On Lucky Labeling of Special Classes of Graphs

S. Akbari [a,c,*], S. Assadi [b], E. Emamjomeh-Zadeh [b], F. Khani [b]

Tehran, Iran

[a] *Department of Mathematical Sciences, Sharif University of Technology,*

[b] *Department of Computer Engineering, Sharif University of Technology,*

[b] *School of Mathematics, Institute for Research in Fundamental Sciences(IPM-P.O.Box 19395-5746),*

Tehran, Iran [†]

Abstract. Suppose the vertices of a graph G were labeled arbitrarily by positive integers, and let $S(v)$ denote the sum of labels over all neighbors of vertex v. A labeling is lucky if the function S is a proper coloring of G, that is, if we have $S(u) \neq S(v)$ whenever u and v are adjacent. The least integer k for which a graph G has a lucky labeling from the set $\{1, 2, ..., k\}$ is the lucky number of G, denoted by $\eta(G)$.

Keywords: Vertex-coloring, Lucky Labeling, Lucky Number .

I. Introduction

Suppose the vertices of a graph G were labeled arbitrarily by positive integers, and let $S(v)$ denote the sum of labels over all neighbors of vertex v. A labeling is lucky if the function S is a proper coloring of G, that is, if we have $S(u) \neq S(v)$ whenever u and v are adjacent. The least integer k for which a graph G has a lucky labeling from the set $\{1, 2, ..., k\}$ is the lucky number of G, denoted by $\eta(G)$.

Let $A \subseteq \mathbb{R}$ be a set. Then we say that G has an *A-lucky labeling* if there exists a lucky labeling using numbers from the set A.

In the first part of the paper we prove that if $G \neq C_n$ (n is odd) is a unicyclic graph, then $\eta(G) = 2$. Moreover, if n is odd, the $\eta(C_n) = 3$. A *unicyclic* graph is a connected graph containing exactly one cycle. It was proved that for every tree T, $\eta(T) \leq 2$[zelanzy]. Also for graphs which

*Corresponding author. S. Akbari

† *E-mail addresses*: s_akbari@sharif.edu (S. Akbari), s_asadi@ce.sharif.edu (S. Assadi), emamjomeh@ce.sharif.edu (E. Emamjomeh-Zadeh), fkhani@ce.sharif.edu (F. Khani).

are only one cycles these two theorem can be proven: Even cycles, cycles consisted of even number of nodes, have lucky number of 2 and odd cycles 3. In this paper we want to find the lucky number of unicyclic graphs.

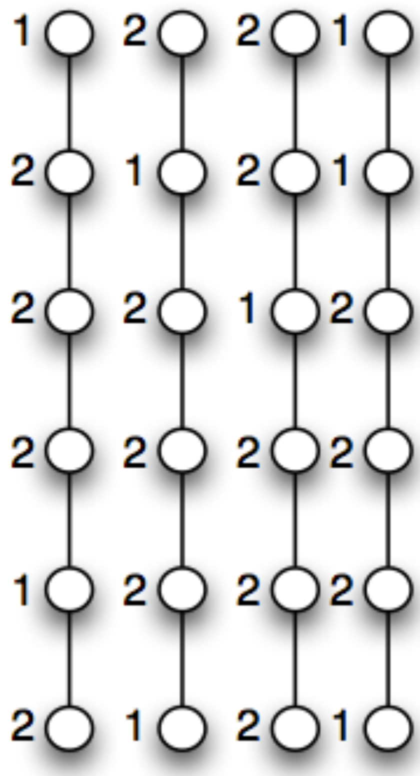
Remark 1. Even cycles have {a,b}-lucky labeling for any $a, b \in \mathbb{R}$. To see that, one can give a and b alternatively to vertices of graph.

