

Modular Harshad Labeling of Graphs: Structure and Enumeration

V. T. Dhokrat¹, P. G. Jadhav², S. B. Gaikwad³

¹Research Scholar, Department of Mathematics, New Art's, Commerce and Science College, Ahmednagar, Maharashtra, India.

²Department of Mathematics, Dnyaneshwar Gramonnati Mandal's Hon. B.J. Art's, Comm. and Sci. College, Ale Phata, Ta: Junnar Dist: Pune, Maharashtra, India

³Department of Mathematics, New Art's, Comm. and Sci. College, Ahmednagar, Maharashtra, India.

Email: vibhadhokrat@gmail.com, pgjmaths@gmail.com, sbgmathsnagar@gmail.com

Corresponding Author : 1) V. T. Dhokrat

Contributory Author: 2) Dr. P. G. Jadhav 3) S. B. Gaikwad.

These authors contributed equally to this work.

Keywords and phrases : Harshad number, graph labeling, modular labeling, bipartite graph, path, enumeration.

Abstract

The great German mathematician Carl Friedrich Gauss said: "Mathematics is the queen of the sciences and number theory is the queen of mathematics. In number theory we generally study numbers as well as the properties of mathematical objects constructed from numbers. Harshad number is one of the interesting number system in number Theory. Graph labeling is an important research area in graph theory with numerous applications in communication networks, coding theory, and combinatorial optimization. In this paper we introduce a finite modular graph-labeling model in which every vertex label is a base-10 Harshad number. Fix integers $K \geq 3$, $N \geq 1$, and a residue r satisfying $2r \not\equiv 0 \pmod{K}$. The permitted labels are the Harshad numbers not exceeding N that are congruent to either r or $-r$ modulo K , and adjacent labels must have sum divisible by K . For connected graphs, we prove that such a labeling exists precisely when the graph is bipartite and the two residue pools are large enough for its bipartition classes. We then derive an exact enumeration formula for connected bipartite graphs, with special cases for paths and stars. We extend the enumeration to disconnected graphs through a residue-orientation polynomial and give consequences for trees, cycles, and complete bipartite graphs. We also establish symmetry, minimum-cutoff, balanced-pool, and edge-stability results. Explicit examples and a verification algorithm are provided.

How to cite this article: V. T. Dhokrat P. G. Jadhav2 S. B. Gaikwad3. Modular Harshad Labeling of Graphs: Structure and Enumeration. Int J Drug Deliv Technol. 2026;16(84s):2079-2085. DOI: 10.25258/ijddt.16.84s.210.

Introduction

Graph labeling has evolved into a major and highly active area of research in discrete mathematics, with significant theoretical and applied implications. Graph labeling assigns non-negative integers to vertices, edges, or both, subject to specified constraints. Since Rosa's early work, many variations of graph labeling have been developed and studied. Broad accounts of graph theory and the graph-labeling literature are available. A positive integer is a base-10 Harshad (or Niven) number when it is divisible by the sum of its decimal digits. The digit-sum condition and a modular edge condition are logically different. Therefore, both conditions must be stated explicitly when they are combined in a graph-labeling model. We do this by selecting a finite pool of Harshad numbers and imposing a congruence on the sums of adjacent labels. Finiteness of the pool makes enumeration meaningful, while the use of two opposite residue classes produces a precise connection with bipartite graphs.

The labeling introduced here is not a graceful labeling. Graceful labeling is based on distinct absolute differences of endpoint labels, whereas the present model is based on modular sums and Harshad vertex labels.

Definitions

All graphs in this paper are finite and simple. For a positive integer m , let $s_{10}(m)$ denote the sum of its decimal digits.

Definition 1. A positive integer m is a *Harshad number* if

$$s_{10}(m) \mid m.$$

Let $\mathcal{H}(N)$ be the set of Harshad numbers not exceeding N .

Fix $K \geq 3$ and a residue $r \in \{1, \dots, K-1\}$ such that $2r \not\equiv 0 \pmod{K}$. Define the two finite label pools

$$A_r(K, N) = \{h \in \mathcal{H}(N) : h \equiv r \pmod{K}\}$$

and

$$A_{-r}(K, N) = \{h \in \mathcal{H}(N) : h \equiv -r \pmod{K}\}.$$

Write $a = |A_r(K, N)|$ and $b = |A_{-r}(K, N)|$.

