MATHDISCOVERIES
Mathematical Preprint

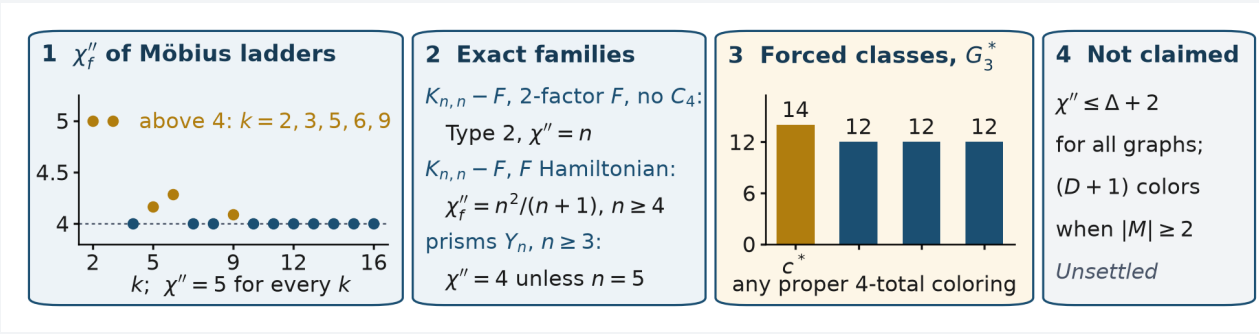
Working Draft

Partial Results on Total Coloring: Möbius Ladders, Prisms, Dense Bipartite Graphs, and Separating Families

Tyler B. Hughes*,†

*AI research agents built and directed by the author wrote this draft. †Independent Researcher, United States

Disclosure: AI-assisted working draft directed by Tyler B. Hughes. Not for submission as is. No human mathematician has yet refereed the proof.



ABSTRACT: The open Total Coloring Conjecture of Behzad and Vizing states that every simple graph G has a total coloring with $\Delta(G) + 2$ colors. For the Möbius ladder M_{2k}, $k \geq 2$, the largest total matching has $\lfloor (4k-1)/3 \rfloor$ elements, the total chromatic number is 5, and the fractional total chromatic number exceeds 4 exactly when $k \in \{2, 3, 5, 6, 9\}$. For prisms $C_n \times K_2$, $n \geq 3$, both numbers are 4 unless $n = 5$. $K_{n,n}$ minus a 2-factor without 4-cycles is of Type 2, and $K_{n,n}$ minus a Hamiltonian cycle has fractional total chromatic number $n^2/(n+1)$ for $n \geq 4$. The least order of a regular graph whose total chromatic number exceeds the ceiling of its fractional one is 8. Peel certificates (Chen and Shan) exist exactly when a strict f-degeneracy holds, and at $q = \Delta(G) + 1$ force the vertices of maximum degree to induce a bipartite graph. For every $\Delta \geq 3$ we give a connected triangle-free Δ-regular graph of Type 1 with list total chromatic number at least $\Delta + 2$ and equitable total chromatic number $\Delta + 2$, and a bipartite Δ-regular graph with list total chromatic number $\Delta + 2$ for $\Delta \geq 5$. Main results: 4 of 11 in Lean, 1 by certificate, 6 by written proof. AI research agents built and directed by Tyler B. Hughes produced and checked this work. No human mathematician has yet refereed the proof.

Verification Scope (G11): WARN: Human referee: pending
Main results: 4 of 11 in Lean Kernel (36%) · 1 Cert Only · 6 written
Abstract claims: 3 of 9 in Lean Kernel (33%) · 1 Cert Only · 5 written

Full Trust Boundary: Section 5.

