Taylor & Francis
Taylor & Francis Group

AKCE International Journal of Graphs and Combinatorics

ISSN: 0972-8600 (Print) 2543-3474 (Online) Journal homepage: https://www.tandfonline.com/loi/uakc20

Generating graceful unicyclic graphs from a given forest

G. Sethuraman & V. Murugan

To cite this article: G. Sethuraman & V. Murugan (2020) Generating graceful unicyclic graphs from a given forest, AKCE International Journal of Graphs and Combinatorics, 17:1, 592-605, DOI: 10.1016/j.akcej.2019.02.007

To link to this article: https://doi.org/10.1016/j.akcej.2019.02.007

AKCE INTERNATIONAL JOURNAL OF GRAPHS AND COMBINATORICS
2020, VOL. 17, NO. 1, 592–605
https://doi.org/10.1016/j.akcej.2019.02.007

Taylor & Francis
Taylor & Francis Group

AKCE International Journal of Graphs and Combinatorics

Generating graceful unicyclic graphs from a given forest

G. Sethuraman*, V. Murugan

Department of Mathematics, Anna University, Chennai, 600 025, India

Received 13 September 2018; received in revised form 14 February 2019; accepted 22 February 2019

Abstract

Acharya (1982) proved that every connected graph can be embedded in a graceful graph. The generalization of this result that, any set of graphs can be packed into a graceful graph was proved by Sethuraman and Elumalai (2005). Recently, Sethuraman et al. (2016) have shown that, every tree can be embedded in an graceful tree. Inspired by these fundamental structural properties of graceful graphs, in this paper, we prove that any acyclic graph can be embedded in an unicyclic graceful graph. This result is proved algorithmically by constructing graceful unicyclic graphs from a given acyclic graph. Our result strongly supports the Truszczynski's Conjecture that "All unicyclic graphs except the cycle C_n with $n \equiv 1$ or 2(mod 4) are graceful".

Keywords: Graceful tree; Graceful unicyclic graph; Graceful tree embedding; Graceful labeling; Graph labeling

1. Introduction

All the graphs considered in this paper are finite and simple graphs. The terms which are not defined here can be referred from [1]. In 1963, Ringel posed his celebrated conjecture, popularly called Ringel Conjecture [2], which states that, K_{2n+1}, the complete graph on $2n+1$ vertices can be decomposed into $2n+1$ isomorphic copies of a given tree with n edges. Kotzig [3] independently conjectured the specialized version of the Ringel Conjecture that the complete graph K_{2n+1} can be cyclically decomposed into $2n+1$ copies of a given tree with n edges. In an attempt to solve both the conjectures of Ringel and Kotzig, in 1967, Rosa, in his classical paper [4] introduced an hierarchical series of labelings called ρ, σ, β and α labelings as a tool to settle both the conjectures of Ringel and Kotzig. Later, Golomb [5] called β-labeling as graceful labeling, and now this term is being widely used. A function f is called graceful labeling of a graph G with q edges, if f is an injective function from $V(G)$ to $\{0, 1, 2, \ldots, q\}$ such that, when every edge (u, v) is assigned the edge label $|f(u) - f(v)|$, then the resulting edge labels are distinct. A graph which admits graceful labeling is called graceful graph. Further, Rosa [4] proved that "If a graph G with q edges has a graceful labeling then the complete graph K_{2q+1} can be cyclically decomposed into $2q+1$ copies of the graph G". This result subsequently induced the popular Graceful Tree Conjecture, which states that "Every tree is graceful". The Graceful Tree Conjecture appears to be hard and it remains open over 5 decades.

Peer review under responsibility of Kalasalingam University.
* Corresponding author.
E-mail address: sethu@annauniv.edu (G. Sethuraman).

https://doi.org/10.1016/j.akcej.2019.02.007

Please cite this article as: G. Sethuraman and V. Murugan, Generating graceful unicyclic graphs from a given forest, AKCE International Journal of Graphs and Combinatorics (2019), https://doi.org/10.1016/j.akcej.2019.02.007.

Rosa [4] also proved that the cycle C_n is graceful if and only if $n \equiv 0$ or 3(mod 4). In 1984, Truszczynski's [6] conjectured that, "All unicyclic graphs except the cycle C_n with $n \equiv 1$ or 2(mod 4) are graceful". The conjecture of Truszczynski is also hard as the Graceful Tree Conjecture. Not much results have been proved to support Truszczynski's Conjecture. Some of the interesting results supporting Truszczynski's Conjecture are listed below.

- Barientos [7] proved that a unicyclic graph in which the deletion of any edge on the cycle results in a caterpillar is graceful. [A tree is called a caterpillar, in which the removal of all the pendant vertices of the tree results in a path].
- Jaya Bagga and Arumugam [8] have constructed a graceful unicyclic graph from a special class of caterpillars.
- Figueroa-Centeno et al. [9] have provided an interesting construction to form a graceful unicyclic graph from a set of α-labeled trees with some special property [A graceful labeling f of a graph G is called an *α-labeling* if there exists an integer λ such that, $\min\{f(x), f(y)\} \leq \lambda < \max\{f(x), f(y)\}$ for each edge (x, y) in G].

For more details on the results supporting Truszczynski's Conjecture refer the dynamic survey on graph labeling by Gallian [10]. Structural characterization of graceful graphs appears to be one of the most difficult problems in graph theory. However, some interesting general structural properties of graceful graphs are established. Acharya [11] proved that every connected graph can be embedded in a graceful graph. In [12], Sethuraman and Elumalai generalized this result and they have shown that every set of graphs can be packed into a graceful graph. Recently, in [13] Sethuraman et al. have also shown that every tree can be embedded in a graceful tree. Inspired by these fundamental structural properties of graceful graphs, in this paper, we prove that any acyclic graph can be embedded in a graceful unicyclic graph. This result is proved algorithmically by constructing graceful unicyclic graphs from a given acyclic graph. More precisely, we prove a general result that from any given acyclic graph F containing n arbitrary trees, we construct graceful unicyclic graphs with cycle length that vary from 3 to $n + 1$. Our result strongly supports the Truszczynski's Conjecture.

2. Main result

In this section, we present our Embedding Algorithm, which generate graceful unicyclic graphs from any acyclic graph.

Embedding Algorithm
Input: A Forest F

Find the number of components of F. If $n \geq 1$ is the number of components of F, then denote the n components of F by $T_1, T_2, \ldots, T_n$.

Step 1: Construction of a Tree T from the Forest F

Consider the input forest $F = \langle T_1, T_2, \ldots, T_n \rangle$. For each i, $1 \leq i \leq n$, choose any vertex in T_i and name that vertex by u_i. For i, $1 \leq i \leq n - 1$, join the vertex u_i of T_i and the vertex u_{i+1} of T_{i+1} by a new edge. Denote the resulting tree thus obtained by T.

Step 2: Arrangement of vertices of T

Consider the tree T constructed by Step 1 as a rooted tree with the vertex u_1 of T_1 as its root. Find the number of level of T. If l is the number of level of T, then arrange the vertices of T in the following way.

First, arrange the children of the root vertex of T in the first level from left to right order based on the decreasing order of their degrees. Then, arrange the children of first level vertices in the following way.

1. If x and y are two vertices of the first level of T such that x appears to the left side of y, then arrange all the children of the vertex x on the left side of all the children of the vertex y in the second level.

