]$ denote its $i^{th}$ component. For any positive integer $n$, let the set $\{1, 2, \ldots, n\}$ be denoted by $[n]$. The distance $\rho(x, y)$ between $x$ and $y$ with respect to the $L_k$-metric (where $k \geq 1$ is a positive integer) is defined to be $\rho_k(x, y) = (\sum_{j=1}^{d} |x[j]| -
\[ y[j]|^k \cdot \frac{1}{2}. \] Note that \( \rho_2(x, y) \) corresponds to the usual notion of the Euclidean distance between the two points \( x \) and \( y \). The distance between \( x \) and \( y \) under the \( L_\infty \)-metric \( \rho_\infty(x, y) \) is defined as \( \max\{ |x[i] - y[i]| : i = 1, 2, \ldots, d \} \). In this paper for the most part we will be concerned about the distance under \( L_\infty \) metric and therefore we will abbreviate \( \rho_\infty(x, y) \) as \( \rho(x, y) \) in our proofs. Also, \( \log \) will always refer to logarithm to the base 2.

1.1 Open Balls and Closed Balls:

For a point \( p \) in \( d \)-dimensional space and a positive real number \( r \), the open ball \( B(p, r) \) under a given metric \( \rho \), is a subset of \( d \)-dimensional space defined by \( B(p, r) = \{ x \mid \rho(x, p) < r \} \). For a point \( p \) in \( d \)-dimensional space and a positive real number \( r \), the closed ball \( C(p, r) \) under a given metric \( \rho \), is a subset of \( d \)-dimensional space defined by \( C(p, r) = \{ x \mid \rho(x, p) \leq r \} \). An open ball in \( d \)-dimensional space, with respect to \( L_\infty \) metric centred at \( p \) and with radius \( r \) is actually a “\( d \)-dimensional cube” defined as the cartesian product of \( d \) open intervals namely \( (p[1] - r, p[1] + r), (p[2] - r, p[2] + r), \ldots \) and \( (p[d] - r, p[d] + r) \). In notation, \( B(p, r) = \prod_{i=1}^{d} (p[i] - r, p[i] + r) \) where \( \prod \) denotes the cartesian product.

1.2 Maximum leaf-degree of a tree

Let \( T = (V, E) \) be an (unrooted) tree with \( |V| \geq 2 \). A vertex \( x \) of \( T \) is called a leaf, if \( \text{degree}(x) = 1 \). For a vertex \( x \in V \), let \( P(x) = \{ y \in V \mid y \text{ is adjacent to } x \text{ in } T \text{ and } y \text{ is a leaf} \} \). We define the maximum leaf degree \( \alpha \) of \( T \) as \( \alpha(T) = \max_{x \in V(T)} |P(x)| \). For our proof it is convenient to visualise the tree \( T \) as a rooted tree. Therefore we define a special rooted tree \( T' \) corresponding to \( T \), by carefully selecting a root, as follows: Let \( z \in V \) be such that \( |P(z)| = \alpha \). Let \( z' \in P(z) \). Let \( T' \) be the rooted tree obtained from \( T \), by fixing \( z' \) as root. In a rooted tree, a vertex is called a ‘leaf’, if it has no children. For \( x \in V \), let \( L(x) = \{ y \in V(T) \mid y \text{ is a child of } x \text{ in } T' \text{ and } y \text{ is a leaf} \} \). We define \( \beta(T) = \max_{x \in V(T)} |L(x)| \). The relation between \( \alpha(T) \) and \( \beta(T) \) is summarized below. While \( \beta(T) \) has the interpretation given above in terms of \( t \) he special rooted tree \( T' \), we take the following as the formal definition of \( \beta(T) \).

**Definition 1.** Let \( T \) be a tree of at least 2 vertices and \( \alpha(T) \) be the maximum leaf-degree of \( T \). Let \( S = \{ v \in V(T) \mid v \text{ is a vertex of maximum leaf degree, i.e. } |P(v)| = \alpha \} \). Then we define \( \beta(T) = \alpha(T) \) if \( |S| \geq 2 \) and \( \beta(T) = \alpha(T) - 1 \), if \( |S| = 1 \).

Clearly for any tree with at least 2 vertices \( \alpha(T) \geq 1 \). Moreover If \( \alpha(T) = 1 \), then \( |S| \geq 2 \) and therefore \( \beta(T) \geq 1 \) for all trees \( T \) with at least 2 vertices.
2 SIG-representation and SIG dimensions

Let $\mathcal{P}$, where $|\mathcal{P}| \geq 2$, be a set of points in $d$-dimensional space with a given metric $\rho$. For a point $p \in \mathcal{P}$, let $r_p$ be the distance of $p$ from its nearest neighbour in $\mathcal{P}$ with respect to $\rho$. Let $B(p, r_p)$ be the open ball with respect to $\rho$ centered at $p$ and having a radius $r_p$. We define the sphere-of-influence graph, $SIG$, of $\mathcal{P}$ as the intersection graph of the family of sets $\{B(p, r_p) \mid p \in \mathcal{P}\}$ i.e the graph will have a vertex corresponding to each set and two vertices will be adjacent if and only if the corresponding sets intersect. Given a graph $G$, a set of points $\mathcal{P}_G$ in $d$-dimensional space (with the metric $\rho$) is called a $d$-dimensional $SIG$ representation of $G$, if $G$ is the $SIG$ of $\mathcal{P}_G$. Note that if $\mathcal{P}$ is a $d$-dimensional $SIG$ representation of $G$ then we are associating to each vertex $x$ of $G$ a point $p(x)$ in $\mathcal{P}$. Given a graph $G$, the minimum positive integer $d$ such that $G$ has a $d$-dimensional $SIG$ representation (with respect to the metric $\rho$) is called the $SIG$ dimension of $G$ (with respect to the metric $\rho$) and is denoted by $SIG_\rho(G)$. It is known that the absence of isolated vertices is a necessary and sufficient condition for a graph to have an $SIG$ representation under $L_\infty$ metric in some space of finite dimension. The $SIG$ dimension under $L_\infty$ metric of a graph $G$ without isolated vertices is defined to be the minimum positive integer $d$ such that $G$ has a $d$-dimensional $SIG$ representation under the $L_\infty$ metric. It is denoted as $SIG_\infty(G)$. In this paper we may sometimes abbreviate this as $SIG(G)$

3 Literature Survey

Toussaint introduced Sphere of Influence graphs to model situations in pattern recognition and computer vision in [2], [3] and [4]. Graphs which can be realised as $SIG$ graphs in the Euclidean plane are considered in [5], [6], [7] and [8]. $SIG$ graphs in general metric spaces are considered in [9].

Toussaint has used the Sphere of Influence graphs under $L_2$-metric to capture low-level perceptual information in certain dot patterns. It is argued in [1] that Sphere of Influence graphs under the $L_\infty$-metric perform better for this purpose. Also, several results regarding $SIG_\infty$ dimension are proved in [1]. Bounds for the $SIG_\infty$ dimension of complete multipartite graphs are considered in [10].

4 Our result

We answer the following open problem regarding $SIG_\infty$ dimension of trees posed in [1].

Problem: (given in page 458 of [1]) Find a formula for the $SIG_\infty$ dimension of a tree (say in terms of its degree sequence and graphical parameters).
In this paper we prove the following theorem:

**Theorem:** For any tree $T$ with at least 2 vertices, $\text{SIG}_\infty(T) = \lceil \log_2(\beta + 2) \rceil$ where $\beta = \beta(T)$, provided $\beta$ is not of the form $2^k - 1$, for some positive integer $k \geq 1$. If $\beta = 2^k - 1$, then $\text{SIG}_\infty(T) \in \{k, k+1\}$. In this case both values can be achieved.

**Remark:** Seeing Theorem 1, the following obvious questions may come naturally to the reader’s mind.

1. What happens if closed balls are considered instead of open balls?
2. What happens if we consider some of the other well-known metrics such as $L_2$ metric instead of $L_\infty$ metric?