Definition 2. Let $G = (V, E)$. A restricted modular (K, r, N) -Harshad labeling of G is an injective function

$$f: V \rightarrow A_r(K, N) \cup A_{-r}(K, N)$$

such that

$$f(u) + f(v) \equiv 0 \pmod{K}$$

for every edge $uv \in E$.

The restriction $2r \not\equiv 0 \pmod{K}$ ensures that two labels from the same residue pool cannot be assigned to adjacent vertices.

Structural Characterization

Theorem 1. Let G be a connected graph with bipartition-class candidates of sizes x and y . A restricted modular (K, r, N) -Harshad labeling exists if and only if G is bipartite and, for a bipartition (X, Y) of G , either $|X| \leq a$ and $|Y| \leq b$, or $|X| \leq b$ and $|Y| \leq a$.

Proof. Suppose that f is such a labeling. Every label has residue r or $-r$ modulo K . If adjacent vertices had labels from the same pool, their sum would have residue $2r$ or $-2r$, neither of which is zero. Hence every edge joins vertices assigned labels from opposite pools. These two residue classes give a bipartition of G . Injectivity gives the stated capacity inequalities.

Conversely, suppose G has a bipartition (X, Y) satisfying one of the two capacity conditions. Assign distinct labels from one pool to X and distinct labels from the opposite pool to Y . Each edge joins the two classes, so its endpoint labels have sum congruent to $r + (-r) = 0$ modulo K . The resulting map is injective and is therefore a restricted modular Harshad labeling. ■

Corollary 1. No graph containing an odd cycle admits a restricted modular (K, r, N) -Harshad labeling.

Proof. A graph containing an odd cycle is not bipartite, so the result follows from Theorem[3.1].

Enumeration

For an integer $t \geq 0$, define the falling factorial by

$$(t)_j = t(t-1) \cdots (t-j+1),$$

with $(t)_0 = 1$ and $(t)_j = 0$ when $j > t$.

Theorem 2. Let G be a connected bipartite graph with bipartition sizes x and y . The number of restricted modular (K, r, N) -Harshad labelings of G is $L_{K,r,N}(G) = (a)_x(b)_y + (b)_x(a)_y$.

Proof. In the first orientation, the x vertices of the first class receive distinct labels from the a -element pool and the y vertices of the second class receive distinct labels from the b -element pool. This gives $(a)_x(b)_y$ labelings. Interchanging the pools gives $(b)_x(a)_y$ further labelings. The two collections are disjoint because the residue pools are disjoint. ■

Corollary 2 (Paths). For the path P_n , let $x = \lfloor n/2 \rfloor$ and $y = \lfloor n/2 \rfloor$.

Then $L_{K,r,N}(P_n) = (a)_x(b)_y +$

$(b)_x(a)_y$.

Corollary 3 (Stars). For the star $K_{1,m}$, $L_{K,r,N}(K_{1,m}) = a(b)_m + b(a)_m$.

Disconnected graphs and an orientation polynomial

For a disconnected graph, the components share the same finite label pools. Consequently, their numbers of labeling cannot simply be multiplied: a label used in one component is unavailable in every other component. The following polynomial records the possible orientations of all bipartition classes.

Let G have connected components $G_1, \dots, G_c$. If every component is bipartite, choose a bipartition (X_i, Y_i) of G_i and put $x_i = |X_i|$ and $y_i = |Y_i|$. Define the residue-orientation polynomial by

$$R_G(z) = \prod_{i=1}^c (z^{x_i} + z^{y_i}) = \sum_{s=0}^{|V(G)|} c_s z^s.$$

The coefficient c_s is the number of ways to orient the components so that exactly s vertices receive labels from $A_r(K, N)$ and the remaining $|V(G)| - s$ vertices receive labels from $A_{-r}(K, N)$. Swapping the names of X_i and Y_i does not change $R_G(z)$.

Theorem 3. If G has a nonbipartite component, then $L_{K,r,N}(G) = 0$. Otherwise, writing $n = |V(G)|$ and $R_G(z) = \sum_s c_s z^s$, one has $L_{K,r,N}(G) = \sum_{s=0}^n c_s (a)_s (b)_{n-s}$. In particular, a labeling exists if and only if there is an index s for which $c_s > 0$, $s \leq a$, and $n - s \leq b$.