Please cite this article as: G. Sethuraman and V. Murugan, Generating graceful unicyclic graphs from a given forest, AKCE International Journal of Graphs and Combinatorics (2019), https://doi.org/10.1016/j.akcej.2019.02.007.

2. Then, for each vertex x in the first level arrange the children of x in the second level based on the decreasing order of their degrees.

Continue the same process of arranging the children of the second level vertices and then third level vertices and so on.

Step 3: Defining bipartition of T

Count the number of vertices which appear in all the even levels of T and count the number of vertices which appear in all the odd levels of T. If r is the number of vertices which appear in all the even levels of T and s is the number of vertices which appear in all the odd levels of T, then name the vertices of T in the following way.

First name the root as x_0 (note $x_0 = u_1$) and name the remaining vertices of T at each level from left to right as follows.

For each level i, $1 \le i \le l$ and i is even, then name the vertices which appear in the level i of T from left to right, as

$$x_{\sum_{i=2}^{i-2} m_{i-2}+1}, x_{\sum_{i=2}^{i-2} m_{i-2}+2}, \ldots, x_{\sum_{i=2}^{i} m_i}. \tag{1}$$

Similarly, for each i, $1 \le i \le l$ and i is odd, name the vertices which appear in the level i from left to right, as

$$y_{\sum_{i=1}^{i-2} m_{i-2}+1}, y_{\sum_{i=1}^{i-2} m_{i-2}+2}, \ldots, y_{\sum_{i=1}^{i} m_i}. \tag{2}$$

Here in (1) and (2), $m_{-1} = 0$, $m_0 = 0$ and for i, $1 \le i \le l$, m_i denotes the number of vertices which appear in the level i of T.

In the above process of naming, the vertices already named with $u_1, u_2, \ldots, u_n$ also again named either with some x_i or with some y_j, for some i, j, such vertices are referred to by either of these two names according to convenience.

Collect all the vertices that appear in all the even levels of T as a set and denote it by X. Also collect all the vertices that appear in all the odd levels of T as a set and denote it by Y. Form a bipartition (X, Y) of T.

Step 4: Arrangement of vertices of X and Y

In the left side, arrange the vertices of X in the following way.
If x_i and x_j are two vertices of A such that $i < j$ then the vertex x_i appear above to the vertex x_j. Consequently, the vertex x_0 is the top most vertex and the vertex x_{r-1} is the bottom most vertex. Refer this arrangement of vertices as the top to bottom order. Similarly, arrange the vertices of Y in the top to bottom order on the right side in which the vertex y_1 is the topmost vertex and the vertex y_s is the bottommost vertex.

Step 5: Labeling the vertices and edges of the tree T
Step 5.1: Labeling the vertices of T

For each $x_i \in X$, $0 \le i \le r-1$, define $f(x_i) = i$ and
for each $y_i \in Y$, $1 \le i \le s$, define $f(y_i) = (s-i+1)r$.

Step 5.2: Labeling the edges of T

For every edge uv of T, define its edge label, $f'(uv) = |f(u) - f(v)|$.

Step 6: Construction of a larger tree T^* containing the input tree T as its subtree
Step 6.1: Defining initial labeled sets needed for constructing the tree T^*

For the tree T, define
Existing Vertex Label Set $V = V(T) = \{0, 1, \ldots, r, 2r, 3r, \ldots, rs\}$,
Existing Edge Label Set $E = E(T) = \{f'(e_1), f'(e_2), \ldots, f'(e_{s+r-1})\}$,

Please cite this article as: G. Sethuraman and V. Murugan, Generating graceful unicyclic graphs from a given forest, AKCE International Journal of Graphs and Combinatorics (2019), https://doi.org/10.1016/j.akcej.2019.02.007.

All Label Set $X = \{0, 1, 2, \dots, rs\}$,
Missing Vertex Label Set $V^c = X \setminus V$,
Missing Edge Label Set $E^c = (X \setminus \{0\}) \setminus E$.
[Note that $|V^c| = |E^c|$]
Initiate $T^* \leftarrow T$,
$V(T^*) \leftarrow V(T)$,
$E(T^*) \leftarrow E(T)$

Step 6.2: If $E^c = \phi$ and $V^c = \phi$ then go to Step 7.1.
Step 6.3: If $E^c \neq \phi$ and $V^c \neq \phi$ then do the following.

Arrange the elements in the sets V^c and E^c as
$V^c = \{a_1, a_2, \dots, a_d\}$ such that $a_1 < a_2 < \cdots < a_d$ and
$E^c = \{b_1, b_2, \dots, b_d\}$ such that $b_1 < b_2 < \cdots < b_d$.
For t, $1 \leq t \leq d$

Find $c_t = a_t - b_t$, add a new vertex with label a_t and join a new edge (a_t, c_t) between the vertex c_t and the new vertex a_t.
Update $T^* \leftarrow T^* + (a_t, c_t)$,
$V(T^*) \leftarrow V(T^*) \cup \{a_t\}$,
$E(T^*) \leftarrow E(T^*) \cup \{(a_t, c_t)\}$.
Delete a_t from V^c and b_t from E^c.
Update $t \leftarrow t + 1$.

Step 7: Constructing a graph G^* form the updated tree T^*
Step 7.1: If the updated tree T^* is obtained from Step 6.2 then do the following

Take a new vertex v, label it with $s + 2$ and join this vertex v with the vertex labeled 0 and the vertex labeled 1 of T^*.
$G^* \leftarrow T^* + \{(s + 2, 0), (s + 2, 1)\}$,
$V(G^*) \leftarrow V(T^*) \cup \{s + 2\}$,
$E(G^*) \leftarrow E(T^*) \cup \{(s + 2, 0), (s + 2, 1)\}$.

Step 7.2: If the updated tree T^* obtained from Step 6.3 then do the following
Step 7.2.1: When $n = 1$, then do the following

Find the vertex, u, which is the left most vertex in the last level l of the rooted tree $T = T_1$. Take a new vertex v and labeled it with $rs + f(u) + 1$. Join the new vertex v with the vertex u_1 and the vertex u of T contained in T^*.
Initiate $G_1 \leftarrow T^* + \{(rs + f(u) + 1, 0), (rs + f(u) + 1, f(u))\}$,
$V(G_1) \leftarrow V(T^*) \cup \{rs + f(u) + 1\}$,
$E(G_1) \leftarrow E(T^*) \cup \{(rs + f(u) + 1, 0), (rs + f(u) + 1, f(u))\}$.

Step 7.2.1.1: Construction of the graph G_1^* from the graph G_1 obtained from Step 7.2.1

For the graph G_1 obtained from Step 7.2.1, define
Existing Vertex Label Set $V(G_1) = \{0, 1, 2, \dots, rs, rs + f(u) + 1\}$,
Existing Edge Label Set $E(G_1) = \{1, 2, \dots, rs, rs + 1, rs + f(u) + 1\}$,
All Label Set $X(G_1) = \{0, 1, 2, \dots, rs + f(u) + 1\}$,
Missing Vertex Label Set $V(G_1)^c = X(G_1) \setminus V(G_i) = \{rs + 1, rs + 2, \dots, rs + f(u)\}$,
Missing Edge Label Set $E(G_1)^c = (X(G_1) \setminus \{0\}) \setminus E(G_1) = \{rs + 2, rs + 3, \dots, rs + f(u)\}$.
Initiate $G_1{}^* \leftarrow G_1$,
$V(G_1{}^*) \leftarrow V(G_1)$,
$E(G_1{}^*) \leftarrow E(G_1)$
While $E(G_1)^c \neq \phi$

Then, find $minE(G_1)^c = a$. Take a new vertex label it with a and join this vertex a with the vertex labeled 0 of G_1^*.
Update $G_1{}^* \leftarrow G_1{}^* + (0, a)$,
$V(G_1{}^*) \leftarrow V(G_1{}^*) \cup \{a\}$,
$E(G_1{}^*) \leftarrow E(G_1{}^*) \cup \{(0, a)\}$.
Delete a from $E(G_1)^c$.