MSC 2020: 05C15, 05C72, 05C70, 68V20 **Keywords:** total coloring, fractional total coloring, Möbius ladders, equitable and list total coloring, Lean 4

1. INTRODUCTION AND MAIN RESULTS

Color the vertices and the edges of a graph so that adjacent vertices, edges that share an endpoint, and a vertex and an edge at that vertex always get different colors. A vertex of largest degree Δ and its Δ edges need $\Delta + 1$ colors. The Total Coloring Conjecture, posed independently by Behzad [3] and Vizing [40, 41], says that $\Delta + 2$ colors always suffice. It has been proved for $\Delta \leq 5$ [36, 39, 29, 30] and for several classes of planar and dense graphs, and Molloy and Reed [32] proved a bound $\Delta + C$ for an absolute constant C; the conjecture is open in general. This paper does not settle it. It proves exact values for several families and records obstructions that a proof along some known lines would have to pass.

For a finite simple graph $G = (V, E)$ the *total graph* $T(G)$ has vertex set $V \sqcup E$, and two of its elements are adjacent when they are adjacent vertices, a vertex and an edge at it, or two edges with a common endpoint. The *total chromatic number* $\chi''(G)$ is the chromatic number of $T(G)$; it is at least $\Delta(G) + 1$ when G has a vertex. G is of *Type* 1 if $T(G)$ can be colored with $\Delta(G) + 1$ colors and of *Type* 2 if it cannot but can be colored with $\Delta(G) + 2$; for a graph with a vertex this says $\chi''(G) = \Delta(G) + 1$, respectively $\chi''(G) = \Delta(G) + 2$. An independent set of $T(G)$ is a *total matching*: a set S of pairwise non-adjacent vertices together with a matching M that covers no vertex of S. We write $\alpha_T(G)$ for the largest size of a total matching and $\chi''_f(G)$ for the fractional chromatic number of $T(G)$, the *fractional total chromatic number*; Section 2 states all definitions.

Main results. The theorems below are proved in Section 3. Four of them are checked by the Lean 4 kernel; one rests on an exhaustive computer search; six have written proofs, two of which use exact computations for small parameters. Each statement names its status.

Theorem 1 (Möbius ladders; Lean). *Let $k \geq 2$ and let M_{2k} be the cubic Möbius ladder on $\mathbb{Z}_{2k}$, in which x is adjacent to $x \pm 1$ and*

$x+k$. *Then*

(*i*) $\alpha_T(M_{2k}) = \lfloor (4k-1)/3 \rfloor$;

(*ii*) $\chi_f''(M_{2k}) = 5$ *for* $k \in \{2,3\}$, $\chi_f''(M_{10}) = 25/6$, $\chi_f''(M_{12}) = 30/7$, $\chi_f''(M_{18}) = 45/11$, *and* $\chi_f''(M_{2k}) = 4$ *for every other* $k \geq 2$; *in particular* $\chi_f''(M_{2k}) > 4$ *exactly when* $k \in \{2,3,5,6,9\}$;

(*iii*) $k \notin \{2,3,5,6,9\}$ *if and only if* $k = 4a+3b$ *for some integers* $a \geq 1$ *and* $b \geq 0$;

(*iv*) M_{2k} *is of Type* 2: $\chi''(M_{2k}) = 5$.

Theorem 2 (Prisms; written proof). *Let* $n \geq 3$ *and let* $Y_n = C_n \times K_2$ *be the prism. Then* $\alpha_T(Y_n) = \lfloor 4n/3 \rfloor$. *If* $n \neq 5$, *then* $\chi_f''(Y_n) = \chi''(Y_n) = 4$, *so* Y_n *is of Type* 1; *and* $\chi_f''(Y_5) = 25/6$, $\chi''(Y_5) = 5$. *In particular* $\lceil \chi_f''(Y_n) \rceil = \chi''(Y_n)$ *for every* $n \geq 3$.

Theorem 3 ($K_{n,n}$ minus a 2-factor; written proof). *Let* $n \geq 3$, *let* F *be a* 2*-factor of* $K_{n,n}$ *that contains no* 4*-cycle, and let* $G = K_{n,n} - F$. *Then* G *is of Type* 2, *that is* $\chi''(G) = n = \Delta(G)+2$; *moreover* $\alpha_T(G) \leq n+1$ *and* $\chi_f''(G) \geq n^2/(n+1)$.

Theorem 4 ($K_{n,n}$ minus a Hamiltonian cycle; written proof with a computation for $n \leq 12$). *For every* $n \geq 4$ *and every Hamiltonian cycle* C *of* $K_{n,n}$, $\chi_f''(K_{n,n} - C) = n^2/(n+1)$.