Proof. A nonbipartite component cannot be labeled by Theorem[1]. Suppose all components are bipartite. For each component, choose which bipartition class uses $A_r(K, N)$. If the chosen orientations place s vertices in that residue class, then the injective assignments can be made in $(a)_s(b)_{n-s}$ ways. The polynomial coefficient c_s counts all orientation choices that produce this value of s . Summing over s proves the formula. A summand is positive exactly when the corresponding two pool capacities are sufficient.

The polynomial is multiplicative over disjoint unions:

$$R_{G \cup H}(z) = R_G(z)R_H(z).$$

This multiplicativity is useful even though the labeling count itself is not generally multiplicative.

Example 1. Let $G = P_3 \cup P_2$. The components have bipartition sizes $(2, 1)$ and $(1, 1)$, so

$$R_G(z) = (z^2 + z)(2z) = 2z^3 + 2z^2.$$

Since $|V(G)| = 5$, Theorem[4.4] gives

$$L_{K,r,N}(G) = 2(a)_3(b)_2 + 2(a)_2(b)_3.$$

The factors of two record the two residue orientations of the balanced component P_2 .

Consequences for standard graph families

The enumeration formula depends on a connected bipartite graph only through the sizes of its two color classes. Thus graphs with different edge sets can have the same number of restricted modular Harshad labeling.

Corollary 4 (Trees). *Let T be a tree whose unique bipartition has class sizes x and y . Then $L_{K,r,N}(T) = (a)_x(b)_y + (b)_x(a)_y$. Therefore, two trees with the same bipartition sizes have the same number of labeling in the present model.*

Proof. Every tree is connected and bipartite, and its bipartition (X, Y) is unique up to interchanging the two classes. Let $|X| = x$ and $|Y| = y$.

First assign labels from $A_r(K, N)$ to the vertices of X and labels from $A_{-r}(K, N)$ to the vertices of Y . Because the labeling is injective, the x vertices of X can be labeled in $(a)_x$ ways, and the y vertices of Y can then be labeled in $(b)_y$ ways. This orientation therefore contributes $(a)_x(b)_y$ labelings. Every edge of T joins X to Y , so the two endpoint residues are r and $-r$; hence every edge sum is divisible by K .

In the reverse orientation, X receives labels from $A_{-r}(K, N)$ and Y receives labels from $A_r(K, N)$. This contributes $(b)_x(a)_y$ labeling. The two collections are disjoint because $A_r(K, N) \cap A_{-r}(K, N) = \emptyset$, which follows from $2r \not\equiv 0 \pmod{K}$. Adding the two contributions gives

$$L_{K,r,N}(T) = (a)_x(b)_y + (b)_x(a)_y.$$

The expression depends only on x and y , so any two trees with the same bipartition sizes have the same number of labelings.

Corollary 5 (Cycles). *For $m \geq 2$, $L_{K,r,N}(C_{2m}) = 2(a)_m(b)_m$, whereas $L_{K,r,N}(C_{2m+1}) = 0$.*

Proof. Write $V(C_{2m}) = \{v_1, v_2, \dots, v_{2m}\}$ in cyclic order. The sets

$X = \{v_1, v_3, \dots, v_{2m-1}\}$ and $Y = \{v_2, v_4, \dots, v_{2m}\}$ form a bipartition of C_{2m} , and both have size m .

If X receives distinct labels from $A_r(K, N)$ and Y receives distinct labels from $A_{-r}(K, N)$, there are $(a)_m(b)_m$ possible injective labeling. Every edge joins X to Y , so the residues of its endpoint labels are r and $-r$ and their sum is divisible by K . Interchanging the two pools gives another $(b)_m(a)_m$ labelings. Since the pools are disjoint, the two cases do not overlap. Therefore, $L_{K,r,N}(C_{2m}) = (a)_m(b)_m + (b)_m(a)_m = 2(a)_m(b)_m$.

Now consider C_{2m+1} . Along any edge, the endpoint labels must come from opposite residue pools because $2r \not\equiv 0 \pmod{K}$. Starting at any vertex and proceeding around the cycle therefore forces the two pools to alternate. After an odd number of edges, the last vertex and the first vertex are forced to lie in the

same pool, although they are adjacent. This is impossible. Hence $L_{K,r,N}(C_{2m+1}) = 0$.

