The Gracefulness of the Merging Graph $N \bowtie C_4$ with Dotnet Framework

Solairaju¹, N. Abdul Ali² and R.M. Karthikkeyan³

³: M.Phil Scholar, Jamal Mohamed College, Trichy – 20.

Abstract:
There are many graceful graphs from standards path, circuit, wheel etc. In this paper a new class of graceful graphs related to $c_4$ [circuits with 4 vertices] is obtained.

Keyword: path limit, $n$ - copies of $c_4$

I. Introduction:
Most graph labeling methods trace their origin to one introduced by Rosa [2] or one given Graham and Sloane [1]. Rosa defined a function $f$, a $\beta$-valuation of a graph with $q$ edges if $f$ is an injective map from the vertices of $G$ to the set \{0, 1, 2, ..., $q$\} such that when each edge $xy$ is assigned the label $|f(x)-f(y)|$, the resulting edge labels are distinct.


A. Solairaju and others [5,6,7,8,9] proved the results that(1) the Gracefulness of a spanning tree of the graph of Cartesian product of $P_n$ and $C_4$, was obtained (2) the Gracefulness of a spanning tree of the graph of cartesian product of $S_m$ and $S_n$, was obtained (3) edge-odd Gracefulness of a spanning tree of Cartesian product of $P_2$ and $C_4$, was obtained (4) Even - edge Gracefulness of the Graphs was obtained (5) ladder $P_2 \times P_n$ is even-edge graceful, and (6) the even-edge gracefulness of $P_n \circ nC_3$ is obtained. (8) Gracefulness of $T_p$-tree with five levels obtained by java programming. (9) Gracefulness of $n^*c_4$ Merging with paths. (10) A new class of graceful trees and (11) Gracefulness of $P_k \circ 2c_4$, is obtained. (12, 13, 14) Used for dot net framework 3.5.

Section I: Preliminaries

Definition 1.1:
Let $G = (V,E)$ be a simple graph with $p$ vertices and $q$ edges.
A map $f: V(G) \rightarrow \{0,1,2,\ldots,q\}$ is called a graceful labeling if
(i) $f$ is one-to-one
(ii) The edges receive all the labels (numbers) from 1 to $q$ where the label of an edge is the absolute value of the difference between the vertex labels at its ends.
A graph having a graceful labeling is called a graceful graph.

Example 1.1: The graph $6 \Delta P_3$ is a graceful graph.

Theorem: The gracefulness of the merging graph $n \bowtie c_4$ generalization:
Example 1.1:

Algorithm for THE GRACEFULNESS OF THE MERGING GRAPH n ** C4 in Dotnet Language

```csharp
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
```

\[
f(T_1) = 0,
f(T_2) = q,
f(T_3) = 1,
f(T_4) = q - 2 \]
\[
f(v_i) = q - 4
f(v_n) = 3.
\]
\[
f(v_i) = \begin{cases} f(v_i - 1) - 4, & \text{for } i = 2, 3, \ldots, (n - 1), \\ f(v_i - 1) + 4, & \text{for } i = n + 1, n + 2, \ldots, 2(n - 1) \end{cases}
\]
namespace Class
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        int StartX = 5;
        int StartY = 5;
        int Width = 500;
        int Height = 500;

        private void btnLine_Click(object sender, EventArgs e)
        {
            // Declare edges and center point
            int edges = Convert.ToInt32(textBox1.Text);
            int center = Width / 2;
            int newedge = (1 + 4 * edges);
            label2.Text = "No of Edges :" + newedge.ToString();

            // Set Graphics Tool
            Graphics g;
            g = this.CreateGraphics();
            g.Clear(Color.White);
            SolidBrush myBrush = new SolidBrush(Color.Black);
            Font font = new Font("Times New Roman", 12.0f);
            Pen myPen = new Pen(Color.Red);
            myPen.Width = 2;

            // Draw center line and Rectangle
            g.DrawLine(myPen, Width, StartX, StartY, Height);
            g.DrawRectangle(myPen, StartX, StartY, Width, Height);

            // draw Inner Rectangle lines
            int nval = (center / edges);
            int sample = center;
            int sample1 = center;
            for (int i = 0; i < edges - 1; i++)
            {
                sample = sample + nval;
                sample1 = sample1 - nval;

                g.DrawLine(myPen, sample, sample, StartY, Height);
                g.DrawLine(myPen, Width, StartX, sample, sample);
                g.DrawLine(myPen, sample, sample1, StartY, Height);
                g.DrawLine(myPen, Width, StartX, sample1, sample1);
            }

            // Find Edges Values
            int q = (1 + 4 * Convert.ToInt32(textBox1.Text));
            int T1 = 0;
            int T2 = q;
            int T3 = 1;
            int T4 = q - 2;
            int P1 = 1;
        }
    }
}
Example 1: n is Even (n=6):

```csharp
int v1 = q - 4;
int vn = 3;
int incrementvalue = center / edges;
int addsub = 0;
int diff = 300;
for (int j = 0; j < edges - 1; j++)
{
    addsub = addsub + incrementvalue;
    g.DrawString(vn.ToString(), font, myBrush, center+addsub, center+addsub);
    g.DrawString(Convert.ToString(vn - T3), font, myBrush, center+(addsub-100), center+addsub);
    g.DrawString(Convert.ToString(vn - T1), font, myBrush, center+addsub, center+(addsub-100));
    g.DrawString(v1.ToString(), font, myBrush, StartX + addsub, StartY + addsub);
    g.DrawString(Convert.ToString(v1 - T3), font, myBrush, StartY +(addsub-60),StartY+addsub+10);
    g.DrawString(Convert.ToString(v1 - T1), font, myBrush, StartX+addsub+10, StartX+(addsub-60));
    v1 = v1 - 4;
    vn = vn + 4;
}
```

Example 1: n is Even (n=6)
Example 2: n is Odd (n=5)

References:
[5] A. Solairaju and P. Sarangapani, even-edge gracefulfulness of $P_n \circ nC_3$, Preprint (Accepted for publication in Serials Publishers, New Delhi).