The first question can be answered easily by combining the results of Michael and Quint [9] with that of Jacobson, Lipman and Mcmorris [7]. Let $M$ be a normed linear space with metric $\rho$ with dimension at least 2. Michael and Quint [9] proved that a tree $T$ has a $\text{SIG}$ representation in the space $M$ using closed balls only if $T$ has a perfect matching. Now it is proved in [7] that a tree with a perfect matching has a $\text{SIG}$ representation using closed balls in the plane. (In fact the authors considered $L_2$-norm in [7], but their proof can be easily adapted to work in the case of $L_\infty$-norm also.) Thus we see that when we use closed balls, a tree either has a $\text{SIG}$ representation in a space of dimension at most 2, or it is not possible to get an $\text{SIG}$ representation in any dimension.

To answer the second question we have to consider 2 cases: (1) When we use closed balls: By the result of [9], mentioned in the above paragraph, we cannot get an $\text{SIG}$ representation of $T$, in any dimension, irrespective of the norm we use, if $T$ does not have a perfect matching. If $T$ has a perfect matching, we can modify the proof of [7] to show that $T$ has an $\text{SIG}$ representation on the plane, for other well-known norms such as $L_2$, $L_3$ etc. (2) When we use open balls: We say that a graph $G$ has a $\{K(1,1), K(1,2)\}$-factor if $G$ has a spanning subgraph such that each connected component of this subgraph is isomorphic to either a $K(1,1)$ (i.e. an edge) or to a $K(1,2)$ (i.e. a path on 3 vertices). Michael and Quint [9] proved that if $M$ is a strictly curved linear space with dimension at least 2, then a tree $T$ has an $\text{SIG}$-representation in the space $M$ using open balls, if and only if $T$ has a $\{K(1,1), K(1,2)\}$-factor. Since a space with $L_p$ norm (with dimension at least 2) is a strictly curved linear space for $1 < p < \infty$, we infer that if $T$ has a $\{K(1,1), K(1,2)\}$ then it has a $\text{SIG}$-representation using open balls in 2-dimensional space itself, whereas if $T$ does not have a $\{K(1,1), K(1,2)\}$-factor then $T$ cannot have any $\text{SIG}$-representation using open balls in space of any dimension, under $L_p$-metric, $1 < p < \infty$. (Also see Theorem 2 of [7]).

## 5 Lower Bound for $\text{SIG}_\infty(T)$

**Lemma 1.** For all trees $T$, $\text{SIG}_\infty(T) \geq \lceil \log_2(\beta + 1) \rceil$. 

**Proof.** If $|V(T)| = 2$, then it is a single edge and in this case $SIG_\infty (T) = 1$ and so the theorem is true in this case. Now let us assume that $V(T) \geq 3$. Let $t = SIG_\infty (T)$. Consider the special rooted tree $T'$ corresponding to $T$ defined in section 1.2. Let $z \in V$ be such that $|L(z)| = \beta$ in the rooted tree $T'$. Let $L(z) = \{y_1, y_2, \ldots, y_\beta\}$. Consider a $SIG$ representation $P_T$ of the tree $T$ in $t$-dimensional space under $L_\infty$ metric. From the definition of $SIG$ representation it is clear that each vertex $x$ has to be adjacent to every vertex $y$ such that $y$ is nearest point of $x$ in $P_T$. Since for each $y \in L(z)$, $z$ is the only adjacent vertex it follows that $z$ is the unique nearest point to $y$ for each $y \in L(z)$.

**Claim.** For each $y \in L(z)$, $Vol(B(y, r_y) \cap B(z, r_z)) \geq (r_z)^t$ where $Vol$ denotes the volume.

Following notation from section 1, $y[i]$ and $z[i]$ will denote the $i$th co-ordinate of $y$ and $z$ respectively. As $z$ is the nearest to $y$ we have $r_y = \rho(y, z)$. From this we can infer that $r_z \leq \rho(y, z) = r_y$. Since $\rho(y, z) = r_y$ we have by the definition of $L_\infty$ metric, $|y[i] - z[i]| \leq r_y$ for $1 \leq i \leq t$. Without loss of generality, we may assume that $z$ is the origin, i.e. $z[i] = 0$ for $1 \leq i \leq t$. This means that $|y[i]| = |y[i] - z[i]| \leq \rho(y, z) = r_y$. Now consider the projection of $B(y, r_y)$ and $B(z, r_z)$ on the $i$th axis. Clearly these projections are the open intervals $(y[i] - r_y, y[i] + r_y)$ and $(-r_z, r_z)$ respectively. We claim that the length of the intersection of these two intervals is at least $r_z$. To see this we consider 2 cases: If $y[i] \leq 0$, then since $|y[i]| \leq r_y$, $y[i] + r_y \geq 0$. Then since $r_z \leq r_y$ the interval $(-r_z, 0)$ is contained in the interval $(y[i] - r_y, y[i] + r_y)$. If $y[i] > 0$, since $|y[i]| \leq r_y$, $y[i] - r_y \leq 0$. Then since $r_z \leq r_y$ the interval $(0, r_z)$ is contained in the interval $(y[i] - r_y, y[i] + r_y)$. It follows that the length of $(y[i] - r_y, y[i] + r_y) \cap (-r_z, r_z)$ is at least $r_z$. It is easy to see that $B(y, r_y) \cap B(z, r_z) = \prod_{i=1}^{t} [(y[i] - r_y, y[i] + r_y) \cap (-r_z, r_z)]$ where $\prod$ stands for the cartesian product. From this it is clear that the $Vol(B(y, r_y) \cap B(z, r_z)) \geq (r_z)^t$. 

Let $\overline{T}$ be the parent of $z$ in $T'$. Note that $z$ always has a parent in $T'$. This is because $z$ cannot be the root of $T'$, since by the way the root of $T'$ is selected, the root can have only one child and this child cannot be a leaf since $V(T) \geq 3$. Note that $\{\overline{T}, y_1, y_2, \ldots, y_\beta\}$ is an independent set in $T$ and therefore for $y, y' \in \{\overline{T}, y_1, y_2, \ldots, y_\beta\}$, $B(y, r_y) \cap B(y', r_y') = \emptyset$. Now, noting that $Vol(B(z, r_z)) = (2r_z)^t$ we get $(2r_z)^t = Vol(B(z, r_z)) \geq Vol(B(\overline{T}, r_{\overline{T}}) \cap B(z, r_z)) + \sum_{y \in L(z)} Vol(B(y, r_y) \cap B(z, r_z))$. Now using the Claim and noting that $|L(z)| = \beta$, we get $2^t (r_z)^t \geq \beta (r_z)^t + Vol(B(\overline{T}, r_{\overline{T}}) \cap B(z, r_z))$. Since $B(z, r_z)$ and $B(\overline{T}, r_{\overline{T}})$ are open balls and $\overline{T}$ is adjacent to $z$, $Vol(B(z, r_z) \cap B(\overline{T}, r_{\overline{T}})) > 0$. We infer that $2^t > \beta$ and therefore $2^t \geq (\beta + 1)$ since $\beta$ and $2^t$ are both integers. Hence $t \geq \lceil \log(\beta + 1) \rceil$. 

\[\square\]
6 Upper Bound for SIG dimension of trees under $L_\infty$ metric

6.1 Basic Notation

For a non-leaf vertex $x$ of the rooted tree $T'$, let $C(x)$ denote the children of $x$. Let $L(x) = \{y \in C(x) \mid y$ is a leaf in $T'\}$. If $L(x) \neq \emptyset$, then we define $A(x) = C(x) - L(x)$. If $L(x) = \emptyset$, then select a vertex ‘$y_x$’ from $C(x)$ and call it a “pseudo-leaf” of $x$. In this case we define $A(x) = C(x) - \{y_x\}$. We call elements of $A(x)$ as “normal” children of $x$.