Corollary 6 (Complete bipartite graphs). *For $p, q \geq 1$, $L_{K,r,N}(K_{p,q}) = (a)_p(b)_q + (b)_p(a)_q$. Such a labeling exists precisely when the two class sizes fit into the two label pools in at least one orientation.*

Proof. Let (X, Y) be the natural bipartition of $K_{p,q}$, where $|X| = p$ and $|Y| = q$. Every vertex of X is adjacent to every vertex of Y , and there are no edges within either class.

First assign distinct labels from $A_r(K, N)$ to the vertices of X and distinct labels from $A_{-r}(K, N)$ to the vertices of Y . There are $(a)_p(b)_q$ such injective assignments. For every edge xy , with $x \in X$ and $y \in Y$, the endpoint residues are r and $-r$. Hence $f(x) + f(y) \equiv r - r \equiv 0 \pmod{K}$,

so each assignment is a valid restricted modular Harshad labeling.

In the reverse orientation, X receives distinct labels from $A_{-r}(K, N)$ and Y receives distinct labels from $A_r(K, N)$. This gives $(b)_p(a)_q$ additional labelings. The two collections are disjoint because the pools are disjoint. Therefore,

$$L_{K,r,N}(K_{p,q}) = (a)_p(b)_q + (b)_p(a)_q.$$

The first term is positive exactly when $p \leq a$ and $q \leq b$, while the second term is positive exactly when $p \leq b$ and $q \leq a$. Consequently, a labeling exists precisely when the two class sizes fit into the two label pools in at least one of these orientations. ■

Examples

Take $K = 5$, $r = 1$, and $N = 120$. Direct calculation gives

$$A_1(5, 120) = \{1, 6, 21, 36, 81, 111\}$$

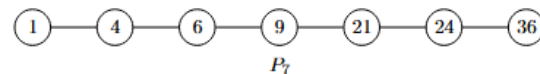
and $A_{-1}(5, 120) = A_4(5, 120) = \{4, 9, 24, 54, 84, 114\}$.

Every displayed number is Harshad, and both pools have size six.

Example 2 (A labeling of P_7). For the vertices $v_1, \dots, v_7$ in path order, define

$$(f(v_1), \dots, f(v_7)) = (1, 4, 6, 9, 21, 24, 36).$$

The consecutive sums are 5, 10, 15, 30, 45, and 60, all divisible by 5. Thus this is a restricted modular $(5, 1, 120)$ -Harshad labeling.



Since $a = b = 6$, Theorem[4.1] gives

$$L_{5,1,120}(P_7) = 2(6)_4(6)_3 = 86,400.$$

Example 3 (A labeling of $K_{1,6}$). Assign the central vertex the label 1 and assign the six leaves the labels 4, 9, 24, 54, 84, and 114. Every edge sum is divisible by 5, and all labels are distinct Harshad numbers. The total number of such labelings is

$$L_{5,1,120}(K_{1,6}) = 6(6)_6 + 6(6)_6 = 8,640.$$

Example 4 (A nonexample). The triangle C_3 has no restricted modular $(5, 1, 120)$ -Harshad labeling, because it is not bipartite. This illustrates why the condition $2r \not\equiv 0 \pmod{K}$ is essential to the structural theorem.

Effect of the label bound

The bound N is part of the definition, and increasing it can only enlarge the two label pools. For $K = 5$ and $r = 1$, the following table shows the pool sizes obtained by direct digit-sum testing. The final column gives the largest path supported by the corresponding capacities.

N	$ A_1(5, N) $	$ A_4(5, N) $	largest admissib le path
30	3	3	P_6
60	4	4	P_8
90	5	5	P_{10}
120	6	6	P_{12}
0			
20	9	7	P_{15}
0			

For example, P_{15} has color-class sizes 8 and 7. When $N = 200$, the larger class can use the nine-element pool and the smaller class can use the seven-element pool. In contrast, P_{16} has two classes of size eight and cannot be labeled because the smaller pool contains only seven labels.

More generally, if $N_1 \leq N_2$, then

$$A_{\pm r}(K, N_1) \subseteq A_{\pm r}(K, N_2).$$

It follows immediately that

$$L_{K,r,N_1}(G) \leq L_{K,r,N_2}(G)$$

for every graph G . Thus the labeling count is monotone in the cutoff N .

Results

Symmetry of the residue classes

Replacing r by $-r$ merely interchanges the names of the two label pools. Thus the labeling model has a basic residue symmetry.