Theorem 1. *Odd cycles have $\{a, b, c\}$-lucky labeling, for every $a, b, c \in \mathbb{R}$.*

Proof. First we prove that odd cycles cannot be labeled with any pair of real numbers. To see this we define *maximal strings* in a cycle. A maximal string in a vertex labeling of C_n is a maximal path whose all vertices have the same label. The proof is based on these two facts. First, the number of maximal strings in an odd cycle is even. It is obvious that number of these blocks cannot be one. If their number is odd, it means that there are two neighbor maximal string with the same label, a contradiction. Second, we show that the size of maximal string is 1 or 3. If the size of a string is more than 3 then there are two neighbors with the same label and thus their sums would be equal. Also block size cannot be 2, because otherwise there are two neighbor vertices with the same label, one equal in string and one unequal label outside of string and so their sum would be equal. So the claim is proved. So according to above facts □

To find the lucky number of unicyclic graphs we need a way for labeling the cycles and the trees. At first we introduce a method for labeling the trees. We should note that the trees can be seen as a set of paths which are joined in their first node to make the tree. So if a method for labeling the paths and merging them can be constructed, the trees can be labeled with this approach.

Lemma 1. *Following paths can be labeled with numbers 1,2 to any depth without having problem with the roots sum.*
Pattern 1: 1 - 2 - 2 - 2 - 1 - 2 - ... $sum(root) \geq 4$
Pattern 2: 2 - 1 - 2 - .... $sum(root) = 3$ *or* $sum(root) \geq 5$
Pattern 3: 2 - 2 - 1 - 2 - ... $sum(root) \geq 4$
Pattern 4: 1 - 1 - ... $sum(root) \geq 4$



Patterns

Proof. Pattern 1: In common, sum of two neighbors are 3 and 4 and when we cut it sum of the last node is 2 or 1 and so there will be no problem.
Pattern 2: By adding the first 1 this pattern is reduced two the Pattern 1 with sum(root) = 4 and so it can be solved that way.
Pattern 3: By adding the first 2 there would be no problem, and after that this pattern is reduced to Pattern 2 with sum(root) = 3 and so on.
Pattern 4: We should solve this pattern in two case. first when depth of path is less than five. In this case the numbering can be like this : 1 - 1 - 1 - 2.
it can be seen that this case doesn't have any problem according to sum of the nodes. The other case is when the depth is more than 4. It can be labeled like this : 1 - 1 - 2 - 1 - 2 - ... , which is Pattern 1 from the third node. □

Now we can use this lemma for labeling the unicyclic graphs. To achieve this goal, we are using strong induction on vertices of a unicyclic graph.

Theorem 2. *In unicyclic graph G, If it is only a mere odd cycle its lucky number is 3 otherwise it is 2.*

Proof. For this proof we need to prove the induction base and induction step correctness. In order to have a better view of induction base, first we prove the step.

Induction Step:

Find the vertex v with maximum depth in G which is not in the cycle and also have more than one child. If such a vertex cannot be found we have reached the induction base. Otherwise, this vertex has two or more outgoing path.

We can see there would be two restriction for sum of v, first its sum must be different with its parent sum, second its sum must not be 3 if its own label is 1 and 4 if it is 2. The second restriction comes from the patterns characteristic of Lemma 3. If v meets this restriction, the paths beginning from this vertex can be labeled with Lemma 3. So it only remains to show that these two restrictions can always be meet. To show this we can see that v have at least two children and they can be numbered at least three ways : 1-1,1-2,2-2. It follows that there is a numbering in which both of the restrictions can be satisfied.

Induction Base:

The induction base is a cycle with some path along it. We solve the base in two cases when the cycle is an even cycle and when it is odd one. We begin with simple part, even cycle.

Even Cycles:

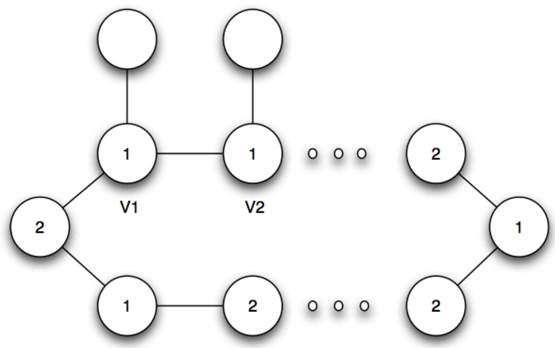
We claim that with this algorithm an even cycle in the graph can be labeled.