6.2 Some more notation under $L_\infty$ metric

Edges and corners of $B(p, r)$: Let $S = \{-1, +1\}^d$ be the set of all $d$-dimensional vectors with each component being either -1 or +1. Then the set $K(p, r) = \{p + r.S \mid s \in S\}$ is the set of corners of $B(p, r)$. Thus if $q \in K(p, r)$, then $\forall i \in [d]$ we have $q[i] = p[i] + r$ or $q[i] = p[i] - r$. Let $q$ and $q'$ be two corners of $B(p, r)$ such that they differ in exactly one co-ordinate position. A line segment between two such corners of $B(p, r)$ is said to be an edge of $B(p, r)$. Let $q$ and $q'$ be two corners of $B(p, r)$ such that the line segment between them defines an edge. Also $i$ be such that $q[i] \neq q'[i]$. Then clearly, $\{q[i], q'[i]\} = \{p[i] - r, p[i] + r\}$. Note that each corner $q$, belongs to exactly $d$-edges. We denote these edges as $A_1(q), A_2(q), \ldots, A_d(q)$, where $A_i(q)$ is the line segment between $q$ and another corner $q'$ such that $q$ and $q'$ differ only in the $i$th co-ordinate.

The shifting operation: Let $q$ be a corner of $B(p, r)$ so that $q = p + r.S$ for some $S \in S$. Consider the edge $A_i(q)$. Let $z$ be a point on $A_i(q)$. Note that for $j \neq i (1 \leq j \leq d)$, $z[j] = q[j]$. Let $\{e_1, e_2, \ldots, e_d\}$ be the canonical basis of $\mathbb{R}^d$. Now for $j \neq i$ and $\delta > 0$ define $z(j, \delta)$ to be the point $z + S[j].\delta.e_j$. We say that the point $z(j, \delta)$ is obtained by shifting $z$ along the $j$th axis by the distance $\delta$.

Crossing Edges: A point $p$ is said to be inside a open ball $B$ if $p \in B$, otherwise it is said to be outside $B$. Let $B_1$ and $B_2$ be two open balls such that $B_1 \cap B_2 \neq \emptyset$. An edge of $B_1$ is said to be a crossing edge with respect to the ball $B_2$ if one of the endpoint of this edge is inside $B_2$ and the other is outside.

6.3 Algorithm to give radius and position vector to each vertex

Let $d = \lfloor \log_2(\beta + 2) \rfloor$. (Note that since $\beta \geq 1, d \geq 2$) The following algorithm associates with each vertex $v$ of $T'$, a point in $d$-dimensional space $p(v)$ called the position of $v$ and a positive real number $r(v)$ called the radius of $v$. We will also associate with $v$ another positive real number $R(v)$, called the super-radius of $v$ by the following rule: If $v$ is a leaf then $R(v) = r(v)$, else $R(v) = 2r(v)$. The open ball $B(p(v), r(v))$ will be named the ball associated with $v$ and will
be denoted by $B(v)$. Also $B(p(v), R(v))$ will be called the super-ball associated with $v$ and will be denoted by $S(v)$. We use $K(u)$ to denote the set of corners of $B(u)$. Also, if $v$ is a normal child of its parent, the algorithm associates a number $J(v)$ to remember the axis along which shifting was done to get $p(v)$. (See step 3.2.2 of the algorithm for more details)

**Algorithm**

**INPUT:** The rooted tree $T'$ obtained from $T$ in section 1.2

**OUTPUT:** Two functions $p : V(T) \rightarrow \mathbb{R}^d$ and $r : V(T) \rightarrow \mathbb{R}^+$.

**Step 1:** For the root $x$, $p(x) = \emptyset$ and $r(x) = 1$.

**Step 2:** For the unique child $x'$ of $x$, $p(x') = \emptyset$ and $r(x') = \frac{r(x)}{8(|A(x)| + 1)} = \frac{1}{8}$ as $A(x) = \emptyset$.

**Step 3:** Suppose $u$ is a non-root vertex for which $r(u)$ and $p(u)$ is already defined by the algorithm.

**Step 3.1:** (Defining $r(y)$ for $y \in C(u)$) For each $y \in C(u)$ do:
- If $y$ is a leaf, then $r(y) = r(u)$
- Else $r(y) = \frac{r(u)}{8(|A(u)| + 1)}$

**Step 3.2:** (Defining $p(y)$ for $y \in C(u)$) For each $y \in C(u)$ do:
- Let $u'$ be the parent of $u$.
- Let $K'(u) = K(u) - B(u')$.
  **Step 3.2.1:** Defining $p(y)$ for $y \in C(u) - A(u)$
  Assign a point from $K'(u)$ to $p(y)$ such that for $y, y' \in C(u) - A(u)$ we have $p(y) \neq p(y')$ if $y \neq y'$.
  (See Lemma 7 for the feasibility of this step).

**Step 3.2.2:** Defining $p(y)$ for $y \in A(u)$
- If $u \notin A(u')$ (i.e. if $u$ is a pseudo-leaf)
  then let $q \in K'(u) - \{p(y) : y \in C(u) - A(u)\}$.
  (See Lemma 7 for the feasibility of this step).
  Let $q^1$ be the other end-point of the edge $A_1(q)$.
  Let $A(u) = \{u_1, u_2, \ldots, u_t\}$ where $t = |A(u)|$.
  Let $p^i = q + (\frac{r(u)}{16(\ell + 1)}) \frac{q^1 - q}{2r(u)} \quad \forall i \in [t]$.
  ($\frac{q^1 - q}{2r(u)}$ is the unit vector along edge $A_1(q)$ from $q$ to $q^1$)

**Shifting $p^i$ along axis 2 to get $p(u_i)$**
(Note that $d \geq 2$. Therefore we have at least 2 axes.)
Define $p(u_i) = p^i(2, \frac{r(u)}{16(\ell + 1)}) \quad \forall i \in [t]$. 
Set $J(u_i) = 2 \ \forall \ i \in [t].$

Else if $u \in A(u')$ (i.e. if $u$ is not a pseudo-leaf)
Let $q \in K(u) \cap B(u').$
(See Lemma 7 for the feasibility of this step)
Let $l = J(u)$

Let $A(u) = \{u_1, u_2, \ldots, u_t\}$ where $t = |A(u)|.$
Let $p^i = q + \left(\frac{r(u)}{2}\right)(1 + \frac{1}{t+1})\frac{q - q'}{2r(u)} \ \forall \ i \in [t].$
\((\frac{q - q'}{2r(u)})\) is the unit vector along edge $A_l(q)$ from $q$ to $q'\)

Let $j \neq l,$ where $j \in [d].$
(Note that $d \geq 2.$ Therefore we have atleast 2 axes.)
For each $u_i \in A(u),$ set $J(u_i) = j.$

**Shifting $p^i$ along $J(u_i) = j^{th}$ axis to get $p(u_i)$**
Define $p(u_i) = p^i(j, \frac{r(u)}{16(t+1)}) \ \forall \ i \in [t].$

6.4 Some comments on the Algorithm

Note that if a vertex $v$ is a normal child of its parent $u$ i.e. if $v \in A(u),$ then the algorithm in step 3.2.2 assigns the position $p(v)$ in one of the two possible ways depending on whether $u$ is a pseudo-leaf or not. In both the cases, the procedure is somewhat similar: We carefully select a corner $q$ of $B(u),$ then select a suitable edge $A_l(q)$ incident on $q,$ locate a suitable position on $A_l(q)$ and then shift this position by distance $\frac{r(u)}{2} = \frac{r(u)}{16|A(u)|+1},$ along a chosen axis $j = J(v) \neq l.$

For the rest of the proof, for a vertex $v \in A(u),$ we say that $v$ is attached to the corner $q$ of $B(u)$ and the edge $A_l(q)$ if the algorithm selects the corner $q$ and the edge $A_l(q)$ in order to find $p(v).$