Step 7.2.2: When $n \geq 2$, then do the following

For each i, $2 \leq i \leq n$,

Take a new vertex v, label it with $rs + f(u_i) + 1$ and join this vertex v with the vertex u_1 of T_1 and the vertex u_i of T_i contained in T^*.
Initiate $G_i \leftarrow T^* + \{(rs + f(u_i) + 1, 0), (rs + f(u_i) + 1, f(u_i))\}$,
$V(G_i) \leftarrow V(T^*) \cup \{rs + f(u_i) + 1\}$,
$E(G_i) \leftarrow E(T^*) \cup \{(rs + f(u_i) + 1, 0), (rs + f(u_i) + 1, f(u_i))\}$.

Step 7.2.2.1: Construction of the graph G_i^* from the graph G_i obtained from Step 7.2.2

For each i, $2 \leq i \leq n$
For the graph G_i obtained from Step 7.2, define
Existing Vertex Label Set $V(G_i) = \{0, 1, 2, \ldots, rs, rs + f(u_i) + 1\}$,
Existing Edge Label Set $E(G_i) = \{1, 2, \ldots, rs, rs + 1, rs + f(u_i) + 1\}$,
All Label Set $X(G_i) = \{0, 1, 2, \ldots, rs + f(u_i) + 1\}$,
Missing Vertex Label Set $V(G_i)^c = X(G_i) \setminus V(G_i) = \{rs + 1, rs + 2, \ldots, rs + f(u_i)\}$,
Missing Edge Label Set $E(G_i)^c = (X(G_i) \setminus \{0\}) \setminus E(G_i) = \{rs + 2, rs + 3, \ldots, rs + f(u_i)\}$.
Initiate $G_i{}^* \leftarrow G_i$,
$V(G_i{}^*) \leftarrow V(G_i)$,
$E(G_i{}^*) \leftarrow E(G_i)$
While $E(G_i)^c \neq \phi$

Then, find $minE(G_i)^c = a$. Take a new vertex label it with a and join this vertex a with the vertex labeled 0 of G_i^*.
Update $G_i{}^* \leftarrow G_i{}^* + (0, a)$,
$V(G_i{}^*) \leftarrow V(G_i{}^*) \cup \{a\}$,
$E(G_i{}^*) \leftarrow E(G_i{}^*) \cup \{(0, a)\}$.
Delete a from $E(G_i)^c$.

Note. For convenience hereafter a vertex in either T or T^* or G_i or $G_i{}^*$ is referred by its label. Similarly an edge in either T or T^* or G_i or $G_i{}^*$ is referred by its label. We make the following observations and prove the following lemmas to establish that the Embedding Algorithm indeed construct graceful unicyclic graphs from the input forest as the output.

Observation 2.1. *From Step 5 of the Embedding Algorithm, the vertices which appear on the left side part of the bipartition of the tree T receive consecutive vertex labels from 0 to $r - 1$ and the vertices which appear on the right side of the bipartition of the tree T receive the vertex labels of the form zr, $1 \leq z \leq s$.*

Observation 2.2. *From Step 2 and Step 5 of the Embedding Algorithm, observe that, in the bipartite graph T, all the children of the vertex i, $1 \leq i \leq r - 1$ are arranged consecutively from top to bottom based on decreasing order of their degrees in the right side just below the last child of the vertex $i - 1$. Similarly, observe that, all the children of the vertex zr, $1 \leq z \leq s - 1$ are arranged consecutively from top to bottom on the left side based on decreasing order of their degrees just below the last child of the vertex $(z + 1)r$.*

Please cite this article as: G. Sethuraman and V. Murugan, Generating graceful unicyclic graphs from a given forest, AKCE International Journal of Graphs and Combinatorics (2019), https://doi.org/10.1016/j.akcej.2019.02.007.

Observation 2.3. *From Observation 2.1, the Missing Vertex Label Set V^c defined in Step 6.1 consists of all the labels (integers) that lie in the interval $(zr, (z+1)r)$, for all z, $1 \le z \le s-1$. More precisely, the Missing Vertex Label Set $V^c = \{r+1, r+2, \dots, 2r-1, 2r+1, \dots, 3r-1, \dots, (s-1)r+1, \dots, rs-1\}$. If we arrange the elements of V^c as a sequence $\{r+1, r+2, \dots, 2r-1, 2r+1, \dots, 3r-1, \dots, (s-1)r+1, \dots, rs-1\} = (a_1, a_2, \dots, a_d)$, then any two consecutive terms, a_{i-1} and a_i, for $2 \le i \le d$ either both lie on the same interval say $(zr, (z+1)r)$ for some fixed z, $1 \le z \le s-1$ or $a_{i-1} \in (zr, (z+1)r)$ and $a_i \in ((z+1)r, (z+2)r)$ for some fixed z, $1 \le z \le s-2$. Thus, either $a_i = a_{i-1}+1$ or $a_i = a_{i-1}+2$.*

Observation 2.4. *From Observation 2.1, the labels of the edges that are incident at the vertex zr, $1 \le z \le s$ must lie on the interval $[(z-1)r, zr]$. Therefore, the maximum of the edge labels of all the edges that are incident at the vertex zr is less than the minimum of the edge labels of all the edges that are incident at the vertex $(z+1)r$, for every $z, 1 \le z \le s-1$. The missing edge labels at the vertex zr, for each z, $1 \le z \le s$, is the set of labels (integers) that lie on $[(z-1)r+1, zr]$ excluding the labels of the edges that are incident at the vertex zr and it is denoted by $MEL(zr)$. More precisely the set $MEL(zr) = \{(z-1)r+1, (z-1)r+2, \dots, zr\} \setminus \{$set of all labels of the edges that are incident at the vertex $zr\}$. Then we can describe the set $E^c = \bigcup_{z=1}^{s} MEL(zr) = \{b_1, b_2, \dots, b_d\}$. As T is a tree, $|V^c| = |E^c|$.*

Lemma 2.5. *The vertex labels as well as the edge labels of the tree T obtained in Step 5 of the Embedding Algorithm are all distinct.*

Proof. It follows from Observation 2.1, the labels of all the vertices of the tree T are distinct.