1. Label every other vertex by 1 and fill the remaining with 2.
2. Label 2 all the child of vertex with label 1.
3. Label one of the children of vertex with label 2, 1, and let the others be 2.

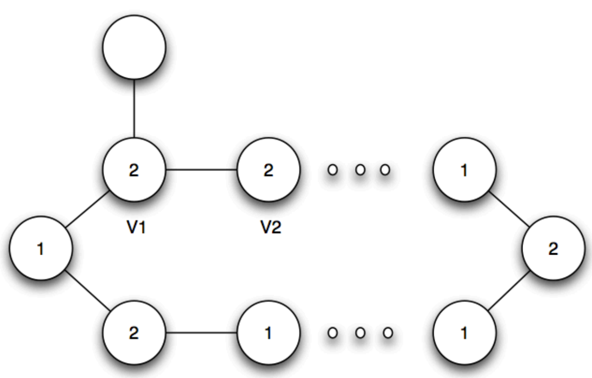
Now we prove our claim. Sum of vertex with 1 is at least 4 and it is also even. So the paths which are beginning with label 1 can be labeled with pattern 1 and 4. Sum of vertex with 2 is also 2 or is odd so the neighbors in cycle doesn't have any problem with each other. Also when they are in a path, their sum is 3 or 5 or more and so the these paths can be labeled with pattern 2 and 3. So the induction base is true for the even cycles.

Odd Cycles:

We have proved before that if a graph G is only an odd cycle its lucky label is 3. So we have to check the odd cycles in a situation when there is at least one vertex in cycle which is first node of a path. We prove this part in two cases:



Case 1



Case 2

Case 1: there is at least two adjacent vertex in cycle with another neighbor outside of cycle. To label this case, we name these two vertices v_1 and v_2. First label them both with 1 and then label 2 all the vertices in cycle which have an neighbor with label 1 and label 1 all vertices in cycle which

have an neighbor with label 2 and so on.
You should notice that there is a special case where none of the above patterns can be used. It is the case that we have a vertex with label 2, with two labeled 1 neighbors in cycle which their sums are 5 and 6 and the vertex only have two children. In order for the vertex not to have the same sum with its neighbor both of it's child should be 1, So its sum become 4 and we don't have any pattern for labeling a path which begins with 2 and its root sum is 4. So We are going to avoid this special case. To do so, We are not letting any vertex in cycle with label 1 to have sum 6. We are suggesting this algorithm for labeling this case:
First we label all the children of v_1. It's sum is bigger than its neighbors. So the only restriction is that its sum shouldn't be 6. And we at least have two numbering for this vertex so this restriction can be handled.
Then we are going to label children of v_2. For v_2 there is two restriction. It's sum should not be equal to v_1 and also 6. If it only has one children its sum is less than 6 and so we have two numbering and just one restriction and it can be handled. Other wise there is at least three choice and two restriction and so it can be handled. So the paths which are begin with these two vertices can be labeled with pattern 1 and 4.
Now check all the vertices with only one child. we label their child with 1. So their sums are 3 or 5. If a vertex label is 1, it's sum would be 5 and it's path can be labeled with pattern 4. If its label is 2, it's sum would be 3 and can be labeled with pattern 2. Also till now only $V1$ and $V2$ are set, and their sum is at least 4 so there would be no problem with sum 3 of their neighbors. Sum of other vertices in cycles which don't have any children is also 4 or 2 and so there would be no problem in sum of vertices in cycle.
Now remains all the vertices which have more than 1 children, first we label children of vertices with label 2. There is only two restriction for them, their neighbors sum, and there is at least three numbering for them. It should be noted that since none of the vertices with label 1 have sum 6 then, the sub graph mentioned above which make a problem doesn't arise and so paths which begin from these vertices can be labeled with pattern 2 and 3.
Then we are going to vertices with label 1 which have more than 1 children. Since now all the other vertices are labeled, there would be no more restriction for their sum to become 6. So they only have two restriction, their neighbors sums, and three numbering and so they can be labeled. So all the vertices in this case are labeled and the base of induction is true in this case.
Case 2: Now we are checking the other case, If the first case is not true, then for every vertex v which has any children out of cycle, its neighbors in cycle are childless. We label v and one of its neighbor v_2 2, and label its other neighbor, v_1 , 1 and then labeling the others like an even cycle, labeling every other vertices with 1 and 2. All of v children are labeled 2, and so the its paths can be labeled via pattern 3. Also v's sum is bigger than its two neighbor and hence there would be no problem. Now it can be seen that sum of v_2 is 3 , and v_1's is 4 and since they don't have any children according to this case, their sums are not going to change, we can omit these three vertices and label the remaining even cycle, since the initial sum of neighbors of v_1 and v_2 are equal to their sum in proper even cycle. According to what is told about the even cycles, sum of vertices with label 1 are at least 4 and since neighbor of v_2 don't have any problem with v_2's sum. In even cycles, vertices with label 2 are also 2 or odd and so there is no problem with v_1 sum. So this case can be labeled this way and the Induction base is proven in all cases.