Let $A(u) = \{u_1, u_2, \ldots, u_t\}$ and let $v = u_i.$ Let $v$ be attached to the corner $q$ of $B(u)$ and the edge $A_l(q).$ Clearly $q = p(u) + r(u).S_1$ for some $S_1 \in S.$ We can describe the co-ordinates of $p(v)$ in terms of co-ordinates of $q$ as follows:
If $k \notin \{l, j\},$ then $p(v)[k] = q[k].$
For the $j^{th}$ co-ordinate we have $p(v)[j] = q[j] + S_1[j].\frac{r(u)}{2}.$
For the $l^{th}$ co-ordinate we have $p(v)[l] = q[l] - S_1[l].\frac{r(u)}{2}(1 + \frac{1}{t+1}).$

This is summarized in the following lemma:

**Lemma 2.** Let $v \in A(u)$ and $|A(u)| = t.$ Let $v$ be attached to the corner $q$ of $B(u)$ and the edge $A_l(q).$ Since $q \in K(u),$ we have $q = p(u) + r(u).S_1$ for
some $S_1 \in \mathcal{S}$. If $\{e_1,e_2,\ldots,e_d\}$ is the canonical basis of $\mathbb{R}^d$, then $p(v) = p(u) + r(u).S_1 + S_1[j].\frac{r(v)}{r(u)}e_j - S_1[l].\frac{r(u)}{r(v)}(1 + \frac{1}{r(u)})e_l$ for some $i \in [l]$ and $j = J(v) \neq l$. As a consequence, note that $\rho(p(u),p(v)) = |p(u)[j] - p(v)[j]| = r(u) + \frac{r(v)}{2}$.

**Lemma 3.** Let $v \in A(u)$ be attached to corner $q \in K(u)$. Then no vertex in $C(u) - A(u)$ will be assigned to the corner $q$.

**Proof.** If $u$ is a pseudo-leaf, then recalling step 3.2.2 of the algorithm, we know that $q \in K'(u) - \{p(y) \mid y \in C(u) - A(u)\}$ and therefore the lemma is true. If $u$ is not a pseudo-leaf, then recalling step 3.2.2 of the algorithm, we know that $q \in K(u) \cap B(u')$. So, $q \notin K'(u)$. But from step 3.2.1 of the algorithm, each vertex in $C(u) - A(u)$ gets assigned to a corner from $K'(u)$. \qed

### 6.5 Correctness of the Algorithm

In this section we provide the Lemmas to establish that the algorithm given in section 6.3 does not get stuck at any step and thus assigns to each vertex a radius and a position vector.

**Lemma 4.** Let $u'$ be the parent of $u$. If $u \in C(u') - A(u')$ i.e. $u$ is either a leaf or a pseudo-leaf, then exactly 1 corner of $B(u)$ lies inside $B(u')$. Moreover, let $p(u)$ be the corner of $B(u')$ given by $p(u') + r(u').S_1$ where $S_1 \in \mathcal{S}$. Then the corner of $B(u)$ which is inside $B(u')$ is given by $p(u) - S_1.r(u)$.

**Proof.** Let $\mathcal{S} = \{-1,+1\}^d$. We shift co-ordinates so that $p(u') = \vec{0}$. Since $u \in C(u') - A(u')$, by step 3.2.1 of the algorithm, $p(u)$ gets assigned to one of the corners of $B(u')$. So, we have $p(u) = p(u') + r(u').S_1 = r(u')S_1$ for some $S_1 \in \mathcal{S}$. Now consider a general corner $q$ of $B(u)$. Since $q \in K(u)$, we know that $q = p(u) + r(u)S_2$ for some $S_2 \in \mathcal{S}$. So, $q = r(u')S_1 + r(u)S_2$. For $q$ to be inside $B(u')$, we require $r(u') > \rho(p(u'),q)$ i.e. $r(u') > \rho(\vec{0},r(u').S_1 + r(u).S_2)$. We know that $r(u) > 0$. If there is any co-ordinate position $n \in [d]$ such that $S_1[n] = S_2[n]$, then $p(u'[n],q) = r(u'[n]) + r(u) > r(u')$. So, we infer that for $q$ to be inside $B(u')$, it is necessary that $S_2 = -S_1$. Also, if $S_2 = -S_1$, then $\rho(p(u'),q) = |r(u') - r(u)| = r(u') - r(u) < r(u')$. Therefore, $q$ is inside $B(u')$ if and only if $S_2 = -S_1$. Clearly, given $S_1 \in \mathcal{S}$ there is a unique $S_2 \in \mathcal{S}$ such that $S_2 = -S_1$. So, there is exactly one corner of $B(u)$ inside $B(u')$ and it is given by $p(u) - S_1.r(u)$. \qed
 Lemma 5. Let $u'$ be the parent of $u$. If $u \in A(u')$ i.e. if $u$ is a normal child of $u'$, then exactly 2 corners of $B(u)$ lies inside $B(u')$. Further, these 2 corners form an edge of $B(u)$. Let $u$ be attached to corner $q$ of $B(u')$ and edge $A_i(q)$ where $q = p(u') + r(u'). S_1$ with $S_1 \in S$. Then the 2 corners of $B(u)$ inside $B(u')$ are given by $p(u) + r(u). S' \leq S''$ with $S', S'' \in S$ where $S' = -S_1$ and $S''$ is the string which differs from $S_1$ in all co-ordinates except the $l$th co-ordinate.

Proof. Let $S = \{-1,+1\}^d$ and $\{e_1, e_2, \ldots, e_d\}$ be the canonical basis of $\mathbb{R}^d$. Let $|A(u')| = t$. Without loss of generality, let $p(u') = \overline{0}$, where $u = u'$. If $u$ be attached to the corner $q \in K(u')$ and edge $A_i(q)$. Let $q = p(u') + r(u'). S_1$ where $S_1 \in S$. Thus by Lemma 2, $p(u) = q + S_1[j]. \frac{r(u)}{2}. e_j - S_1[l]. \frac{r(u)}{2}. (1 + \frac{i}{l+1}). e_l = r(u). S_1 + S_1[j]. \frac{r(u)}{2}. e_j - S_1[l]. \frac{r(u)}{2}. (1 + \frac{i}{l+1}). e_l$ for some $i \in [t]$ and $j \neq l$. Consider any general corner $s$ of $B(u')$. Then $s = p(u) + r(u). S_2$ for some $S_2 \in S$. So,

$$s = r(u). S_1 + S_1[j]. \frac{r(u)}{2}. e_j - S_1[l]. \frac{r(u)}{2}. (1 + \frac{i}{l+1}). e_l + r(u). S_2$$

For $s$ to be inside $B(u')$, we need $r(u') > \rho(p(u'), s)$ i.e. $r(u') > \rho(\overline{0}, s)$.

1. $s[l] < r(u')$
   $$|s[l]| = |S_1[l]. r(u') - S_1[l]. (\frac{r(u)}{2}. (1 + \frac{i}{l+1}) + S_2[l]. r(u))| \leq |r(u') - (\frac{r(u)}{2}). (1 + \frac{i}{l+1})| + |S_2[l]. r(u)| = |\frac{r(u)}{2}. (1 - \frac{i}{l+1})| + |r(u)|. Now, i \in [t] and r(u) = \frac{r(u')}{8(l+1)} \leq \frac{r(u')}{8}. Therefore, |s[l]| \leq |\frac{r(u)}{2}. (1 - \frac{i}{l+1})| + |r(u)| \leq |\frac{r(u')}{2}| + |\frac{r(u')}{8}| < r(u')$$
   So, independent of value of $S_2[l]$, we get that $|s[l]| < r(u')$.

2. $s[j] < r(u')$ if and only if $S_2[j] = -S_1[j]$
   If $S_1[j] = S_2[j]$, then $p(u', q) \geq r(u') + \frac{3r(u)}{2} > r(u')$. Hence, $S_1[j] \neq S_2[j]$ and $s[j] = r(u') - \frac{r(u)}{2} < r(u')$.