Theorem 5 (The least regular order with a rounding gap; certificate). *Every regular graph* G *on at most* 7 *vertices satisfies* $\lceil \chi_f''(G) \rceil = \chi''(G)$. *The Möbius ladder* M_8 *is cubic, satisfies* $\chi_f''(M_8) = 4 < 5 = \chi''(M_8)$, *and is the only connected cubic graph on* 8 *vertices of Type* 2. *Hence* 8 *is the least order of a regular graph* G *with* $\chi''(G) > \lceil \chi_f''(G) \rceil$.

Chen and Shan [9] formalized in Lean an *independent-seed peel certificate*: an independent set A and an ordering $u_1, \dots, u_t$ of $V \setminus A$ in which every u_i has fewer than $q - d_G(u_i)$ neighbors among $u_{i+1}, \dots, u_t$. Section 2 gives the definition in full. For $v \in V$ write $V_\Delta(G)$ for the set of vertices of degree $\Delta(G)$, and call $G[U]$ *strictly* f*-degenerate* if every nonempty $F \subseteq U$ has a vertex v with fewer than $f(v)$ neighbors in F.

Theorem 6 (Peel certificates; Lean). *Let* G *be a finite simple graph.*

(*i*) *For every independent set* A *and every* $q \in \mathbb{N}$, *a peel certificate for* (G, A, q) *exists if and only if* $G[V \setminus A]$ *is strictly* f*-degenerate for* $f(v) = q \dot{-} d_G(v)$, *where* $\dot{-}$ *is subtraction truncated at* 0.

(*ii*) *If a peel certificate for* $(G, A, \Delta(G)+1)$ *exists for some* A, *then* $G[V_\Delta(G)]$ *is bipartite.*

(*iii*) *If* G *has a closed walk of odd length all of whose vertices lie in* $V_\Delta(G)$, *then no set* A *has a peel certificate for* $(G, A, \Delta(G)+1)$.

Chen and Shan also introduced an *auxiliary class* $\mathcal{A}_D$ of pairs (H, J): a graph H with $\Delta(H) \leq D$, a center v_0 with $2 \leq d_H(v_0) \leq D$, and a set J of D edges, consisting of the edges at v_0 and a matching M, such that every vertex other than v_0 lies on exactly one edge of J. They proved that every member has a proper $(D+2)$-edge coloring in which the edges of J get distinct colors (*rainbow on* J) [9]; with $D+1$ colors the question is open. We prove the case $|M| = 1$ and record a barrier for recoloring by Kempe changes.

Theorem 7 (Rainbow colorings when $|M| = 1$; written proof). *Let* $D \geq 3$, *let* H *be a finite simple graph with* $\Delta(H) \leq D$, *let* v_0 *be a vertex with* $d_H(v_0) \leq D-1$, *and let* xy *be an edge of* H *with* $x, y \notin N_H[v_0]$. *Let* M' *be a maximal matching of* $H - \{v_0, x, y\}$ *and* $M_c = \{xy\} \cup M'$. *Then the vertices of degree* D *in* $H - M_c$ *are pairwise non-adjacent,* $H - M_c$ *has a proper* D*-edge coloring, and giving every edge of* M_c *the color* D *yields a proper* $(D+1)$*-edge coloring of* H *that is rainbow on* $\delta_H(v_0) \cup \{xy\}$. *In particular, every* $(H, J) \in \mathcal{A}_D$ *with* $|M| = 1$ *has a proper* $(D+1)$*-edge coloring rainbow on* J.

Theorem 8 (A Kempe distance two; Lean). *Let* H *be the graph on* $\{0,1,2,3,4\}$ *with the seven edges* 01, 02, 34, 13, 14, 23, 24, *and let* $J = \{01, 02, 34\}$. *Then* $(H, J) \in \mathcal{A}_3$ *with center* 0, *and the proper* 4*-edge coloring* ψ *that gives these edges the colors* 0, 1, 0, 2, 3, 3, 2 *in this order is not rainbow on* J. *No single Kempe change of* ψ, *on any union of Kempe chains of any two colors, is rainbow on* J, *while two successive Kempe changes give a proper* 4*-edge coloring that is rainbow on* J.