$\square$

By proving Theorem 3 we proved that lucky number of unicyclic graphs are 3 in general and 2 if they are not odd cycle. Now we prove a sufficient condition for lucky number of 2 in bipartite graphs.

Theorem 3. *Let $G = (X, Y)$ be a bipartite graph and $\mid X \mid= m$ and $\mid Y \mid= n, m \leq n$. If the number of matching of G with size m is odd, then $\xi(G) \leq 2$.*

Proof. Let

$$M = \left[\begin{array}{c|c} 0 & B \\ \hline B^T & 0 \end{array}\right]$$

be the adjacency matrix of G, where B is an $m \times n$ matrix. Consider A as a matrix over $\mathbb{Z}_2$. Clearly, $det(A) \equiv per(A) = k^2 = 1$ where k is the number of G's matching of size m. Since A is non-singular, we find that $rank(B) = m$. This implies that there is a vector $Z = [z_1, ..., z_n]$ such that $BZ = j$. Now, assign the value z_i to y_i, for $i = 1, ..., n$ and assign 2 to each x_i to obtain a lucky labeling with $\{1, 2\}$. $\square$

Now we can prove this theorem on edge lucky labeling of graphs.

Lemma 2. *Let T be a tree and the block decomposition of $L(T)$ is a star in which every leaf block is $K_{\Delta(T)}$. Then $\eta(L(T)) \leq \Delta(T)$.*

Proof. Assign value 1 to all vertices of block center B of $L(T)$. Let $B_1, B_2,, B_k$ be all leaf block of $L(T)$ and $B_i = K_{\Delta(T)}$, for $i = 1,, k$. Assume that $V(B) - V(B_i) = \{V_i\}$, for $i = 1, , k$. Now, assign values $1, 2,, i-1, i+1,, \Delta(T)$ to vertices of $V(B_i)\{V_i\}$, for $i = 1,, k$. Obviously, this is a lucky labeling for $L(T)$ and so $\eta(L(T)) \leq \Delta(T)$.

$\square$

Theorem 4. *Let T be a tree. Then $\eta(L(T)) \leq \Delta(T)$.*

Proof. Clearly, each blog of $L(T)$ is a complete graph. For the proof we use induction on the blocks of $L(G)$. For the step each time we take one or more than one complete graph from $line(G)$ and label the remaining properly and then prove that the taken part can be lucky labeled and attach to this graph.

