Iteractive algorithm for computing irregularity strength of complete graph

Article *in* Ars Combinatoria · July 2018

Iterative algorithm for computing irregularity strength of complete graph

M. A. Asim[1,2], A. Ahmad[1], R. Hasni[2],

[1]College of Computer Science & Information Systems,
Jazan University, Jazan, KSA.
mr.ahsan.asim@gmail.com, ahmadsms@gmail.com,

[2]School of Informatics and Applied Mathematics,
University Malaysia Terengganu,
21030 Kuala Terengganu, Terengganu, Malaysia.
hroslan@umt.edu.my

Abstract

Algorithms help in solving many problems, where other mathematical solutions are very complex or impossible. In this paper edge irregularity strength of a complete graph $es(K_n)$ is computed using the algorithm that is impossible to compute manually on bigger graphs. Using the values of $es(K_n)$ an upper-bound is suggested that is far better than previous upper bound F_n.

Keywords *: irregular assignment, irregularity strength, decrease and conquer, graph algorithms, computational complexity*

1 Introduction

In computer science graphs are used in variety of applications, directly or indirectly. Especially quantitative labeled graphs have played a vital role in computational linguistics, decision making software tools, coding theory and path determination in networks. In fifth-generation-computers, graphs are used to represent the interconnection network of the parallel processors. Simplest approach to represent this interconnection is a complete graph K_n with n vertices, by considering vertices as n processors and undirected edges as two-way link between each processor [21].

To get full advantage from graph theory, plenty of algorithms are designed and are used in computer applications. Various design strategies are used to design efficient algorithms whereas optimum solution can be achieved by applying appropriate design strategy according to nature of the problem. Computational complexity of designed algorithm gives the idea whether solution is acceptable according to available resources or not. In this paper we considered a simple undirected complete graph $G = (V, E)$ and provided its irregularity strength according to below mathematical definitions.

Let $G = (V, E)$ be the connected, simple and undirected graph with vertex set V and edge set E. By *labeling* we mean any mapping that carries a set of graph elements to a set of numbers (usually positive integers), called *labels*. If the domain is vertex-set or edge-set, the labelings are called respectively *vertex labelings* or *edge labelings*. If the domain is $V \cup E$ then it is called *total labeling*. Thus, for an edge k-labeling $\phi : E(G) \to \{1, 2, \ldots, k\}$ the associated weight of a vertex $x \in V(G)$ is

$$w_\phi(x) = \sum \phi(xy),$$

where the sum is over all vertices y adjacent to x, and for a total k-labeling $\varphi : V(G) \cup E(G) \to \{1, 2, \ldots, k\}$ the associated weight of edge $xy \in E(G)$ is

$$wt_\varphi(xy) = \varphi(x) + \varphi(xy) + \varphi(y).$$

Chartrand *et al.* in [9] introduced edge k-labeling ϕ of a graph G such that $w_\phi(x) \neq w_\phi(y)$ for all vertices $x, y \in V(G)$ with $x \neq y$. Such labelings are called *irregular assignments* and the *irregularity strength* $es(G)$ of a graph G is known as the minimum k for which G has an irregular assignment using labels at most k. This parameter has attracted much attention in [5, 6, 8, 10, 11, 15, 16] articles.

Motivated by these papers, Bača *et al.* in [7] started to investigate two modifications of the irregularity strength of graphs, namely a *total edge irregularity strength*, denoted by $tes(G)$, and a *total vertex irregularity strength*, denoted by $tvs(G)$. Some results on total edge irregularity strength and total vertex irregularity strength can be found in these [2, 3, 4, 13, 14, 17, 19, 20] articles.

Combining both previous modifications of the irregularity strength, Marzuki, Salman and Miller [18] introduced a new irregular total k-labeling of a graph G called *totally irregular total* k-labeling, which is required to be at the same time vertex irregular total and also edge irregular total. They have given an upper bond and a lower bond of the totally irregular total k-labeling, denoted by $ts(G)$. The most complete recent survey of graph labelings is [12].

A vertex k-labeling $\phi : V \to \{1, 2, \ldots, k\}$ is defined to be an *edge irregular k-labeling* of the graph G if for every two different edges e and f there is $w_\phi(e) \neq w_\phi(f)$, where the weight of an edge $e = xy \in E(G)$ is $w_\phi(xy) = \phi(x) + \phi(y)$. The minimum k for which the graph G has an edge irregular k-labeling is called the *edge irregularity strength* of G, denoted by $es(G)$.