Claim. *The labels of all the edges of the tree T are distinct*

Since the labels of the vertices that lie on the left side part of the bipartition of T are consecutive from 0 to $r-1$, it follows that, the labels of the incident edges at each vertex zr, $1 \le z \le s$ are all distinct and these labels lie in the set $\{(z-1)r+1, (z-1)r+2, \dots, zr\}$. Further, for any two consecutive vertices on the right side part of the bipartition of T, say zr and $(z+1)r$, we have maximum over the labels of all the edges that are incident at the vertex $zr \le zr < zr+1 \le$ minimum over the labels of all the edges that are incident at the vertex $(z+1)r$, for z, $1 \le z \le s-1$. Thus, the labels of the edges that are incident at any two distinct vertices on the right side part of the bipartition of T are always distinct. This would imply that the labels of all the edges of the tree T are distinct. □

Lemma 2.6. *The sets V^c and E^c, defined in Step 6.1 of the Embedding Algorithm are empty if and only if the tree T is a star.*

Proof. Assume that the tree T is a star with $s+1$ vertices. Without loss of generality, we assume that $|X| \le |Y|$, where (X, Y) is the bipartition of the tree T. By Step 3 of the Embedding Algorithm, $|X| = r = 1$ and $|Y| = s$. After the execution of Step 5 of the Embedding Algorithm, Existing Vertex Label Set $V(T) = \{0, 1, 2, \dots, s\}$ and the Existing Edge Label Set $E(T) = \{1, 2, 3, \dots, s\}$. As the set $X = \{0, 1, 2, 3, \dots, s\}$, the set $V^c = X \setminus V = \phi$ and the set $E^c = (X \setminus \{0\}) \setminus E = \phi$.

Conversely, assume that each of the sets V^c and E^c, defined in the Step 6.1 of Embedding Algorithm is empty. We claim that the tree T is a star. Suppose that the tree T is not a star. Then, $2 \le |X| = r \le |Y|$. By using Step 5 of the Embedding Algorithm, the Existing Vertex Label Set $V(T) = \{0, 1, 2, \dots, r-1, r, 2r, \dots, rs\}$. Since $r \ge 2$, $V^c = X \setminus V = \{r+1, r+2, \dots, 2r-1, 2r+1, \dots, 3r-1, \dots, rs-1\} \ne \phi$. A contradiction to our assumption that $V^c = \phi$. Hence the input tree T must be a star. □

Lemma 2.7. *The label c_t defined in Step 6.3 of the Embedding Algorithm is a non-negative integer for all values of t and the label c_t exists as the vertex label of a vertex of the current tree T^* that is being used in the current execution of Step 6.3.*

Proof. Observe that Step 6.3 of the Embedding Algorithm is executed when the sets V^c and E^c are non-empty. Hence by Lemma 2.6, the tree T is not a star. Thus, $s \ge r \ge 2$. Further, for every t, $1 \le t \le d$, $a_t \in V^c$, $b_t \in E^c$, the label $c_t = a_t - b_t$ is found in Step 6.3.

Please cite this article as: G. Sethuraman and V. Murugan, Generating graceful unicyclic graphs from a given forest, AKCE International Journal of Graphs and Combinatorics (2019), https://doi.org/10.1016/j.akcej.2019.02.007.

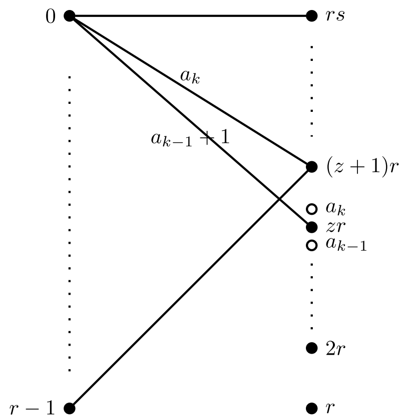

Fig. 1. The structure of the bipartite tree T under the Case 1.

Claim. *c_t is a non-negative integer*

To ascertain the claim we prove that $a_t \geq b_t$, for every t, $1 \leq t \leq d$ by using induction on t.

When $t = 1$, from Observation 2.3, we have $a_1 = r + 1, r \geq 2$ and from Observation 2.4, $b_1 \in \{1, 2, \ldots, r\}$. Hence $a_1 > b_1$.

We assume that the result is true for up to $t = k - 1$. That is, we assume that $a_t \geq b_t$, for each t, $1 \leq t \leq k - 1$.

We now prove that the statement is true for $t = k$. More precisely, we prove that $a_k \geq b_k$. Suppose that $a_k < b_k$. From Observation 2.3, a_{k-1} and a_k differ either by 1 or 2. It follows from our assumption ($a_k < b_k$) and the inductive assumption ($b_{k-1} \leq a_{k-1}$), that

$$b_{k-1} \leq a_{k-1} < a_k < b_k. \tag{3}$$

Since, either $a_k = a_{k-1} + 2$ or $a_k = a_{k-1} + 1$, we consider the following two cases.

Case 1. $a_k = a_{k-1} + 2$

Since b_{k-1} and b_k are consecutive missing edge labels and from Eq. (3), the labels of the sequence $C = (b_{k-1} + 1, \ldots, a_{k-1}, a_{k-1} + 1, a_{k-1} + 2 = a_k, a_k + 1, \ldots, b_k - 1)$ are consecutive existing edges labels that lie between b_{k-1} and b_k. For whatever may be the value of k, the sequence C always contains the labels $a_{k-1} + 1$ and a_k. Since a_{k-1} and a_k are consecutive missing vertex labels, the label $a_{k-1}+1$ must be an existing vertex label which appears on the right side of the bipartition of the tree T. From Step 5 of the Embedding Algorithm, $a_{k-1} + 1 = zr$, for some $z, 2 \leq z \leq s - 1$. As $a_{k-1} + 1$ belongs to C, the label $a_{k-1} + 1$ is also an existing edge label, the label $a_{k-1}+1$ appears as existing vertex label as well as existing edge label. As $a_{k-1}+1 = zr$, for some $z, 2 \leq z \leq s-1$, by Observation 2.4 the edge label $a_{k-1} + 1$ is only obtained from the edge connecting the vertex 0 and the vertex zr. The label $zr = a_{k-1} + 1$ is the maximum edge label obtained at the vertex zr. As $a_k = a_{k-1} + 2 = zr + 1$ also belongs to C, $zr + 1$ is the next existing edge label just after zr. As by Observation 2.4, the edge label $zr + 1 = a_k$ must be obtained at the edge connecting the vertex $(z+1)r$ and the vertex $r-1$, for $z, 2 \leq z \leq s-1$. Since $r-1$ is the bottommost vertex of the left side part of the bipartition of T and $r-1$ is also a child of the vertex $(z+1)r$, this would imply from our construction of the tree T, the vertex pr, for each i, $1 \leq p \leq z$ are pendant. Hence the only vertex adjacent with zr is 0. From Observation 2.4, the set $MEL(zr) = \{(z-1)r+1, (z-1)r+2, \ldots, zr-1 = a_{k-1}\}$. Therefore the label a_{k-1} is a missing edge label. Then from Eq. (3) that, $b_{k-1} \leq a_{k-1} < b_k$, we have $b_{k-1} = a_{k-1}$. (See Fig. 1.)

As $k - 1 = |\{b_1, b_2, \ldots, b_{k-1}\}|$. By Observation 2.4, $|\{b_1, b_2, \ldots, b_{k-1}\}| = |\bigcup_{p=1}^{z} MEL(pr)|$. Since the vertex pr is pendant, for each p, $1 \leq p \leq z$, $|MEL(pr)| = r-1$. Thus, $k-1 = z(r-1)$. Since $k-1 = |\{a_1, a_2, \ldots, a_{k-1}\}|$ and by Observation 2.3, $|\{a_1, a_2, \ldots, a_{k-1}\}| = |\{r + 1, r + 2, \ldots, 2r - 1, 2r + 1, \ldots, 3r - 1, \ldots, (z - 1)r + 1, \ldots, zr - 1 = a_{k-1}\}|$, we have $k - 1 = (z - 1)(r - 1)$. But from the above discussion $k - 1 = z(r - 1)$. This leads to a contradiction. Therefore, our assumption that $a_k < b_k$ is wrong. Hence $a_k \geq b_k$.