Theorem 4 (Residue symmetry). *For every graph G , $L_{K,r,N}(G) = L_{K,K-r,N}(G)$.*

Proof. Modulo K , the residues $K - r$ and $-(K - r)$ are $-r$ and r , respectively. Hence the union of the two permitted pools is unchanged, and the edge condition is identical. Every labeling counted on one side is therefore counted on the other side. ■

The orientation polynomial has a stronger coefficient symmetry.

Theorem 5 (Palindromicity). *Let G be a bipartite graph on n vertices. Then $z^n R_G(z^{-1}) = R_G(z)$. Equivalently, if $R_G(z) = \sum_{s=0}^n c_s z^s$, then $c_s = c_{n-s}$ for every s .*

Proof. For a component with bipartition sizes (x_i, y_i) and $x_i + y_i = n_i$,

$$z^{n_i}(z^{-x_i} + z^{-y_i}) = z^{y_i} + z^{x_i}.$$

Multiplying this identity over all components gives the stated reciprocal identity because $\sum_i n_i = n$. Equality of the corresponding coefficients follows. ■

In particular, the feasible numbers of vertices assigned to the first pool occur symmetrically around $n/2$. This is the polynomial form of reversing all component orientations simultaneously.

The minimum feasible label bound

For fixed K and r , list the Harshad numbers in the two residue classes in increasing order:

$$\alpha_1 < \alpha_2 < \dots, \quad \beta_1 < \beta_2 < \dots,$$

where $\alpha_j \equiv r \pmod{K}$ and $\beta_j \equiv -r \pmod{K}$. If a required term does not exist, assign it the formal value $+\infty$. Define

$$N_{\min}(G; K, r)$$

to be the smallest cutoff N for which G has a restricted modular (K, r, N) -Harshad labeling.

Theorem 6 (Minimum cutoff). *Let G be a connected bipartite graph with class sizes x and y . Then $N_{\min}(G; K, r) = \min\{\max(\alpha_x, \beta_y), \max(\alpha_y, \beta_x)\}$. If G is not bipartite, then $N_{\min}(G; K, r) = +\infty$.*

Proof. In the first residue orientation, the pool $A_r(K, N)$ must contain at least x labels and $A_{-r}(K, N)$ must contain at least y labels. The smallest cutoff for which both conditions hold is $\max(\alpha_x, \beta_y)$. Reversing the orientation gives $\max(\alpha_y, \beta_x)$. Taking the smaller value gives the optimal cutoff. A nonbipartite graph is excluded by Theorem[3.1] for every N . ■

Example 5. For $K = 5$ and $r = 1$, the seventh and eighth terms of the first pool sequence are $\alpha_7 = 126$ and $\alpha_8 = 156$, while $\beta_7 = 144$ and $\beta_8 = 204$. Since P_{15} has class sizes 8 and 7,

$$N_{\min}(P_{15}; 5, 1) = \min\{\max(156, 144), \max(126, 204)\} = 156.$$

For P_{16} , both classes have size eight, so the minimum cutoff is 204. These exact thresholds refine the coarser observations in the preceding table.

Balanced pools

The formulas simplify substantially when the two residue pools have equal size.

Corollary 7 (Balanced-pool formula). *Suppose $a = b = t$ and G is connected and bipartite with class sizes x and y . Then $L_{K,r,N}(G) = 2(t)_x(t)_y$. A labeling exists if and only if $\max\{x, y\} \leq t$.*

Proof. Substitute $a = b = t$ into Theorem[4.1]. The product is nonzero exactly when both x and y are at most t . ■

Corollary 8 (Balanced-pool path formulas). *If $a = b = t$, then for every $m \geq 1$, $L_{K,r,N}(P_{2m}) = 2(t)_m^2$ and $L_{K,r,N}(P_{2m+1}) = 2(t)_{m+1}(t)_m$. Consequently,*

whenever the quantities are nonzero, $\frac{L_{K,r,N}(P_{2m+1})}{L_{K,r,N}(P_{2m})} = t - m$.

Proof. The two color classes of P_{2m} both have size m , while those of P_{2m+1} have sizes $m + 1$ and m . Apply Corollary[6.5] and cancel common falling-factorial terms in the ratio. ■

Stability under edge additions

The next result explains precisely which edge additions change the set of labelings.