3. $s[k] < r(u')$ if and only if $S_2[k] = -S_1[k] \forall k \in [d], k \notin \{l,j\}$
   Since $r(u) > 0$, we must have $S_1[k] \neq S_2[k] \forall k \notin \{l,j\}$. So, $s[k] = r(u') - r(u) < r(u') \forall k \notin \{l,j\}$

From the above we can conclude that for given $S_1 \in S$, the corner $s = p(u) + S_2, r(u)$ is inside $B(u')$ if and only if $S_2[n] \neq S_1[n] \forall n \neq l$. We infer that the strings $S', S''$ (where $S' = -S_1$ and $S''$ is the string which differs from $S_1$ in all co-ordinates except the $l$th co-ordinate) correspond to the two corners of $B(u)$ inside $B(u')$.

Lemma 6. Let $u$ be any non-root non-leaf vertex. Let $K'(u) = K(u) - B(u')$ where $u'$ is parent of $u$. Then, if $u$ is a pseudo-leaf we have $|K'(u)| > \beta \geq |C(u) - A(u)|$ and if $u$ is a normal child we have $|K'(u)| \geq \beta \geq |C(u) - A(u)|$
Proof. $|C(u) - A(u)|$ is the number of children of $u$ that are not “normal”. Recall from section 1.2 that $\beta \geq 1$. If $u$ has a child which is a leaf, then $|C(u) - A(u)| \leq \beta$.

Else, $|C(u) - A(u)| = 1 \leq \beta$.

If $u$ is a “pseudo-leaf” of its parent $u'$, then by Lemma 4 and recalling that $d = \lceil \log_2(\beta + 2) \rceil$, $|K'(u)| = |K(u)| - 1 = 2^d - 1 \geq \beta + 1 > \beta$.

If $u$ is not a “pseudo-leaf” of its parent $u'$, then by Lemma 5 and recalling that $d = \lceil \log_2(\beta + 2) \rceil$, $|K'(u)| = |K(u)| - 2 = 2^d - 2 \geq \beta$. 

\[\square\]

Lemma 7. The algorithm given in the previous section runs correctly i.e. it does not get stuck at any of the three points which give a reference to this Lemma. So each vertex gets a position vector and a radius, when the algorithm terminates.

Proof. By Lemma 6, we know that $|K'(u)| \geq |C(u) - A(u)|$. So, distinct points in $C(u) - A(u)$ can be assigned distinct points from $K'(u)$ in step 3.2.1 of the algorithm.

If $u$ is a “pseudo-leaf” of its parent $u'$, then by Lemma 6, $|K'(u)| > \beta \geq |C(u) - A(u)|$. So, $(K'(u) - \{ p(y) : y \in C(u) - A(u) \}) \neq \emptyset$ as is required in the “if” part if step 3.2.2 of the algorithm.

If $u$ is not a “pseudo-leaf” of its parent $u'$, then by Lemma 5, there are two corners of $B(u)$ inside $B(u')$ and therefore $K(u) \cap B(u') \neq \emptyset$ as is required in “else” part of step 3.2.2 of the algorithm. 

\[\square\]

6.6 \textbf{T is the intersection graph of the family \{B(u) \mid u \in V(T)\}}

Recall that for every vertex $u$ in $V(T)$, $B(u) = B(p(u), r(u))$ where $p(u)$ and $r(u)$ are the position vector and radius computed by the algorithm. In this section we show that $T$ is the intersection graph of the family $\{B(u) \mid u \in V(T)\}$.

Lemma 8. Let $uv \in E(T)$. Then, $B(u) \cap B(v) \neq \emptyset$.

Proof. Immediate from Lemma 4 and Lemma 5.

\[\square\]

Lemma 9. Let $v, w$ be children of $u$. Then $S(v) \cap S(w) = \emptyset$.

Proof. We have the following 3 cases:

Case 1:
Both $v, w \in C(u) - A(u)$. Then both $v$ and $w$ must be leaves. Thus, $R(v) = R(w) = r(u) = r(w) = r(u)$. By step 3.2.1 of the algorithm, $p(v)$ and $p(w)$ will
ge placed at distinct corners of $B(u)$. So $\rho(p(v), p(w)) = 2r(u) = R(v) + R(w)$ and hence $S(v) \cap S(w) = \emptyset$.

**Case 2:**
Both $v, w \in A(u)$ and let $|A(u)| = t$. So, $r(v) = \frac{r(u)}{8(t+1)} = r(w)$ and $R(v) = 2r(v)$ and $R(w) = 2r(w)$. Following terminology of section 6.4, let $q \in K(u)$ be the corner and let $A_t(q)$ be the edge to which $v$ and $w$ are attached. Let $q = p(u) + S_1.r(u)$ where $S_1 \in S$. Applying Lemma 2, we have $p(v) = q + S_1[j] \cdot \frac{r(u)}{2}.e_j - S_1[l] \cdot \frac{r(u)}{2}.(1 + \frac{k}{t+1}).e_t$ for some $i \in [t]$. Hence, $r(v) = r(w)$ and hence $S(v) \cap S(w) = \emptyset$.

**Case 3:**
Let $v \in C(u) - A(u)$ and $w \in A(u)$. Let $|A(u)| = t$. So, $r(v) = \frac{r(u)}{8(t+1)} < \frac{r(u)}{8}$ and $R(v) = 2r(v) < r(u)$ and $R(w) = 2r(w) < r(u)$. If $v$ is a leaf, then $r(v) = R(v) = r(u)$. If $v$ is a pseudo-leaf, then $R(v) = 2r(v)$ and $r(v) = \frac{r(u)}{2(t+1)} < \frac{r(u)}{8}$. In either case, $R(v) \leq r(u)$.

Translate the co-ordinates so that $p(v) = v$. By step 3.2.1 of the algorithm, $v$ gets assigned to a corner $q_1 \in K(u)$. So, $p(v) = p(u) + S_2.r(u) = S_2.r(u)$ where $S_2 \in S$. Let $q \in K(u)$ be the corner and $A_0(q)$ be the edge to which $w$ is attached. Let $q = p(u) + S_1.r(u) = S_1.r(u)$. From Lemma 2, we have $p(w) = S_1.r(u) + S_1[j] \cdot \frac{r(u)}{2}.e_j - S_1[l] \cdot \frac{r(u)}{2}.(1 + \frac{k}{t+1}).e_t$ for some $i \in [t]$ and $j \neq l$.

**Claim.** $q \neq q_1$

Recall that at step 3.2.1 of the algorithm the corner $q_1$ of $B(u)$ given to $p(v)$ is from $K'(u)$ since $v \in C(u) - A(u)$. Now, we know that $w \in A(u)$. In step 3.2.2 of the algorithm, if $u$ is a pseudo-leaf, then $q$ is chosen from $K'(u) - \{p(y) : y \in C(u) - A(u)\}$. If $u$ is not a pseudo-leaf, then $q$ is chosen from $K(u) \cap B(u')$ which is disjoint from $K'(u)$. So, in both the cases we get that $q \neq q_1$.

In view of the above claim, $q$ and $q_1$ (and therefore $S_1$ and $S_2$) differ in at least one co-ordinate say $k$. If $k \notin \{l, j\}$, then considering distance along $k^{th}$ co-ordinate, $\rho(p(v), p(w)) \geq |p(v)[k] - p(w)[k]| \geq 2r(u) \geq R(v) + R(w)$. If $k = j$, then considering distance along $j^{th}$ co-ordinate, we have $\rho(p(v), p(w)) \geq |p(v)[j] - p(w)[j]| \geq 2r(u) + \frac{r(u)}{2} > 2r(u) \geq R(v) + R(w)$. If $k = l$, the considering distance along $l^{th}$ co-ordinate we get $\rho(p(v), p(w)) \geq |p(v)[l] - p(w)[l]| \geq 2r(u) - \frac{r(u)}{2}(1 + \frac{1}{t+1}) = r(u) + (r(u) - \frac{r(u)}{2}(1 + \frac{1}{t+1})) = r(u) + \frac{r(u)}{2}(1 - \frac{1}{t+1}) \geq r(u) + \frac{r(u)}{2(t+1)}$ as $i \in [t]$. Hence, recalling that $r(w) = \frac{r(u)}{2(t+1)}$ we get $\rho(p(v), p(w)) \geq r(u) + \frac{r(u)}{2(t+1)} > r(u) + 2r(w) = R(v) + R(w)$. Hence $S(v) \cap S(w) = \emptyset$.