Case 2. $a_k = a_{k-1} + 1$

Please cite this article as: G. Sethuraman and V. Murugan, Generating graceful unicyclic graphs from a given forest, AKCE International Journal of Graphs and Combinatorics (2019), https://doi.org/10.1016/j.akcej.2019.02.007.

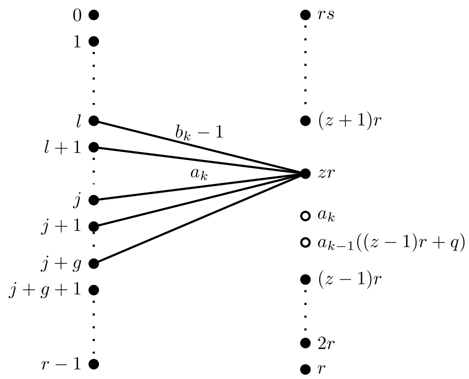

Fig. 2. The structure of the bipartite tree T under the Case 2.

By Observation 2.3, the labels a_{k-1} and a_k lie in the interval $((z-1)r, zr)$, for some fixed $z, 2 \leq z \leq s$. Therefore, $a_{k-1} = (z-1)r + q$ for some q, $1 \leq q \leq r-2$. Since b_{k-1} and b_k are consecutive missing edge labels and from Eq. (3), the labels of the sequence $D = (b_{k-1}+1, \ldots, a_{k-1}, a_{k-1}+1 = a_k, a_k+1, \ldots, b_k-1)$ are consecutive existing edges labels that lie between the label b_{k-1} and the label b_k. For whatever may be the value of k the sequence D should contain the label $a_{k-1}+1 = a_k$. Hence, by Observation 2.4, the existing edge label a_k must be obtained at the edge incident with the vertices zr and j, for some j, $1 \leq j \leq r-1$.

Claim. *The edge label $b_k - 1$ obtained at the edge (zr, l) for some l, $1 \leq l \leq j \leq r-1$*

Suppose that the edge label $b_k - 1$ is not obtained at the edge (zr, l), for any l, $1 \leq l \leq r-1$. Then by Observation 2.4, $b_k - 1$ is obtained at the edge incident at the vertex $(z+i)r$, for some i, $1 \leq i \leq s-z$. Since the labels $a_k, a_k+1, \ldots, b_k-1$ are consecutive existing edge labels and the edge label a_k is obtained at the edge (zr, j), for some j, $1 \leq j \leq r-1$, by Observation 2.4, the vertex zr must be adjacent with the vertices $j, j-1, \ldots, 0$ also the vertices $(z+1)r, (z+2)r, \ldots, (z+i-1)r$ must all be adjacent with all the vertices on the left side part of the tree T. As $zr, (z+1)r, (z+2)r, \ldots, (z+i-1)r$, for $i \geq 1$, are all adjacent with the vertices $j, j-1, \ldots, 0$, there exists a cycle in T if $i \geq 2$. Hence $i \leq 1$. Suppose $i = 1$, then the vertex $(z+1)r$ must be adjacent with the vertices $r-1, r-2, \ldots, x$, $x > j$ [if $x \leq j$, then there exists a cycle in T]. In this situation, all the children of the vertex $(z+1)r$ lie below to all the children of the vertex zr which is not possible by our construction of the tree T. Thus, $i \neq 1$. This implies that the edge label $b_k - 1$ is obtained at the edge (zr, l) for some $l, 0 \leq l \leq r-1$. As, the edge label a_k is obtained at the edge (zr, j), for some j, $1 \leq j \leq r-1$ and as $a_k \leq b_k - 1$, $l \leq j$.

When $l = 0$, the vertex zr must be adjacent with the vertices $j, j-1, \ldots, 0$. Hence the parent of the vertex zr is 0 and the children of zr must be $1, 2, \ldots, j$, for $j, j \geq 1$. By construction of the tree T, the vertices $(z+1)r, (z+2)r, \ldots, rs$ have a common parent vertex 0. As the vertices $1, 2, \ldots, j$ are the children of the vertex zr, the children of each of the vertices $(z+1)r, (z+2)r, \ldots, rs$ must lie between 0 and 1. But no such vertex possibly exists which is a contradiction. Thus, $l \neq 0$. Therefore the edge label $b_k - 1$ is obtained at the edge (zr, l) for some l, $1 \leq l \leq j \leq r-1$. Hence the claim. (See Fig. 2.)

The number of missing vertex labels from a_1 to a_{k-1} is $k-1$, which is nothing but the number of all the labels in the interval $(ir, (i+1)r)$, for each i, $1 \leq i \leq z-2$ plus the number of labels that belong to the set $\{(z-1)r+1, \ldots, (z-1)r+q = a_{k-1}\}$. Therefore, $k-1 = (z-2)(r-1)+q$. This would imply that the number of missing edge labels from b_1 to b_{k-1} is also $k-1 = (z-2)(r-1)+q$. Hence the number of existing edges that are incident at the vertices $r, 2r, \ldots, (z-1)r$ and the number of existing edges that are incident at the vertex zr having the other end vertices $l, l+1, \ldots, r-1$ is equal to the number of possible edges that can be incident at the vertices $r, 2r, \ldots, (z-1)r$ and the number of possible edges that can be incident at the vertex zr having the other end vertices $l, l+1, \ldots, r-1$ minus the number of missing edge labels from b_1 to b_{k-1}. Thus, the number of existing edges that are incident at the vertices $r, 2r, \ldots, (z-1)r$ and the number of existing edges that are incident at the vertex zr having the other end vertices $l, l+1, \ldots, r-1$ is $[(z-1)r+(r-l)] - ((z-2)(r-1)+q) = z+2r-l-q-2$. These $z+2r-l-q-2$ edges of the tree T are nothing but the edges which are incident at the vertices $r, 2r, \ldots, (z-1)r$

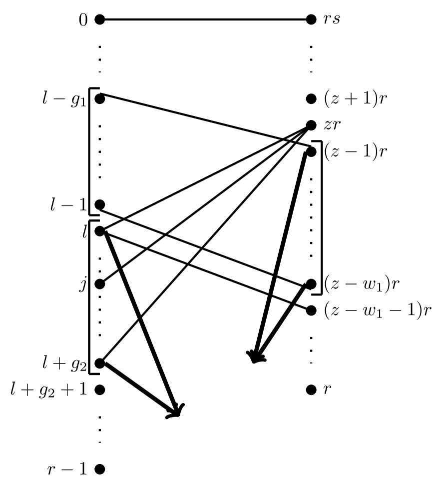

Fig. 3. The structure of the bipartite tree T under the Case I.

and edges which are incident at the vertex zr from $r-1$ to l. The graph induced by these $z+2r-j-q-2$ edges of T is a forest and it is denoted by H. Hence

$$|E(H)| = z + 2r - l - q - 2 \quad (4)$$

Now we count the number of vertices that belong to the forest H in T. To count, we consider the following two cases on the nature of the ends of the edge (zr, l). Note that, in the edge (zr, l) of the rooted tree T, either zr is a parent of l or l is a parent of zr.

Case I: zr is a parent of l

From construction of the tree T, the structure of the tree T under this situation is given in Fig. 3.