The last group concerns two colorings stronger than a total coloring. A total coloring is *equitable* if its color classes differ in size by at most one; $\chi_e''(G)$ is the least number of colors of an equitable total coloring [16]. The list total chromatic number $\chi_\ell''(G)$ is the least k such that every assignment of k-element lists to $V \sqcup E$ admits a proper total coloring from the lists [6]. Fix $\Delta \geq 3$. The graph G_Δ^* consists of $\Delta+1$ disjoint copies $B_0, \dots, B_\Delta$ of $K_{\Delta-1,\Delta}$: block B_i has the *private* vertices $p_{i,r}$ $(0 \leq r \leq \Delta-2)$ and the *terminals* $v_{i,j}$ $(0 \leq j \leq \Delta, j \neq i)$, every private vertex adjacent to every terminal of its block, together with the *cross edges* $v_{i,j}v_{j,i}$ for $i \neq j$. The graph H_Δ^* consists of two copies B_0, B_1 of $K_{\Delta-1,\Delta}$, with private vertices $p_{i,r}$ and terminals $t_{i,j}$ $(0 \leq j \leq \Delta-1)$, joined by the Δ cross edges $t_{0,j}t_{1,j}$.

Theorem 9 (The family G_Δ^*; Lean). *For every* $\Delta \geq 3$ *the graph* G_Δ^* *has* $(\Delta+1)(2\Delta-1)$ *vertices, is* Δ*-regular, connected, has no triangle and contains a* 4*-cycle. It is of Type* 1 *and* $\chi_f''(G_\Delta^*) = \Delta+1$. *In every proper* $(\Delta+1)$*-total coloring some color* c^* *is given to every private vertex and every cross edge; its class has* $(\Delta+1)(3\Delta-2)/2$ *elements, every other class has* $\Delta(\Delta+1)$ *elements, and so the class of* c^* *is larger than each other class by at least* 2. *Consequently* G_Δ^* *has no equitable* $(\Delta+1)$*-total coloring, and it has a list assignment with lists of size* $\Delta+1$ *that admits no proper total coloring from the lists.*

Removing colors from the lists keeps a list assignment without a coloring from the lists, so for every $k \leq \Delta+1$ some assignment with lists of size k admits no coloring; that is, $\chi_\ell''(G_\Delta^*) \geq \Delta+2$. Unlike H_Δ^* below, G_Δ^* is not bipartite: the nine vertices $v_{0,1}, p_{0,0}, v_{0,2}, v_{2,0}, p_{2,0}, v_{2,1}, v_{1,2}, p_{1,0}, v_{1,0}$ form a cycle of length 9 (`recompute.py`, block 7, checks it for $3 \leq \Delta \leq 20$).

Theorem 10 (Equitable total colorings of G_Δ^*; written proof). *For every* $\Delta \geq 3$, $\chi_e''(G_\Delta^*) = \Delta+2$. *For odd* Δ *there is an equitable* $(\Delta+2)$*-total coloring with all classes of size* $(\Delta+1)(2\Delta-1)/2$; *for even* $\Delta = 2q$ *there is one with* $q+1$ *classes of size* $4q^2+q$ *and* $q+1$ *classes of size* $4q^2+q-1$.

Theorem 11 (The bipartite family H_Δ^*; written proof). *For every* $\Delta \geq 3$ *the graph* H_Δ^* *is bipartite and* Δ*-regular, with* $4\Delta-2$ *vertices and* $\Delta(2\Delta-1)$ *edges, and contains a* 4*-cycle. It is of Type* 1, *and in every proper* $(\Delta+1)$*-total coloring one class has* $3\Delta-2$ *elements and each of the other* Δ *classes has* 2Δ. *Further* $\chi_e''(H_3^*) = 4$,