The following theorems given in [1], established bounds for the edge irregularity strength of a graph G.

Theorem 1 *[1] Let $G = (V, E)$ be a simple graph with maximum degree $\Delta = \Delta(G)$. Then*

$$es(G) \geq \max\left\{\left\lceil \frac{|E(G)| + 1}{2} \right\rceil, \Delta(G)\right\}.$$

Theorem 2 *[1] Let G be a graph of order p. Let the sequence F_n of Fibonacci numbers be defined by the recurrence relation $F_n = F_{n-1} + F_{n-2}$, $n \geq 3$, with seed values $F_1 = 1$ and $F_2 = 2$. Then $es(G) \leq F_p$.*

In the next section, we provide the better upper bound for complete graph by using algorithm way.

2 Main result

In this section original algorithm is given that is designed using decrease and conquer approach to compute the value of k as output and by providing n the order of complete graph as input.

Input: A positive integer $n > 3$ that will be considered as the number of vertices.
Output: Label of Vertices $\phi : V[n] \rightarrow \{1, 2, 3, \ldots, k\}$ according to definition of K_n used in this problem.

Algorithm 1 KG-Labeling(n)

$V[n] \leftarrow \{1, 2, 3\}$
for each edge $w_\phi(x, y) \leftarrow 1$ where $x \neq y$
$t \leftarrow 4$
$m \leftarrow 3$
repeat
    $V[t] \leftarrow m + 1$
    EdgeCalculate(G, t)
    if (EdgeDuplicate(G, t) $\neq$ TRUE)
        $t \leftarrow t + 1$
until $t \leftarrow n$
return V

Description: $V[n]$ is a linear Array that holds the labels of vertices as outcome of this algorithm, $\{1, 2, 3\}$ are assigned as seed values to initialize the algorithm. The computed value at n^{th} location will be k i.e irregularity strength of K_n. In order to compute $\phi : V[n]$ decrease-and-conquer algorithmic design strategy is applied that compute labels of $K_{n-1}, K_{n-2}, \ldots K_4$ that are used as $V[t]$ in algorithm, where t means temporary value of n. Similarly $E[n][n]$ is an adjacency matrix that holds the weights of edges.

Computational Complexity: Cost of KG-Labeling, depends on the cost of EdgeCalculate and EdgeDuplicate algorithms, but mainly it depends how many times "repeat-until" loop will execute to compute the $K_{n-1}, K_{n-2}, \ldots K_4$ values as a phenomenon of back-tracking.

Algorithm 2 EdgeCalculate(G, t)

```
for i ← 1 to n − t
    for j ← i + 1 to t
        E[i][j] ← V[i] + V[j]
```

Description: Algorithm of $EdgeCalculate(G,t)$ is based on two nested loops, that are executed in a fashion of Gaussian arithmetic series, to calculate the weight of edge by adding weights of corresponding vertices.

Computational Complexity: For a complete graph of t vertices, $EdgeCalculate(G,t)$ will execute $\frac{t(t-1)}{2}$ times, that is equal to total number of edges.Due to symmetry of complete graph, cost of algorithm will be same on all asymptotic ranges $[O\,|\Theta\,|\Omega\,]T(t) = E$.

Algorithm 3 EdgeDuplicate(G, t)

```
for i ← 2 to t − 2
    for j ← i + 1 to t − 1
        for l ← 1 to i − 1
            for w ← j + 1 to t
                if E[i][j] = E[l][w]
                    return TRUE
                    break
return FALSE
```

Description: $EdgeDuplicate(G,t)$ algorithm is applied to verify the condition $w_\phi(u,v) \neq w_\phi(u',v')$, for any edges $e = (u,v)$ and $f = (u',v')$.

Computational Complexity: Cost of $EdgeDuplicate(G,t)$ is based on four nested loops, if brute-force strategy would be applied to design the algorithm, the worst case cost of algorithm would be $\frac{(\frac{t^2-t}{2})^2-\frac{t^2-t}{2}}{2}$, that means $O(t^4)$. But taking the advantage of decrease-and-conquer design strategy, number of comparisons are reduced significantly. As a result, cost of the algorithm fall in a range of $t \leq O(t) \leq t^2$.
For illustration, there are some examples of complete graphs K_n given in Table 2, one can see that as we increase the values the order of the complete graph, computing the edge irregularity become complex.