From Fig. 3, the vertices $(z-1)r, (z-2)r, \ldots, (z-w_1)r$ are the children of the vertices $l-1, l-2, \ldots, l-g_1$. Hence, the forest H is the union of the subtrees of T rooted at $zr, l-1, l-2, \ldots, l-g_1$. Thus, from Fig. 3, the total number of vertices of H, $|V(H)| = z+r-l+g_1$. This would imply that $|E(H)| = z+r-l-1$. By Eq. (4), $|E(H)| = z+2r-l-q-2$. Thus, $q = r-1$. Hence $a_k = a_{k-1}+1 = (z-1)r+q+1 = zr$. Then from Observation 2.2, the label a_k is an existing vertex label. But by Case 2, the label a_k is a missing vertex label. This is a contradiction. Hence, under this case our assumption $a_k < b_k$ is wrong.

Case II: When l is the parent of zr

Under this case, we consider the following two subcases.

Case II_a: When $j \neq l$

By our construction of the tree T the structure of the tree T under this situation is given in Fig. 4. From Fig. 4, the vertices $l+1, l+2, \ldots, j, j+1, \ldots, l+g_3$ are the children of the vertex zr, where $g_3 \geq 1$.

Then from Fig. 4, H is the subtree rooted at the vertex l. Hence, the number of vertices of H, $|V(H)| = z+r-l$. This would imply that $|E(H)| = z+r-l-1$. By Eq. (4), $|E(H)| = z+2r-l-q-2$. Thus, $q = r-1$. Hence $a_k = a_{k-1}+1 = (z-1)r+q+1 = zr$. This implies that the label a_k is an existing vertex label. But by Case II, the label a_k is a missing vertex label which is a contradiction. Hence, under this case our assumption $a_k < b_k$ is wrong.

Case II_b: When $j = l$

By our construction of the tree T the structure of the tree T under this situation is given in Fig. 5.

From Fig. 5, the vertices $l+1, l+2, \ldots, j, j+1, \ldots, l+g_4$ are the children of the vertex $zr, (z-1)r, (z-2)r, \ldots, (z-w_3)r$, where $g_4 \geq 0$. Then from Fig. 5, H is the union of the subtrees rooted at $l, l+1, \ldots, l+g_4$. The number of vertices in H, $|V(H)| = z+r-l$. This would imply that $|E(H)| = z+r-l-g_4-1$. By Eq. (4), $|E(H)| = z+2r-l-q-2$. This would imply that $q = r+g_4-1$. Hence $a_k = a_{k-1}+1 = (z-1)r+q+1 = zr+g_4 \geq zr$, which leads to a contradiction to the fact that the label a_k is a missing vertex label which lies in the interval $((z-1)r, zr)$. Hence, under this case our assumption $a_k < b_k$ is wrong.

Please cite this article as: G. Sethuraman and V. Murugan, Generating graceful unicyclic graphs from a given forest, AKCE International Journal of Graphs and Combinatorics (2019), https://doi.org/10.1016/j.akcej.2019.02.007.

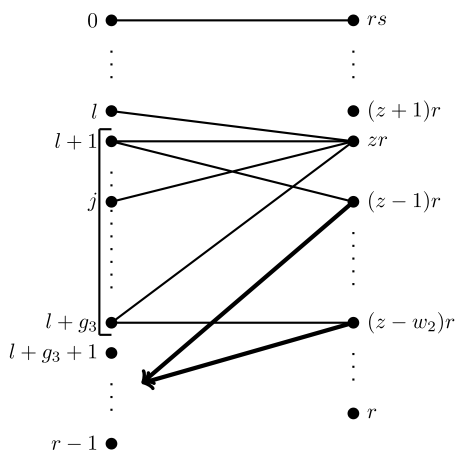

Fig. 4. The structure of the bipartite tree T under the Case II_a.

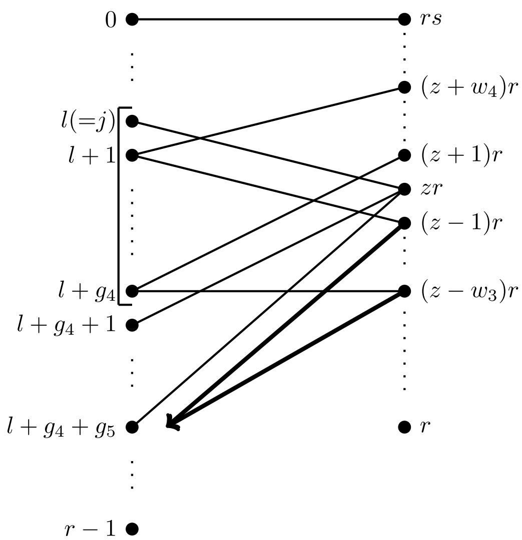

Fig. 5. The structure of the bipartite tree T under the Case II_b.

Hence from all the cases, the assumption $a_k < b_k$ is wrong. Therefore, $a_k \geq b_k$. Hence, by induction, $a_t \geq b_t$, for every t, $1 \leq t \leq m$. This means that the label $c_t = a_t - b_t$ is non-negative for every t, $1 \leq t \leq m$.

Since the current graph T^* should contain all the vertex labels 0, 1, 2, ... , $a_t - 1$. As c_t is a non-negative integer and as $c_t = (a_t - b_t) < a_t$, c_t must be a label of a vertex in that current graph T^*. □

Lemma 2.8. *The graph T^* obtained in the Embedding Algorithm is a graceful tree.*

Proof. From Lemma 2.5, it is clear that after the execution of Step 5 of the Embedding Algorithm, we obtain the tree T in which the vertices of T are labeled with distinct labels and the edges of T are also labeled with distinct labels. From the Embedding Algorithm, we observe that the graph T^* is obtained either after the complete execution of the Step 6.2 or after the complete execution of the Step 6.3.

Case 1: The graph T^* obtained after the complete execution of the Step 6.2

As the Step 6.2 is executed only when $E^c = V^c = \phi$, by Lemma 2.6, the tree T must be a star and it remains unchanged when Step 6.2 is executed. Consequently, the tree T^* should have been labeled as shown in Fig. 6. From Fig. 6, it is clear that the graph T^* is a tree with distinct vertex labels and distinct edge labels. More precisely, the

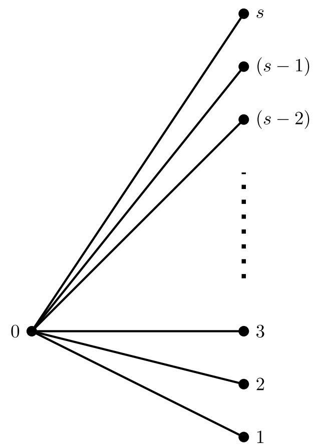

Fig. 6. The labeled tree T^* obtained at the end of Step 6.2 of the embedding algorithm.

vertex label set is $\{0, 1, 2, \ldots, s\}$ and the edge label set is $\{1, 2, \ldots, s\}$. Thus, by the definition of graceful labeling of the tree, the tree T^* is graceful.

Case 2: The graph T^* obtained after the complete execution of the Step 6.3

In Step 6.3, the tree T^* obtained after the execution of the Step 6.1 is taken as an input. The sets V^c and E^c are considered, where the elements of V^c and that of E^c are arranged in the increasing order respectively. Then for each t, $1 \le t \le d$, the label $c_t = a_t - b_t$ is found. By Lemma 2.7, the label c_t is non-negative and there always exists a vertex in the current graph T^* which has the label c_t. In the Step 6.3, for every t, $1 \le t \le d$, a new vertex labeled with a_t is taken and it is joined with the existing vertex labeled c_t of the current graph T^*.