Induction Step: Choose a K_Δ graph in $line(G)$ as root. Now assume that G_1 is one of the furthest graph from the root. We want to cut G_1 and label the remaining with strong induction and then connect G_1 back. To do so we name connective vertex of G_1 to remaining part of the graph, v. We call the neighbors of v outside of G_1 set $out(v)$ and neighbors of v inside G_1, $in(v)$. So the problem here is to label vertices in $in(v)$ so that two following conditions are met:

First) sum of the vertices in G_1 is not equal.
Second) sum of v would not be equal to sum of the vertices in $out(v)$.
If these can be done properly G_1 can be attached to the remaining part of the graph which is lucky labeled before by induction step.
Now we try two satisfy both condition above.
First) There are two kind of vertices in G_1, v and vertices in $in(v)$. If the sum of vertices in $in(v)$ is not going to be equal, they all should be labeled with different numbers. This is true since for any v_1 and v_2 in $in(v)$ we have:

$$\begin{aligned} sum(v_1) &= (\textstyle\sum_{u \in in(v)} label(u)) + label(v) - label(v_1) \\ sum(v_2) &= (\textstyle\sum_{u \in in(v)} label(u)) + label(v) - label(v_2) \\ sum(v_1) = sum(v_2) &\iff label(v_1) = label(v_2) \end{aligned}$$

If sum of the vertex v_1 in $in(v)$ is not going to be equal to sum of v the following condition should be met:

$$\begin{aligned} sum(v_1) &= (\textstyle\sum_{u \in in(v)} label(u)) + label(v) - label(v_1) \\ sum(v) &= (\textstyle\sum_{u \in in(v)} label(u) + (\textstyle\sum_{w \in out(v)} label(w)) \\ sum(v_1) = sum(v) &\iff label(v_1) = label(v) - (\textstyle\sum_{w \in out(v)} label(w)) \end{aligned}$$

We call the value of $label(v) - (\sum_{w \in out(v)} label(w))$ *restrict value*. If *restrict value* is in set $\{1, 2, ..., \Delta\}$ then we can not use this value in our labeling. Second) Now according to previous condition for labeling each vertex in $in(v)$ we have a set of labels with size at least equal to $\Delta - 1$. Also according to definition of $line(G)$ we have that

$$1 \leq size(in(v)), size(out(v)) \leq \Delta - 1$$

Now we use following lemma to label the vertices in $in(v)$.

Lemma 3. *If $size(in(v)) < \Delta - 1$ then G_1 can be lucky labeled correctly.*

Proof. As with conditions first and second above, the labels of $in(v)$ should be chosen differently and from a set with at least $\Delta - 1$ element. It can be seen that if size of $out(v)$ is equal to $\Delta - 1$ then the restrict value defined above cannot be positive. It is because that we chose the element of $out(v)$ from a set of distinct value and at most two of them can be equal so their sum is much greater than Δ. In this case we can choose Δ value for labels of $in(v)$. So we have $\binom{\Delta}{in(v)}$ value for labeling the vertices of $in(v)$ and since $1 \leq in(v) \leq \Delta - 1$ there is at least Δ different values of $\sum_{u \in in(v)} label(u)$. So at most $\Delta - 1$ number of these labeling cause the sum of v get equal to one

vertices in $out(v)$, hence we can find a proper labeling in this case.
The other case is where the size of $out(v)$ is less than $\Delta - 1$. In this case we can see that there is a possibility for the *restrict value* to be positive. If it is positive then we cannot use this number in our labeling for vertices of $in(v)$. So here we have exactly $\Delta - 1$ number to choose from and at most $\Delta - 2$ vertices in $in(v)$ to label them. So again it can be seen that they provide at least $\Delta - 1$ distinct value for sum so there is at least one labeling for the vertices in $in(v)$ where it can meet all the $out(v)$ constraints. □