Theorem 8 (Edge stability). *Let G be a connected bipartite graph with bipartition (X, Y) .*

1. *If H is obtained from G by adding edges only between X and Y , then every restricted modular Harshad labeling of G is a labeling of H , and $L_{K,r,N}(H) = L_{K,r,N}(G)$.*
2. *If an edge is added between two distinct vertices of X , or between two distinct vertices of Y , then the resulting graph has no restricted modular (K, r, N) -Harshad labeling.*

Proof. In every labeling of G , one bipartition class uses one residue pool and the other class uses the opposite pool. Therefore, the labels at the ends of any new edge between X and Y still have sum divisible by K . No new restriction is introduced, proving the first statement.

Two vertices in the same class of a connected bipartite graph are joined by an even-length path. Adding an edge between them creates an odd cycle. The second statement now follows from the odd-cycle corollary. ■

Thus, for connected bipartite graphs, the labeling count is invariant under all edge additions that preserve the bipartition. This gives another proof that the count depends on the class sizes rather than on the number or arrangement of cross edges.

Verification Algorithm for Modular Harshad Labeling
Given G , K , r , N , and a proposed labeling f , verification consists of the following checks:

1. verify that the vertex labels are distinct positive integers not exceeding N ;
2. compute the decimal digit sum of every label and verify the Harshad divisibility condition;
3. verify that every label has residue r or $-r$ modulo K ; and
4. for every edge uv , verify $f(u) + f(v) \equiv 0 \pmod{K}$.

Apart from digit-sum computation, these checks take $O(|V| + |E|)$ arithmetic operations. Existence can be decided by a bipartite graph search followed by the two capacity tests in Theorem[3.1].

To construct a labeling of a connected graph, first run a breadth-first or depth-first search to obtain its bipartition (X, Y) . Test both possible orientations

against the capacities a and b . If one orientation succeeds, assign any $|X|$ distinct values from the corresponding pool to X and any $|Y|$ distinct values from the other pool to Y . No edge-by-edge search or backtracking is required after the bipartition is known. Computational representations of graceful graph labelings provide a related example of using programming to construct and inspect graph labelings. For a disconnected graph, the coefficients of $R_G(z)$ can be computed by dynamic programming. Begin with the coefficient array for the constant polynomial 1 and, for each component with class sizes (x_i, y_i) , shift and add the array by x_i and y_i . The resulting array has at most $|V(G)| + 1$ entries. It simultaneously decides feasibility and supplies the coefficients needed in Theorem[4.4]. A straightforward implementation uses $O(c|V(G)|)$ integer additions after the components and their bipartitions have been found.

Relation to Classical Labelings

Labeling	Vertex requirement	Edge requirement
Graceful	injection into $\{0, \dots, q\}$	distinct absolute differences
Harmonious	modular vertex labels	distinct modular sums
Present model	finite Harshad pools	sums congruent to zero modulo K

Applications

Graph-labeling methods have been used as abstract models in areas such as network addressing, channel assignment, coding, and database organization. The present model is not itself a deployed network or security protocol. Its direct value is as a mathematically tractable assignment and consistency model. The following applications are therefore interpretations of the proved results rather than claims of empirical performance.

1) Finite two-class resource assignment

Consider a system whose interaction graph is bipartite, with the two classes representing two different roles. Examples of such abstract roles include producers and consumers, requesters and providers, or transmitters and receivers. Suppose that each object must receive a distinct integer identifier from one of two finite catalogs and that identifiers on interacting objects must have sum divisible by K . The sets $A_r(K, N)$ and $A_{-r}(K, N)$ serve as the two identifier catalogs. Theorem[1] decides whether the catalogs have enough identifiers for the two roles, while Theorem[2] gives the exact number of feasible assignments. The minimum-cutoff theorem identifies the smallest numerical range $\{1, \dots, N\}$ that supports the system.

Thus the model can be used to study finite assignment problems in which uniqueness, two-role separation, and a modular compatibility rule are required simultaneously.

The edge-stability theorem is relevant in this interpretation. Adding a new interaction between opposite roles does not require relabeling, because every cross-class pair already satisfies the modular rule. Adding an interaction within one role, however, makes the constraints infeasible in the restricted two-residue model. system. Thus the model can be used to study finite assignment problems in which uniqueness, two-role separation, and a modular compatibility rule are required simultaneously.

The edge-stability theorem is relevant in this interpretation. Adding a new interaction between opposite roles does not require relabeling, because every cross-class pair already satisfies the modular rule. Adding an interaction within one role, however, makes the constraints infeasible in the restricted two-residue model.