**Lemma 10.** If $v$ is a child of $u$, then $S(v) \subset S(u)$.
**Proof.** We have the following 2 cases depending on what type of child \( v \) is.

**Case 1 :** \( v \in C(u) - A(u) \). Recall that if \( v \) is a leaf then \( R(v) = r(v) = r(u) \) and if \( v \) is a pseudoleaf then \( R(v) = 2r(v) < r(u) \). In both cases, we have \( r(u) \geq R(v) \)

Also note that \( R(u) = 2r(u) \) as \( u \) is not a leaf. By step 3.2.1 of the algorithm \( \rho(p(u), p(v)) = r(u) \). Let \( p(y) \) be any point in \( S(v) \). Then, \( \rho(p(v), p(y)) < R(v) \).

By triangle inequality, \( \rho(p(u), p(y)) \leq \rho(p(u), p(v)) + \rho(p(v), p(y)) < r(u) + R(v) \leq 2r(u) = R(u) \). Hence \( S(v) \subseteq S(u) \).

**Case 2 :** \( v \in A(u) \). Let \( |A(u)| = t \). Then, \( r(v) = \frac{r(u)}{8(t+1)} < \frac{r(u)}{8} \) and thus \( R(v) = 2r(v) < \frac{r(u)}{4} \).

Also note that \( R(u) = 2r(u) \) as \( u \) is not a leaf. By Lemma 2, \( \rho(p(u), p(v)) = r(u) + \frac{r(u)}{16} \). Let \( p(y) \) be any point in \( S(v) \). Then, \( \rho(p(v), p(y)) < R(v) \). By triangle inequality, \( \rho(p(u), p(y)) \leq \rho(p(u), p(v)) + \rho(p(v), p(y)) < r(u) + \frac{r(u)}{2} + R(v) < r(u) + \frac{r(u)}{16} + \frac{r(u)}{4} = 2r(u) = R(u) \).

Hence \( S(v) \subseteq S(u) \).

\( \square \)

**Lemma 11.** Let \( u \) be any vertex of \( T' \). For every descendant \( v \) of \( u \), \( B(v) \subseteq S(u) \).

**Proof.** Let the depth of a vertex \( x \) in \( T' \) be the number of vertices in the path from the root to \( x \). We prove the lemma by induction on the depth. Let \( D \) be the maximum depth among all vertices of \( T' \). Clearly any vertex of depth \( D \) must be a leaf. Now, if \( u \) is a leaf then its only descendant is itself and the lemma holds trivially. So, we infer that the lemma holds for all vertices of depth \( D \). We take this as the base case of the induction. Now suppose that the lemma holds true for all vertices of depth greater than \( h \) where \( 1 < h < D \). Now, let \( u \) be a vertex of depth \( h \). If \( u \) is a leaf then the lemma holds trivially. Otherwise, let the children of \( u \) be \( \{u_1, u_2, \ldots, u_m\} \). Clearly, depth of \( u_i \) equals \( (h+1) \forall i \in [m] \).

Now, let \( v \) be a descendant of \( u \) such that \( v \neq u \). If \( v = u_j \) for some \( j \in [m] \), then by Lemma 10 we have \( B(v) \subseteq S(v) \subseteq S(u) \). Else, \( v \neq u_j \forall j \in [m] \).

Then \( v \) is a descendant of \( u_k \) for some \( k \in [m] \). Recalling that depth of \( u_k \) equals \( (h+1) \) and using induction hypothesis, we have \( B(v) \subseteq S(u_k) \). By Lemma 10, \( S(u_k) \subseteq S(u) \). Therefore, \( B(v) \subseteq S(u_k) \subseteq S(u) \).

\( \square \)

**Lemma 12.** Let \( u \) be the parent of \( v \) and \( w \) be the parent of \( w \). Then \( B(u) \cap S(w) = \emptyset \)

**Proof.** Let \( S = \{-1, +1\}^d \). We shift co-ordinates so that \( p(u) = \overline{0} \). Note that \( v \) cannot be a leaf. So, we have the following four cases.

**Case 1 :** \( v \) is a pseudoleaf

**Case 1.1 :** \( w \in C(v) - A(v) \):

So \( p(v) \) is a corner of \( B(u) \) and is hence given by \( p(v) = p(u) + r(u).S_1 = r(u).S_1 \)
for some $S_1 \in \mathcal{S}$. By step 3.2.1 of the algorithm, we know that $p(w)$ belongs to $K'(v)$. Thus $p(w) = p(v) + r(v).S_2 = r(u).S_1 + r(v).S_2$ for some $S_2 \in \mathcal{S}$. Since $p(w) \in K'(u)$ we can infer that $S_2 \neq S_1$ by Lemma 4. That is, there is some $k \in [d]$ such that $S_1[k] = S_2[k]$. Considering distance along $k^{th}$ co-ordinate, we have $\rho(p(u), p(w)) = \rho(\overrightarrow{0}, r(u).S_1 + r(v).S_2) = r(v) + r(v)$. Now, if $w$ is a leaf then $R(w) = r(w) = r(v)$. Else, if $w$ is a pseudo-leaf, then $r(w) \leq \frac{r(v)}{2}$ and $R(w) = 2r(w)$. In both cases, we are assured that $r(v) \geq R(w)$. Hence, $\rho(p(u), p(w)) = r(w) + r(v) \geq r(u) + R(w)$. Therefore $B(u) \cap S(w) = \emptyset$.

**Case 1.2 : $w \in A(v)$ :**

Let $|A(v)| = t$. Note that from step 3.1, $r(w) = \frac{r(v)}{s(t+1)} < \frac{r(v)}{8}$ and thus $r(v) > 2r(w) = R(w)$. Also as in the previous case, $p(v) = p(u) + r(u).S_1 = r(u).S_1$ for some $S_1 \in \mathcal{S}$. Suppose that $w$ gets attached to corner $q \in K(v)$ and edge $A_i(q)$. (See section 6.4). Clearly $q = p(v) + S_2.r(v) = r(u).S_1 + r(v).S_2$ for some $S_2 \in \mathcal{S}$. By Lemma 2, we have $p(w) = r(u).S_1 + r(v).S_2 - \frac{r(v)}{2}(1 + \frac{1}{r+1}).S_2[i].e_i + S_2[j].\frac{(u-w)}{2}.e_j$ for some $i \in [t]$ and $j \neq l$. By step 3.2.2 of the algorithm, we know that $q \in K'(v) - \{p(y) \mid y \in C(v) - A(v)\}$. Also from Lemma 4 we know that the only corner of $B(v)$ that is inside $B(u)$ is given by $p(v) - S_1.r(v)$ and since $q \in K'(v)$ is outside $B(u)$ we can infer that $S_2 \neq S_1$, i.e. there is $k \in [d]$ such that $S_1[k] = S_2[k]$. If $k \notin \{l, j\}$, then considering the distance along $k^{th}$ co-ordinate to $\rho(p(u), p(w)) = \rho(\overrightarrow{0}, p(w)) \geq r(u) + r(v) > r(u) + R(w)$. If $k = j$, then $\rho(p(u), p(w)) \geq r(u) + r(v) + \frac{r(v)}{2} > r(u) + r(v) > r(u) + R(w)$. If $k = l$, then $\rho(p(u), p(w)) \geq r(u) + r(v) - \frac{r(v)}{2}(1 + \frac{1}{r+1}) = r(u) + \frac{r(v)}{2}(1 - \frac{1}{r+1})$. Note that $i \in [t]$ implies that $\frac{r(v)}{2}(1 - \frac{1}{r+1}) = \frac{r(v)}{2(r+1)} > 2r(w) = R(w)$. So, $\rho(p(u), p(w)) \geq r(u) + \frac{r(v)}{2}(1 - \frac{1}{r+1}) \geq r(u) + R(w)$. Therefore $B(u) \cap S(w) = \emptyset$.