Now it remains the part when size of $in(v)$ is equal to $\Delta - 1$. In this part if the *restrict value* is positive then only remains one labeling for $in(v)$ and so it would make problem. To handle this problem it can be seen that in any labeled graph in previous part only the vertex with maximum value of label may have its *restrict value* positive. Also since the order of labeling is not important in their sum, we can always change the order of labeling as we want and it would not change the result.
So in our labeling we should always keep in mind that if a the vertex is connected to an K_Δ we do not assign the maximum value to it. So if G_1 is K_Δ and the maximum value is not assigned to v we can be sure that *restrict value* is not positive and thus remains Δ value to choose for $in(v)$ and so they create Δ distinct sum which meets the at most $\Delta - 1$ constraints of $out(v)$.
Now just remain one part, when all the vertices of a graph we want to label are connected to a K_Δ then we cannot avoid assigning the maximum label to one of the vertices in K_Δ. In this part, instead of taking one complete graph as G_1 we take its parent graph with its child which is a complete graph with M vertices and any of its vertices is connected to a K_Δ. So here v is the vertices which is connected to M vertices of $in(v)$ which they all are in separate K_Δ graphs.
Now first lets assume that $M > 1$. In this case we do not need to label the vertices in $in(v)$ so that their sum do not get equal because afterward we are going to label the K_Δ graphs with Δ value and hence they can provide Δ distinct sum so sum of any vertices in $in(v)$ would be unequal to others vertices in $in(v)$.
So we have at least $\Delta - 1$ value for vertices of $in(v)$ and they can be chosen equal too, since we do not need to check the equality of vertices sum. One way to label vertices is to label two of them with set of pairs $\{(\Delta,\Delta),(\Delta,\Delta-1),(\Delta-1,\Delta-1),(\Delta-1,\Delta-2),...\}$ minus pairs having *restrict value* which creates more than Δ sum and set the remaining vertices all equal. All of the labeling defined above would not create any positive *restrict value* for vertices in K_Δ and there would be at least one lucky label of them that meets the constraints of $out(v)$.
Afterward we can label all the vertices in K_Δ with Δ different sum and so sum of any vertices in $in(v)$ after labeling the K_Δ attached to it, would be unequal to remaining vertices in $in(v)$.
The remaining part is when $M = 1$. This part is again need to be verified one step deeper. Current graph is a vertex with no other neighbor on its level and one parent outside and a K_Δ graph as its child. We call this kind of graph $G_{1,\Delta}$. In this part, we take graph G_1 the parent of one graph $G_{1,\Delta}$. So G_1 is a graph which its v is set before hand and $out(v)$ are its neighbor outside. $in(v)$ now contains m vertices which are in one $G_{1,\Delta}$, called set M and n vertices which are in another type of graph, called N. Now we examine two part separately
First) $m = 1$ and $n = 0$: In this part we can there is at least one labeling for $in(v)$ which satisfy $out(v)$ constraints. Then after labeling this it would remains labeling the $G_{1,\Delta}$. It can be labeled

easily since there is just one constraints for labeling whole $G_{1,\Delta}$ and there are $\Delta - 1$ labeling for it with different labels in its root and thus it meet the constraints. Second) $m > 1$: In this part we first label all the root of K_Δ graphs is $G_{1,\Delta}$ graph with 1 so if we can label the $in(v)$ properly, for the vertices in K_Δ there would be no positive *restrict value* and so they can be labeled with Δ distinct sum and meet the constraints. To label the $in(v)$ vertices it can be seen that:

$$\text{for any } m \in M : sum(m) = \sum_{u \in M} label(u) + \sum_{w \in N} label(w) + label(v) - label(m) + 1$$
$$\text{for any } n \in N : sum(n) = \sum_{u \in M} label(u) + \sum_{w \in N} label(w) + label(v) - label(n)$$
$$\text{for any } n \in N, m \in M : sum(n) = sum(m) \iff label(m) = label(n) + 1$$

Also sum of two vertices in M or N would be equal if and only if label of them be equal. So there would be two *restrict value* one for M and one for N, but each of them can be used in the other set. So here is again possible to create at least Δ different sum for v with labeling the $in(v)$ and thus meet the $out(v)$ constraints. So this case is also labeled and with this one and we've check all the possibilities.

So with following lemmas we can always take one graph G_1 and label the remaining part with induction step and then attach G_1 back as it was described above.
Induction Base: Base is a complete graph K_Δ. We can always label this graph using Δ value. So Induction is true for any graph K_Δ.

We have proven induction base and step for any graph $line(G)$ and thus Theorem 5 correctness is proved.