3 Conclusion

Importance of algorithmic solutions is obvious from Table-1 as it shows the exact values of $es(K_n)$, computed by algorithm. Given algorithm is valid for any order of graph as long as resources of computer supports. Value of upper bound obtain

V	E :Number of edges	$es(K_n)$	$3 * E\log(V)$	F_n
1	0	1	0	1
10	45	53	135	55
20	190	413	741.64	6765
30	435	1161	1927.64	832040
40	780	2497	3748.82	102334155
50	1225	4447	6243.71	12586269025
60	1770	6980	9441.98	1548008755920
70	2415	11110	13367.73	190392490709135
80	3160	15470	18041.29	23416728348467685
90	4005	21492	23480.22	2880067194370816120
100	4950	27602	29700	354224848179261915075

Table 1: Output of algorithm and comparison between its two upper bounds

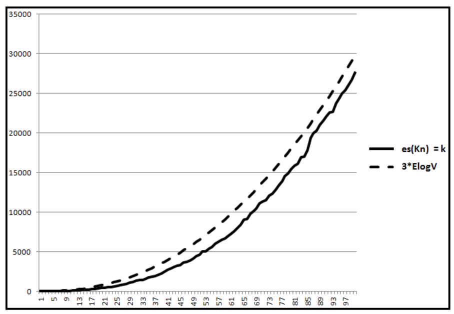

Figure 1: Shows relation between $es(K_n)$ and $3 * E\log(V)$

in this article is compared with previously identified upper bound to show the difference in mathematical proofs and accuracy of algorithmic results.

4 Acknowledgement

This research work is supported by Jazan University, KSA.

References

[1] A. Ahmad, O. Al-Mushayt and Martin Bača , *On edge irregularity strength of graphs*, *Applied Mathematics and Computation*, **243**(2014) 607–610.

[2] A. Ahmad, M. Bača, Y. Bashir and M.K. Siddiqui, Total edge irregularity strength of strong product of two paths, *Ars Combin.* **106** (2012), 449-459.

[3] A. Ahmad, Martin Bača, M.K. Siddiqui , On edge irregular total labeling of categorical product of two cycles, *Theory of Comp. Systems,* **54**(2014), 1–12.

[4] A. Ahmad, E.T. Baskoro and M. Imran, Total vertex irregularity strength of disjoint union of Helm graphs, *Discussiones Mathematicae Graph Theory,* **32(3)** (2012), 427-434.

[5] M. Aigner and E. Triesch, Irregular assignments of trees and forests, *SIAM J. Discrete Math.* **3** (1990), 439–449.

[6] D. Amar and O. Togni, Irregularity strength of trees, *Discrete Math.* **190** (1998), 15–38.

[7] M. Bača, S. Jendroľ, M. Miller and J. Ryan, On irregular total labellings, *Discrete Math.* **307** (2007), 1378–1388.

[8] T. Bohman and D. Kravitz, On the irregularity strength of trees, *J. Graph Theory* **45** (2004), 241–254.

[9] G. Chartrand, M.S. Jacobson, J. Lehel, O.R. Oellermann, S. Ruiz and F. Saba, Irregular networks, *Congr. Numer.* **64** (1988), 187–192.

[10] R.J. Faudree, M.S. Jacobson, J. Lehel, R.H. Schelp, Irregular networks, regular graphs and integer matrices with distinct row and column sums, *Discrete Math.* **76** (1989), 223?-240.

[11] A. Frieze, R.J. Gould, M. Karonski, and F. Pfender, On graph irregularity strength, *J. Graph Theory* **41** (2002), 120–137.

[12] J.A. Gallian, A Dynamic Survey Graph labeling, *Electron. J. Combin.* **18** (2015), # DS6, 1–389.

[13] J. Ivančo and S. Jendroľ, Total edge irregularity strength of trees, *Discussiones Math. Graph Theory* **26** (2006), 449–456.