**Case 2 : $v$ is not a pseudoleaf i.e. $v \in A(u)$**

**Case 2.1 : $w \in C(v) - A(v)$ :**

If $w$ is leaf, then $R(w) = r(w) = r(v)$. If $w$ is a pseudo-leaf, then by step 3.1, we have $r(w) \leq \frac{r(v)}{8}$ and therefore $R(w) = 2r(w) < r(v)$. In both cases, $R(w) \leq r(v)$. Let $|A(u)| = t$. Suppose that $v$ gets attached to corner $q \in K(u)$ and edge $A_i(q)$. (See section 6.4). Clearly $q = p(u) + S_1.r(u) = S_1.r(u)$ for some $S_1 \in \mathcal{S}$. By Lemma 2, we have $p(v) = r(u).S_1 - \frac{r(v)}{2}(1 + \frac{1}{r+1}).S_1[i].e_i + S_1[j].\frac{(u-v)}{2}.e_j$ for some $i \in [t]$ and a suitably selected $j \neq l$. Also, step 3.2.1 of the algorithm assigns $p(w)$ to a corner of $B(v)$ and hence $p(w) = p(v) + S_2.r(v)$ for some $S_2 \in \mathcal{S}$. That is, $p(w) = r(v).S_2 + r(u).S_1 - \frac{r(v)}{2}(1 + \frac{1}{r+1}).S_1[i].e_i + S_1[j].\frac{(u-v)}{2}.e_j$. By step 3.2.1 of the algorithm, we know that $p(w) \in K'(v)$. Also, by Lemma 5, we know that the 2 corners of $B(v)$ that are in $B(u)$ are given by $p(v) + r(v).S'$ and $p(v) + r(v).S''$ where the strings $S'$ and $S''$ are such that $S' = S_2$ and $S''$ is the string which differs from $S_1$ in all co-ordinates other than $l^{th}$ co-ordinate. It follows that $\exists m \in [d], m \neq l$ such that $S_1[m] = S_2[m]$ since $p(w) = p(v) + S_2.r(v)$ is a corner of $B(v)$ outside $B(u)$. If $m = j$, then $\rho(p(u), p(w)) = \rho(\overrightarrow{0}, p(w)) \geq r(v) + r(u) + \frac{r(v)}{2} > r(v) + r(u) \geq R(w) + r(u)$. If $m \neq j$, then $\rho(p(u), p(w)) \geq r(v) + r(u) \geq R(w) + r(u)$. Therefore $B(u) \cap S(w) = \emptyset$. **
Case 2.2: \(w \in A(v)\):

Suppose that \(v\) gets attached to corner \(q \in K(u)\) and edge \(A_l(q)\). (See section 6.4). Clearly \(q = p(u) + r(u).S_1 = r(u).S_1\) for some \(S_1 \in S\). Suppose that \(w\) gets attached to corner \(q' \in K(v)\) and edge \(A_l(q')\) where \(l' = J(v)\). Clearly \(q' = p(v) + r(v).S_2\) for some \(S_2 \in S\). From step 3.2.2 of the algorithm, we know that \(q' \in K(v) \cap B(u)\). By Lemma 5, the 2 corners of \(B(v)\) which are inside \(B(u)\) are given by \(p(v) + S'.r(v)\) and \(p(v) + S''.r(v)\) where \(S'\) and \(S''\) are such that \(S' = -S_1\) and \(S''\) differs from \(S_1\) in all co-ordinates other than the \(l^{th}\) co-ordinate. Thus the two strings \(S_1\) and \(S_2\) are related as follows:

\[
S_2[m] \neq S_1[m], \forall m \neq l \text{ and } S_2[l] \text{ may or may not be equal to } S_1[l]
\]

Let \(|A(u)| = t\) and \(|A(v)| = t'\). By Lemma 2, \(p(v) = r(u).S_1 - \frac{r(u)}{t} e_i + S_1[j]. \frac{r(v)}{2} e_j\) for some \(i \in [t]\) and \(j = J(v) \neq l\). Again by Lemma 2, \(p(w) = p(v) + r(v).S_2 - \frac{r(v)}{t + 1} e_i + S_2[j']\) for some \(j' \in [t']\) and \(j' = J(w) \neq l'\). Replacing value of \(p(v)\) in previous equation, we get

\[
p(w) = r(u).S_1 - \frac{r(u)}{t + 1} e_i + S_1[j]. \frac{r(v)}{2} e_j + r(v).S_2 - \frac{r(v)}{t + 1} e_i + S_2[j']\]

Also, recall from step 3.2.2 of the algorithm that \(l' = J(v) = j\). Now, recalling that \(j \neq l\), and thus \(S_1[j] \neq S_2[j]\) and also \(j = l'\), we consider the distance between \(p(u)\) and \(p(w)\) along the \(j^{th}\) co-ordinate: we get \(\rho(p(u), p(w)) = \rho(\mathbf{0}, p(w)) \geq |r(u) + \frac{r(v)}{2} - r(v) + \frac{r(v)}{2}(1 + \frac{\gamma'}{t + 1})| = r(u) + \frac{r(v)}{2}(\frac{\gamma'}{t + 1})\). Now recall that \(R(w) = 2r(w) = 2\frac{r(v)}{2(\gamma' + 1)} \leq \frac{r(v)}{2(\gamma' + 1)}\) for any \(i' \in [t]\). Substituting this we get \(\rho(p(u), p(w)) > r(u) + R(w)\) and we infer that \(B(u) \cap S(w) = \emptyset\).

\[\square\]

Lemma 13. If \(uv \notin E(T)\), then \(B(u) \cap B(v) = \emptyset\)

Proof. First suppose that none of \(u,v\) is an ancestor of the other. Let \(z\) and \(w\) are their least common ancestor. Let \(z_a, z_v\) be children of \(z\) such that \(u\) is a descendant of \(z_a\) and \(v\) is a descendant of \(z_v\). By Lemma 11, \(B(u) \subset S(z_a)\) and \(B(v) \subset S(z_v)\). But by Lemma 9, \(S(z_a) \cap S(z_v) = \emptyset\). So, \(B(u) \cap B(v) = \emptyset\).

Otherwise, without loss of generality, let \(u\) be an ancestor of \(v\). Since \(uv \notin E(T)\) we know that \(v\) is not a child of \(u\). Consider the path from \(u\) to \(v\) in \(T\). Let \(u, w, y\) be the first three vertices of this path. Clearly \(v\) is a descendant of \(v\) and therefore by Lemma 11, \(B(v) \subset S(y)\). By Lemma 12, \(S(y) \cap B(u) = \emptyset\). So, \(B(v) \cap B(u) = \emptyset\)

\[\square\]
Lemma 14. \(T\) is the intersection graph of the family \(\{B(u) | u \in V(T)\}\)

Proof. Immediate from Lemma 8 and Lemma 13.

\[\square\]

6.7 \(T\) is the SIG graph of the family \(\{p(u) | u \in V(T)\}\)

Consider the set \(P\) in \(\mathbb{R}^d\) given by \(P = \{p(u) | u \in V(T)\}\) computed by the algorithm. For each \(u \in V(T)\), let \(r_u\) be distance of \(p(u)\) from its nearest neighbour(s) in \(P\).