□

Theorem 5. *Let G be a graph and a,b be two positive integers. If G has an $\{a, b\}$-lucky labeling, then except for finitely many coprime pairs, G has a $\{x, y\}$-lucky labeling.*

Proof. Let $f(v)$ and $g(v)$ be the number of adjacent vertices of v with label a and b, respectively. In the $\{a, b\}$-lucky labeling of G we can see that:
for every $(v, u) \in E(G)$:

$$f(v)a + g(v)b \neq f(u)a + g(u)b$$

So we can write for any $(v, u) \in E(G)$
$(f(v) - f(u))a \neq (g(u) - g(v))b$
Since the graph was labeled properly the above equation is always true for any vertices of an edge. According to this fact, for any pair of adjacent v, u $f(v) - f(u)$ and $g(u) - g(v)$ cannot be zero simultaneously. So we can define a fraction equal to $\frac{f(v)-f(u)}{g(v)-g(u)}$ or $\frac{g(v)-g(u)}{f(v)-f(u)}$ respectively. This way, we have a fraction for each edge. Now it can be seen that all this fractions are constant numbers so there would be at most *edge* pair of numbers with $gcd = 1$ which would be equal to one of them and thus giving an invalid labeling of vertices. This results in this fact that just a limited number

of pairs cannot be chosen for labeling the graph, the others all would work the same as a, b in the labeling.

$\square$

Let n be a positive integer, $0 \leq p \leq 1$. The random graph $G(n, p)$ is probability space over the set of graphs on the vertex set $\{1, ..., n\}$ determined by
$Pr[\{i, j\} \in G] = p$ [Alon-Spenser]
We define function $f(G)$ the probability that G cannot be labeled with any two number where G is a random graph.

Theorem 6. $f(G) \leq \binom{n}{2} p \times (2p(p-1)+1)^{n-2}$ *where n is the number of vertices in G .*

Proof. For this proof we first prove the following lemma which is a sufficient condition for graph G to be labeled with a pair of two number.

Lemma 4. *Graph G can be labeled with a pair of two numbers, if there is no adjacent pair of vertices v, u which are having exactly the same number of adjacent vertices.*

$\nexists (u, v) \in E(G) : deg_u = deg_v$

Proof. If for all of adjacent vertices v, u $deg_u \neq deg_v$ then we can $\square$

With following lemma, we can see that if there is no pair of such vertices the graph would be labeled with a pair of numbers. We define $g(G)$ the probability that there is at least two adjacent vertices with same degree. So according to following lemma $f(G)$ defined above is smaller or equal to $g(G)$. Now we shall calculate $g(G)$. For that we define independent variable $X_{i,j}$ accordingly:
$X_{i,j}$:The event that there exists and edge between i , j vertices and their degree is equal.
Now it can be seen that $P(X_{i,j})$ is equal to the conditional probability of being an edge between i , j times the probability of them having both k neighbor. So

$$P(X_{i,j}) = p \times \sum_{k=1}^{n-2} (\binom{n-2}{k} \times p^k \times (1-p)^{n-k})^2).$$

$\square$

So with Theorem 5 it can be seen that if we choose a graph at random the probability that this graph can't be labeled with any two number tends to zero.

Theorem 7. *Any graph can be labeled with $\chi(G)$ number.*

Proof. We prove that any graph can be labeled with numbers $1, n, n^2, ...n^{\chi(G)}$. $\square$

Theorem 8. *Lucky Labeling with 0,1 is NP-Hard.*

Proof. We prove that this problem can be reduced to known NPC problem 3-SAT e.g if we can solve lucky labeling with 0,1 in polynomial time we can do it to 3-SAT too. □

Conjecture. *Edge Lucky number of a unicyclic graph G is Δ.*

Conjecture. *Edge Lucky number of a Regular graph G with vertex coloring Δ is Δ*

Conjecture. *If a graph can be labeled with 0,1 then it there is a proper labeling for it with 1,2 too.*

Conjecture. *Any graph with n vertices can be labeled with set 1,2,3,...,n .*

Conjecture. *Let $x, y \in \mathbb{R}$. Then every bipartite graph has $\{x, y\}$-lucky labeling.*