[14] S. Jendroľ, J. Miškuf and R. Soták, Total edge irregularity strength of complete graphs and complete bipartite graphs, *Discrete Math.* **310** (2010), 400–407.

[15] M. Kalkowski, M. Karonski, F. Pfender, A new upper bound for the irregularity strength of graphs, *SIAM J. Discrete Math.*, **25(3)**, (2011) 1319-?1321.

[16] P. Majerski, J. Przybylo, On the irregularity strength of dense graphs, *SIAM J. Discrete Math.*, **28(1)**, (2014), 197?-205.

[17] K.M. Mominul Haque, Irregular total labellings of generalized Petersen graphs, *Theory Comput. Syst.* **50** (2012), 537?-544.

[18] C.C. Marzuki, A.N.M. Salman and M. Miller, On the total irregularity strength on cycles and paths, *Far East Journal Of Mathematical Sciences*, **82(1)**, (2013), 1–21.

[19] Nurdin, E.T. Baskoro, A.N.M. Salman and N.N. Gaos, On the total vertex irregularity strength of trees, *Discrete Math.* **310** (2010), 3043–3048.

[20] J. Przybylo, Linear bound on the irregularity strength and the total vertex irregularity strength of graphs, *SIAM J. Discrete Math.* **23** (2009), 511–516.

[21] K. H. Rosen, Discrete Mathematics and Its Applications, edition 7^th, *McGraw-Hill Companies, Inc.*, 2012.

K_n	Labels of vertices	Weights of edges
4	1 2 3 5	3 4 5 6 7 8
5	1 2 3 5 8	3 4 5 6 7 8 9 10 11
6	1 2 3 5 8 13	3 4 5 6 7 8 9 10 11 13 14 15 16 18 21
7	1 2 3 5 8 13 21	3 4 6 9 14 22 5 7 10 15 23 8 11 16 24 13 18 26 21 29 34
8	1 2 3 5 8 13 21 30	3 4 6 9 14 22 31 5 7 10 15 23 32 8 11 16 24 33 13 18 26 35 21 29 38 34 43 51
9	1 2 3 5 8 13 21 30 39	3 4 6 9 14 22 31 40 5 7 10 15 23 32 41 8 11 16 24 33 42 13 18 26 35 44 21 29 38 47 34 43 52 51 60 69
10	1 2 3 5 8 13 21 30 39 53	3 4 6 9 14 22 31 40 54 5 7 10 15 23 32 41 55 8 11 16 24 33 42 56 13 18 26 35 44 58 21 29 38 47 61 34 43 52 66 51 60 74 69 83 92
15	1 2 3 5 8 13 21 30 39 53 74 95 128 152 182	3 4 6 9 14 22 31 40 54 75 96 129 153 183 5 7 10 15 23 32 41 55 76 97 130 154 184 8 11 16 24 33 42 56 77 98 131 155 185 13 18 26 35 44 58 79 100 133 157 187 21 29 38 47 61 82 103 136 160 190 34 43 52 66 87 108 141 165 195 51 60 74 95 116 149 173 203 69 83 104 125 158 182 212 92 113 134 167 191 221 127 148 181 205 235 169 202 226 256 223 247 277
20	1 2 3 5 8 13 21 30 39 53 74 95 128 152 182 212 258 316 374 413	3 4 6 9 14 22 31 40 54 75 96 129 153 183 213 259 317 375 414 5 7 10 15 23 32 41 55 76 97 130 154 184 214 60 318 376 415 8 11 16 24 33 42 56 77 98 131 155 185 215 261 319 377 416 13 18 26 35 44 58 79 100 133 157 187 217 263 321 379 418 21 29 38 47 61 82 103 136 160 190 220 266 324 382 421 34 43 52 66 87 108 141 165 195 225 271 29 387 426 51 60 74 95 116 149 173 203 233 279 337 395 434 69 83 104 125 158 182 212 242 288 346 404 443 92 113 134 167 191 221 251 297 355 413 452 127 148 181 205 235 265 311 369 427 466 169 202 226 256 286 332 390 448 487 223 247 277 307 353 411 469 508 280 310 340 386 444 502 541 334 364 410 468 526 565 394 440 498 556 595 470 528 586 625 574 632 671 690 729 787

Table 2: edge irregular labeling for some complete graphs K_n