In Step 6.3, initially the graph T^* is a tree and every execution of Step 6.3 a new vertex is added with existing vertex in T^* by a new pendant edge to the current graph T^*. Therefore the current graph T^* must be a tree. As a_t is always distinct for every t, $1 \le t \le d$, and the vertex labels of the initial tree T^* are also distinct, the updated tree $T^* \leftarrow T^* + (c_t, a_t)$ contains distinct vertex labels in every execution. Further, note that in every execution of the Step 6.3, the distinct edge label $b_t = a_t - c_t$ is obtained. As the edge labels of the initial tree T^* are also distinct, the final updated tree T^* contains distinct edge labels for all the edges. More precisely, the vertex label set of T^* are $\{0, 1, 2, \ldots, rs\}$ and the edge label set of T^* are $\{1, 2, \ldots, rs\}$. Thus, by the definition of graceful labeling of the tree, the tree T^* is graceful. □

Theorem 2.9. *The output graph obtained in the Step 7 of the Embedding Algorithm is a graceful unicyclic graph.*

Proof. From the Embedding Algorithm, we observe that the output graph G^* is obtained after the complete execution of the Step 7.1 or the output graph ${G_i}^*$ is obtained after the complete execution of the Step 7.2.

Case 1: The graph G^* obtained after the complete execution of the Step 7.1

Step 6.1 of the Embedding Algorithm is executed only when $E^c = V^c = \phi$. Then by Lemma 2.6, the tree T is a star. Thus, after the execution of Step 6.2, the labeled tree T^* will be as shown in Fig. 7.

Thus, the labeled graph G^* obtained after the execution of the Step 7.1 will be as shown in Fig. 8.

It is clear from Fig. 8, the graph G^* contains a unique cycle of length 3. Also, all the vertex labels are distinct and range over the set $\{0, 1, 2, \ldots, s+2\} \setminus \{s+1\}$ and all the edge labels are also distinct and range over the set $\{1, 2, \ldots, s+2\}$. Then it follows from the definition of graceful labeling, the unicyclic graph G^* is graceful.

Case 2: The graph G_i^* obtained after the complete execution of the Step 7.2, for i, $1 \le i \le n$

Here we consider two subcases depending on the value of n.

Case 2.1: $n = 1$

Claim 1. *The graph G_1 is unicyclic*

Please cite this article as: G. Sethuraman and V. Murugan, Generating graceful unicyclic graphs from a given forest, AKCE International Journal of Graphs and Combinatorics (2019), https://doi.org/10.1016/j.akcej.2019.02.007.

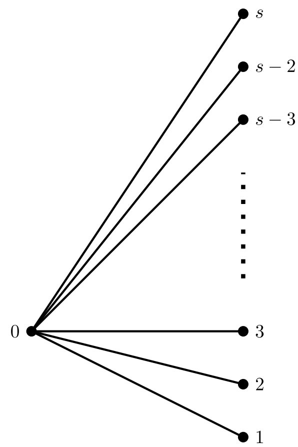

Fig. 7. The labeled tree T^* after the execution of Step 6.2.

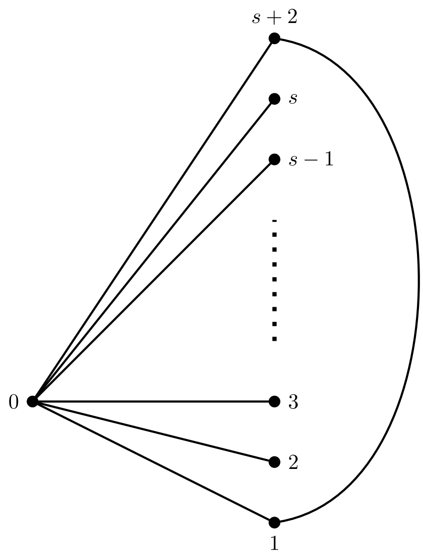

Fig. 8. The labeled graph G^* obtained after the execution of Step 7.1.

In this case the input forest F contains only one component, T_1. In Step 7.2.1, the fixed vertex u_1 of T_1 is joined with a new vertex v and this new vertex v is again joined with a chosen vertex u(which appear in the last level) of the tree T_1 that contained in T^* as its subtree. Thus, after the execution of Step 7.2.1, the vertex set of the graph G_1 is updated with a new vertex v with the label $rs + f(u) + 1$ and the edge set of G_1 is also updated with two new edges having the labels $rs + f(u) + 1$ and $rs + 1$. (See Fig. 9.)

Since T^* is a tree, the unique cycle connecting the vertex u_1 of T_1 to the vertex u of T_1 which contained in T^* followed by the edge (u, v) and the edge (v, u_1) form a unique cycle of length $l + 2$ in G_1. Thus, the graph G_1 is an unicyclic graph. By Lemma 2.8, the labels of all the vertices of the tree T^* are distinct and the labels of all the edges of the tree T^* are distinct. Therefore, after the execution of Step 7.2.1, the labels of the vertices of G_1, $0, 1, 2, \ldots, rs, rs + f(u) + 1$ are all distinct and edge labels of the edges of G_1, $1, 2, \ldots, rs, rs + 1, rs + f(u) + 1$ are all distinct.

Claim 2. *The graph G_1^* is a graceful unicyclic graph*

After the complete execution of Step 7.2.1.1, the vertex set of the graph G_1^* is updated with $f(u) - 3$ new pendant vertices which are labeled with $rs + 2, rs + 3, \ldots, rs + f(u)$ and the edge set of G_1^* is updated with $f(u) - 3$

Please cite this article as: G. Sethuraman and V. Murugan, Generating graceful unicyclic graphs from a given forest, AKCE International Journal of Graphs and Combinatorics (2019), https://doi.org/10.1016/j.akcej.2019.02.007.

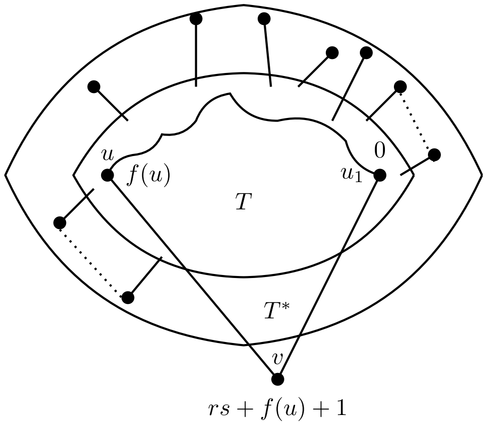

Fig. 9. The structure of the graceful unicyclic graph G_1.

new pendant edges which get the labels $rs+2, rs+3, \ldots, rs+f(u)$. Thus, after the complete execution of Step 7.2.1.1, the output graph G_1^* remains unicyclic.

After the complete execution of Step 7.2.1.1, in the output graph G_1^*, the newly added vertices have distinct labels $rs+2, rs+3, \ldots, rs+f(u)$ that are all different from the labels of all the vertices of G_1 and the newly added edges have distinct labels $rs+2, rs+3, \ldots, rs+f(u)$ that are all different from the labels of all the edges of G_1. Thus, the labels of all the vertices of G_1^* are distinct and the labels of all the edges of G_1^* are also distinct. More precisely, the set of labels of the vertices of G_1^* is $\{0, 1, 2, \ldots, rs+f(u)+1\} \setminus \{rs+1\}$ and the set of labels of the edges of G_1^* is $\{1, 2, \ldots, rs+f(u)+1\}$. Then it follows from the definition of graceful labeling, the unicyclic graph G_1^* is graceful.