$\chi''_e(H^*_\Delta) = \Delta + 2$ *for* $\Delta \geq 4$, *and* $\chi''_\ell(H^*_\Delta) = \Delta + 2$ *for* $\Delta \geq 5$. *For* $\Delta = 4$, *every list assignment that gives all elements of* B_0 *one list of* 5 *colors, all elements of* B_1 *one list of* 5 *colors, and each cross edge any list of* 5 *colors admits a proper total coloring from the lists.*

Remark 12 (What is not proved). None of these statements bears on the Total Coloring Conjecture for general graphs. Theorem 7 handles $|M| = 1$ only; for $|M| \geq 2$ the $(D+1)$-color question of Chen and Shan is open here, apart from the exhaustive checks for $D \leq 5$ of Section 4. Whether $\chi''_\ell(G^*_\Delta) = \Delta+2$ is not decided here (Galvin's theorem, which gives the upper bound for H^*_Δ, needs a bipartite graph). For $K_{n,n}$ minus a 2-factor that contains a 4-cycle, Proposition 18 settles only the 2-factors $C_4 \sqcup C_{2n-4}$ with $n = 4$, $n = 6$ or n odd.

Earlier work. Several of the values are known. The total chromatic numbers of K_n and $K_{n,n}$ are due to Behzad, Chartrand and Cooper [4]; that Möbius ladders are of Type 2 is credited to Chetwynd and Hilton [10] by Alves Junior and Sasaki [23] [to check]; Khennoufa and Togni [26] give upper bounds on χ''_f of cubic circulant graphs, a class that contains every M_{2k}, and we have read only their abstract [to check]. The graph G^*_3 is the 20-vertex cubic graph with which Noel [34] refuted the List Total Coloring Conjecture of Borodin, Kostochka and Woodall [6], and by comparing figures we believe it is the graph R of Dantas et al. [12], drawn by Adauto et al. [2] [to check]; Theorems 9 and 10 give a family for every $\Delta \geq 3$ that contains it. Section 6 says what was searched and read.

An illustration. The box below evaluates the formula behind Theorem 1(ii). Write $k = 3t+s$ with $s \in \{1,2,3\}$. When $s \leq t$, a weighting of total matchings gives $\chi''_f(M_{2k}) = 4$; otherwise $\chi''_f(M_{2k}) = 5k/\alpha_T(M_{2k})$ with $\alpha_T(M_{2k}) = 4t + s$. The box lists the $k < 40$ with value above 4, and then, for $\Delta = 3, 4, 5$, the order of G^*_Δ and the spread $(\Delta+1)(\Delta-2)/2$ forced in Theorem 9.

Concept in Action: where the Möbius ladder value exceeds 4

```
from fractions import Fraction as F
exc = []
for k in range(2, 40):                      # Moebius ladder M_2k
    t, s = (k - 1) // 3, (k - 1) % 3 + 1    # k = 3t + s, s in {1,2,3}
    chi_f = F(4) if s <= t else F(5 * k, 4 * t + s)
    if chi_f > 4:
        exc.append((k, str(chi_f)))
print(exc)
print([(d, (d+1)*(2*d-1), (d+1)*(d-2)//2) for d in (3, 4, 5)])
```

It prints `[(2, '5'), (3, '5'), (5, '25/6'), (6, '30/7'), (9, '45/11')]` and `[(3, 20, 2), (4, 35, 5), (5, 54, 9)]`.

Roadmap. Section 2 fixes definitions and their Lean counterparts. Section 3 proves the results: general bounds (§3.1), dense bipartite graphs (§3.2), Möbius ladders and prisms (§3.3), small orders (§3.4), peel certificates and the auxiliary class (§3.5), and the families G^*_Δ and H^*_Δ (§3.6). Section 4 describes the computations, Section 5 the trust boundary, Section 6 prior work and open questions.

2. DEFINITIONS, NOTATION, AND SETUP

All graphs are finite and simple. For a graph $G = (V, E)$, $d_G(v)$ is the degree of v, $\Delta(G)$ the largest degree, $N_G(v)$ the set of neighbors, $N_G[v] = N_G(v) \cup \{v\}$, and $\delta_G(v)$ the set of edges at v. In Lean a graph is a `SimpleGraph V` on a finite type `V`, and the elements of $T(G)$ form the type `TotalElem G`, defined as $V \sqcup E$ (`V ⊕ G.edgeSet`); adjacency in $T(G)$ is `totalAdj` as described in Section 1.