2) Exact benchmarks for enumeration algorithms

The closed formulas in this paper provide graph instances whose answers are known exactly. They can therefore serve as small benchmark families for programs that generate constrained vertex assignments, verify injectivity, or count solutions of graph-based constraint systems. Paths, stars, cycles, and complete bipartite graphs give connected test cases with different bipartition sizes.

Disconnected graphs provide a second level of testing. Their counts are not obtained by simply multiplying component counts because all components share the same label pools. The residue-orientation polynomial and the dynamic programming procedure in Section 7 give exact reference values against which an implementation can be checked. The palindromic coefficient identity in Theorem[6.1] supplies an additional internal validation test.

3) Teaching and computational experimentation

The proposed labeling framework provides an accessible setting in which students can study several mathematical ideas simultaneously, including digit sums, divisibility, modular arithmetic, injective mappings, bipartite graphs, falling factorials, graph-search algorithms, and generating polynomials. In a classroom or computational laboratory, students may first construct the two Harshad residue pools for chosen parameters K , r , and N . They may then produce a valid graph labeling and verify that a computer program returns the value predicted by the corresponding enumeration formula. Additional investigations may focus on the dependence of the pool sizes on the cutoff N , the effect of unequal pool

capacities on graph feasibility, and the variation of the minimum admissible cutoff among trees with different bipartition sizes. Such activities demonstrate how number-theoretic restrictions influence graph structure and combinatorial counting, while remaining within the scope of the mathematical model developed in this paper.

Conclusion

The finite-pool formulation separates three ingredients that were previously conflated: the digit-sum definition of a Harshad number, injective vertex labeling, and a modular condition on adjacent sums. This produces a valid bipartite characterization with explicit capacity conditions and an exact enumeration formula for connected bipartite graphs.

The model can be studied further for disconnected graphs, alternative residue pairs, other numeral bases, and optimization of the bound N . The residue-orientation polynomial also suggests questions about which coefficient sequences can occur for prescribed graph classes and how pool imbalance affects the smallest feasible cutoff.

The present enumeration has an intentional limitation: among connected bipartite graphs it distinguishes only the two color-class sizes, not the detailed edge structure. Stronger variants could add local conditions, such as requiring the edge sums themselves to be Harshad numbers or requiring distinct edge sums. Such variants would have genuinely graph-dependent enumeration, but the structural and counting arguments would need to be developed again rather than inherited from the current model.

Possible computational applications should be presented as future research questions; claims concerning error detection, coding, security, or network performance require separate algorithms, experiments, and comparisons with established methods.

References

- 1] J. A. Gallian, *A Dynamic Survey of Graph Labeling*, The Electronic Journal of Combinatorics, Dynamic Survey DS6, twenty-first edition, 2018.
- 2] F. Harary, *Graph Theory*, Addison-Wesley, Reading, Massachusetts, 1969.
- 3] V. Nițică, High degree b -Niven numbers, arXiv:1807.02573, 2018.
- 4] R. Cattell, Graceful labellings of paths, *Discrete Math.*, 307 (2007) 3161–3176.
- 5] J. A. Gallian, A guide to the graph labeling zoo, *Discrete Appl. Math.*, 49 (1994) 213–229.
- 6] D. R. Kaprekar, On Harshad numbers, *J. Recreat. Math.* (1980), 81–82.
- 7] V. T. Dhokrat, P. G. Jadhav, and S. B. Gaikwad, A survey study of association between graph and graph labeling techniques, *International Journal of Education*,

Modern Management, Applied Science and Social Science, 5(3), 2023, 19–30.

8] V. T. Dhokrat, P. G. Jadhav, and S. B. Gaikwad, A survey study of some graph labeling techniques, Futuristic Trends in Contemporary Mathematics, e-ISBN: 978-93-6252-623-6, IIP Series, Volume 3, Book 1, Part 2, Chapter 5, (2023) 171-183.

9] V. T. Dhokrat, P. G. Jadhav, and S. B. Gaikwad, Representations of graceful labeling of a graph using Python programming language, The Indian Journal of Technical Education, 47, Special Issue 1 (2024), 157–163.

10] Wikipedia, Harshad number, https://en.wikipedia.org/wiki/Harshad_number.

11] GeeksforGeeks, Harshad or Niven number,