Case 2.2: $n \geq 2$

Claim 3. *For every i, $2 \leq i \leq n$, the graph G_i is unicyclic*

In this case the input forest has at least two components. After the execution of Step 7.2, the fixed vertex u_i of T_i that is contained in T^* as its subtree is joined with a new vertex v and this new vertex v is again joined with a fixed vertex u_1 of T_1 that is contained in T^* as its subtree. Thus, the vertex set of the graph G_i is updated with a new vertex v with the label $rs+f(u_i)+1$ and the edge set of G_i is also updated with two new edges having the labels $rs+f(u_i)+1$ and $rs+1$. Since T^* is a tree, the unique cycle connecting the vertex u_1 of T_1 to the vertex u_i of T_i which is contained in T^* followed by the edge (u_i, v) and the edge (v, u_1) form a unique cycle of length $i+1$ in G_i. Thus, the graph G_i is unicyclic. By Lemma 2.8, the labels of all the vertices of the tree T^* are distinct and the labels of all the edges of the tree T^* are distinct. Therefore, after the execution of Step 7.2.1, the labels of the vertices of G_i, $0, 1, 2, \ldots, rs, rs+f(u_i)+1$ are all distinct and edge labels of the edges of G_i, $1, 2, \ldots, rs, rs+1, rs+f(u_i)+1$ are all distinct. (See Fig. 10.)

Claim 4. *The graph G_i^* is a unicyclic graceful graph, for every i, $2 \leq i \leq n$*

After the complete execution of Step 7.2.2.1, the vertex set of the graph G_i^* is updated with $f(u_i)-3$ new pendant vertices which are labeled with $rs+2, rs+3, \ldots, rs+f(u_i)$ and the edge set of G_i^* is updated with $f(u_i)-3$ new pendant edges which get the labels $rs+2, rs+3, \ldots, rs+f(u_i)$. After the complete execution of the Step 7.2.2.1, the output graph G_i^* remains as unicyclic.

After the complete execution of Step 7.2.2.1, in the output graph G_i^*, the newly added vertices have distinct labels $rs+2, rs+3, \ldots, rs+f(u_i)$ that are all different from the labels of all the vertices of G_i and the newly added edges have distinct labels $rs+2, rs+3, \ldots, rs+f(u_i)$ that are all different from the labels of all the edges of G_i. Thus, the labels of all the vertices of G_i^* are distinct and the labels of all the edges of G_i^* are also distinct. More precisely, the set of labels of the vertices of G_i^* is $\{0, 1, 2, \ldots, rs+f(u_i)+1\} \setminus \{rs+1\}$ and the set of labels of the edges of G_i^* is $\{1, 2, \ldots, rs+f(u_i)+1\}$. Then it follows from the definition of graceful labeling, the unicyclic graph G_i^* is graceful. □

Please cite this article as: G. Sethuraman and V. Murugan, Generating graceful unicyclic graphs from a given forest, AKCE International Journal of Graphs and Combinatorics (2019), https://doi.org/10.1016/j.akcej.2019.02.007.

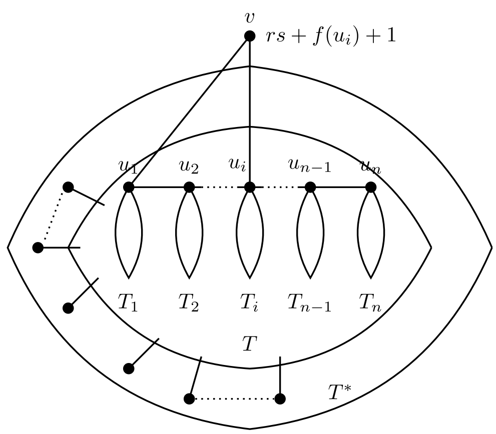

Fig. 10. The structure of the graceful unicyclic graph G_i.

3. Discussion

From Theorem 2.9, we observe that the Embedding Algorithm generates distinct unicyclic graceful graphs and each unicyclic graph contains the given forest F having n arbitrary trees T_i, for $1 \leq i \leq n$ as its subgraph. Also the Embedding Algorithm generates a unicyclic graceful graph with a fixed cycle length n if the input forest contains $n - 1$ components. All these unicyclic graphs obtained from Embedding Algorithm supports the Truszczynski's conjecture [6], that all unicyclic graphs except cycle C_{4n+1} and C_{4n+2} are graceful.

In this paper, we have embedded a given forest in a graceful tree as well as a graceful unicyclic graph. In this direction of graceful graph embedding, it would be interesting to explore the following questions.

- Is it possible to embed, a given unicyclic graph in a graceful unicyclic graph?
- What would be the minimum number of additional edges required for the embedding of a unicyclic graph into a graceful unicyclic graph?

Acknowledgments

The second author gratefully acknowledges Centre for Research, Anna University, Chennai, India for the financial support through the Anna Centenary Research Fellowship under the grant Ref: CR/ACRF/2014/40, Dated: 28.12.2013.

References

[1] D.B. West, Introduction to Graph Theory, second ed., Prentice Hall of India, 2001.
[2] G. Ringel, Problem 25, in: Theory of Graphs and its Applications, Proc. Symposium Smolenice, Prague, 1963, p. 162.
[3] A. Kotzig, Decompositions of a complete graph into 4k-gons, Matematicky Casopis 15 (1965) 229–233 (in Russian).
[4] A. Rosa, On certain valuations of the vertices of a graph, theory of graphs, in: International Symposium, Rome, 1966, Gordon and Breach, N.Y. and Dunod Paris, 1967, pp. 349–355.
[5] S.W. Golomb, How to number a graph, in: R.C. Read (Ed.), Graph Theory and Computing, Academic Press, New York, 1972, pp. 23–37.
[6] M. Truszczynski, Graceful unicyclic graphs, Demonstatio Math. 17 (1984) 377–387.
[7] C. Barrientos, Graceful graphs with pendant edges, Australas. J. Combin. 33 (2005) 99–107.
[8] Jay Bagga, S. Arumurugam, New classes of graceful unicyclic graphs, Electron. Notes Discrete Math. 48 (2015) 27–32.
[9] R. Figueroa-Centeno, R. Ichishima, F. Muntaner-Batle, A. Oshima, Gracefully cultivating trees on a cycle, Electron. Notes Discrete Math. 48 (2015) 143–150.
[10] J.A. Gallian, A dynamic survey of graph labeling, Electron. J. Combin. 20 (2017) DS6.
[11] B.D. Acharya, Construction of certain infinite families of graceful graphs from a given graceful graph, Def. Sci. J. 32 (1982) 231–236.
[12] G. Sethuraman, A. Elumalai, Packing of any set of graphs into a graceful/harmonious/elegant graph, Ars Combin. 76 (2005) 297–301.
[13] G. Sethuraman, P. Ragukumar, Peter J. Slater, Embedding an arbitrary tree in a graceful tree, Bull. Malays. Math. Sci. Soc. 39 (2016) 341–360.

Please cite this article as: G. Sethuraman and V. Murugan, Generating graceful unicyclic graphs from a given forest, AKCE International Journal of Graphs and Combinatorics (2019), https://doi.org/10.1016/j.akcej.2019.02.007.