Lemma 15. Let \(u \in V(T)\) be a non-leaf vertex. Then there is a child \(v\) of \(u\) such that \(\rho(p(u), p(v)) = r(u)\)

Proof. If \(u\) has a leaf-child then we are through since any leaf-child of \(u\) is placed at a corner of \(B(u)\). If \(u\) does not have a leaf-child, then recall that we had designated a special child \(v\) of \(u\) as a pseudo-leaf. This pseudo-leaf is placed at a corner of \(B(u)\) in step 3.2.1 of algorithm. So, \(\rho(p(u), p(v)) = r(u)\)

\[\square\]

Lemma 16. Let \(u \in V(T)\). Then there is a vertex \(w \in V(T)\) such that \(\rho(p(u), p(w)) = r(u)\)

Proof. If \(u\) is not a leaf, then we are through by Lemma 15. If \(u\) is a leaf, consider its parent \(w\). Step 3.2.1 of the algorithm places \(u\) at a corner of \(B(w)\). So, \(\rho(p(u), p(w)) = r(w) = r(u)\)

\[\square\]

Lemma 17. Let \(uv \in E(T)\). Then \(\rho(u, v) \geq \max\{r(u), r(v)\}\).

Proof. Without loss of generality, let \(u\) be the parent of \(v\). Clearly, \(r(u) \geq r(v)\).

If \(v \in C(u) - A(u)\), then \(\rho(p(u), p(v)) = r(u)\). If \(v \in A(u)\), then by Lemma 2,

\(\rho(p(u), p(v)) = r(u) + \frac{r(w)}{2} > r(u)\).

\[\square\]

Lemma 18. Let \(u \in V(T)\). Then \(r_u = r(u)\).

Proof. Let \(v \in V(T)\) be such that \(uv \notin E(T)\). Then, by Lemma 14, we have \(\rho(p(u), p(v)) \geq r(u) + r(v) > r(u)\). Let \(w \in V(T)\) such that \(uw \in E(T)\). Then, by Lemma 17, we have \(\rho(p(u), p(v)) \geq r(u)\). Also, Lemma 16 guarantees us a vertex \(y \in V(T)\) such that \(\rho(p(u), p(y)) = r(u)\). Therefore, the least distance of \(p(u)\) from its nearest neighbour(s) in \(P\) is \(r(u)\).

\[\square\]
Lemma 19. \(T\) is the SIG graph of the family \(\{p(u) | u \in V(T)\}\)

Proof. Immediate from Lemma 14 and Lemma 18. □

6.8 SIG dimension of trees under \(L_\infty\) metric

In the preceding sections we have shown that the set \(\mathcal{P} = \{p(u) | u \in V(T')\}\) given by the algorithm gives a \(d\)-dimensional SIG representation of \(T\) for \(d = \lceil \log_2(\beta + 2) \rceil\). Thus, we have \(\text{SIG}(T) \leq \lceil \log_2(\beta + 2) \rceil\) where \(\beta = \beta(T)\). Recall that by Lemma 1, \(\text{SIG}(T) \geq \lceil \log(\beta + 1) \rceil\). We note that \(\lceil \log_2(\beta + 1) \rceil = \lceil \log_2(\beta + 2) \rceil\) except when \(\beta\) is one less than a power of 2. Therefore we have the following theorem:

Theorem 1:
For any tree \(T\), \(\text{SIG}_\infty(T) \leq \lceil \log_2(\beta + 2) \rceil\) where \(\beta = \beta(T)\). If \(\beta\) is not of the form \(2^k - 1\), for some integer \(k \geq 1\), we have \(\text{SIG}_\infty(T) = \lfloor \log_2(\beta + 2) \rfloor\).

7 When \(\beta = 2^k - 1\) for some \(d \geq 1\)

By Theorem 1 and Lemma 1, we know that \((k + 1) = \lceil \log_2(\beta + 2) \rceil \geq \text{SIG}_\infty(T) \geq \lceil \log_2(\beta + 1) \rceil = k\). In this section we show that both values namely \(k\) and \(k + 1\) are achievable.

Example where \(\text{SIG}_\infty(T) = k\) with \(\beta(T) = 2^k - 1\)
Consider a star graph on \(\beta + 2\) vertices. Since this is the complete bipartite graph with one vertex on one part and \(\beta + 1\) vertices on the other part we denote it as \(K_{1,\beta+1}\). Recalling the definition of \(\beta\) from section 1.2, we note that \(\beta(K_{1,\beta+1}) = \beta\). From Theorem 7 of [1], we know that \(\text{SIG}_\infty(K_{1,\beta+1}) = \lceil \log_2(\beta + 1) \rceil = k\), since \(\beta = 2^k - 1\).

Example where \(\text{SIG}_\infty(T) = k + 1\) with \(\beta(T) = 2^k - 1\)
Consider the tree \(H\) illustrated in figure on \((2\beta + 7)\) vertices. we show that its SIG dimension is \(k + 1\), where \(\beta = 2^k - 1\).

Lemma 20. \(\text{SIG}_\infty(H) = (k + 1)\)

Proof. Recalling the definition of \(\alpha\) and \(\beta\) from section 1.2, we note that \(\alpha(H) = \beta(H) = 2^k - 1\). By Theorem 1, we have \(\text{SIG}_\infty(H) \leq \lceil \log(2^k - 1 + 2) \rceil = k + 1\). Now we will show that \(\text{SIG}_\infty(H) \geq k + 1\). In a SIG graph, every vertex is adjacent to all of its nearest neighbour. Since the only two neighbours of \(z\) are \(x\) and \(y\) either \(x\) or \(y\) or both must be nearest neighbour(s) of \(z\). Without loss
of generality, let $x$ be the nearest neighbour of $z$. Also, for each $x_i$, $1 \leq i \leq \beta$, $x$ is the nearest neighbour, since $x$ is the only vertex adjacent to $x_i$. Note that \{x, z, x_1, x_2, \ldots, x_\beta\} form an independent set and $x$ is the nearest neighbour of each vertex from the set \{z, x_1, x_2, \ldots, x_\beta\}. Let $\text{SIG}_\infty(H) = t$. Doing a similar analysis as in the proof of Lemma 1, we get $2^t > (\beta + 1)$ i.e $2^t > 2^k$. Therefore, $t \geq (k + 1)$. It follows that $!\text{SIG}_\infty(H) = k + 1$.

\[ \Box \]

References

1. T.S. Michael and Thomas Quint. Sphere of influence graphs and the $l_\infty$-metric. *Discrete Applied Mathematics*, 127:447–460, 2003.
2. G.T. Toussaint. Pattern recognition and geometric complexity. *Proceedings of the 5th International Conference on Pattern Recognition*, 1324–1347, 1980
3. G.T. Toussaint. Computational geometric problems in pattern recognition. *Pattern Recognition Theory and Applications, NATO Advanced Study Institute, Oxford Univ.*, 73–91, 1981
4. G.T. Toussaint. A graph-theoretic primal sketch. *Computational Morphology, Elsevier, Amsterdam*, 220–260, 1998
5. D. Avis and J. Horton. Remarks on the sphere of influence graph. *Discrete Geometry and Convexity, New York Academy of Sciences*, 323–327, 1985
6. F. Harary, M.S. Jacobson, M.J. Lipman and F.R. McMorris. Abstract sphere-of-influence graphs. *Mathl. Comput. Modelling*, 17 11: 77–84, 1993
7. M.S. Jacobson, M.J. Lipman and F.R. McMorris. Trees that are sphere-of-influence graphs. *Appl. Math. Lett.* 8 6: 89–94, 1995
8. M.J. Lipman. Integer realizations of sphere-of-influence graphs. *Congr. Numer.* 91: 63–70, 1991
9. T.S. Michael and T. Quint. Sphere of influence graphs in general metric spaces. *Mathl. Comput. Modelling*, 29 7: 45–53, 1999
10. E. Boyer, L. Lister and B. Shader. Sphere-of-influence graphs using the sup-norm. *Mathl. Comput. Modelling*, 32: 1071–1082, 2000