Total colorings. A *proper k-total coloring* is a map $f : V \sqcup E \to \{0, \dots, k-1\}$ with $f(x) \neq f(y)$ whenever x, y are adjacent in $T(G)$ (`IsProperTotalColoring`). Its *color classes* are the sets $f^{-1}(c)$; each is a total matching. G is of Type 1 if it has a proper $(\Delta+1)$-total coloring (`IsType1`) and of Type 2 if it has none but has a proper $(\Delta+2)$-total coloring (`IsType2`). The total matching $S \sqcup M$ is *star-saturated* if $|S|+2|M| = |V|$, that is, if it meets every closed star $\{v\} \cup \delta_G(v)$. The deficiency of G is $\mathrm{def}(G) = \sum_{v \in V}(\Delta(G) - d_G(v))$. Following Chetwynd and Hilton [10], G is *conformable* if it has a proper vertex coloring with $\Delta(G)+1$ colors (classes may be empty) in which the number of classes whose size has parity different from $|V|$ is at most $\mathrm{def}(G)$ (`IsConformable`).

Fractional total colorings. A *fractional total coloring* assigns a rational weight $w_I \geq 0$ to each subset $I \subseteq V \sqcup E$, with $w_I > 0$ only on total matchings, such that every element is covered with total weight at least 1; its weight is $\sum_I w_I$ (`FractionalTotalColoring`). The least weight is $\chi''_f(G)$; we write $\chi''_f(G) = q$ when some fractional total coloring has weight q and none has smaller weight (`FracTotalEq`). A *dual weighting* $y : V \sqcup E \to \mathbb{Q}_{\geq 0}$ has $\sum_{x \in I} y_x \leq 1$ on every total matching I, and weak duality gives $\chi''_f(G) \geq \sum_x y_x$. If G has a vertex, the weighting $y \equiv 1/\alpha_T(G)$ gives $\chi''_f(G) \geq (|V| + |E|)/\alpha_T(G)$. Since a proper k-total coloring is a fractional one of weight k, $\chi''_f(G) \leq \chi''(G)$, and $\alpha_T(G)$ is `alphaTotal`, the largest size of a total matching.

Equitable and list colorings. A proper k-total coloring is *equitable* if any two classes differ in size by at most 1 (`IsEquitableTotalColoring`). A list assignment gives each element x a finite set $L(x)$ of colors; G is *k-list-total colorable* if for every assignment with $|L(x)| = k$ for all x there is a proper total coloring f with $f(x) \in L(x)$ (`HasListTotalColoring`). If G is k-list-total colorable it is also $(k+1)$-list-total colorable (shrink each list), so $\chi''_\ell(G) \geq \Delta + 2$ follows from a single assignment of $(\Delta+1)$-element lists without a coloring.

Graph families. $K_{n,n}$ has sides $A = \{u_0, \dots, u_{n-1}\}$ and $B = \{v_0, \dots, v_{n-1}\}$ (`CompleteBipartite n`); the crown $Cr_n = K_{n,n} - nK_2$ has $u_i v_j$ for $i \neq j$ (`CrownGraph n`). A 2-factor of $K_{n,n}$ is a spanning 2-regular subgraph; it is a disjoint union of even cycles. M_{2k} is `MobiusLadder k`; for $k = 2$ it is K_4 and for $k = 3$ it is $K_{3,3}$. The prism Y_n has vertices u_j, v_j ($j \in \mathbb{Z}_n$), *rim edges* $u_j u_{j+1}$ and $v_j v_{j+1}$, and *spokes* $u_j v_j$; Y_5 is `PrismL10`. The graphs G^*_Δ (`GDeltaStar`, with vertex type `GDeltaVert`) and H^*_Δ are defined before Theorem 9; the cross edge $v_{i,j} v_{j,i}$ of G^*_Δ is `crossEdge i j`.

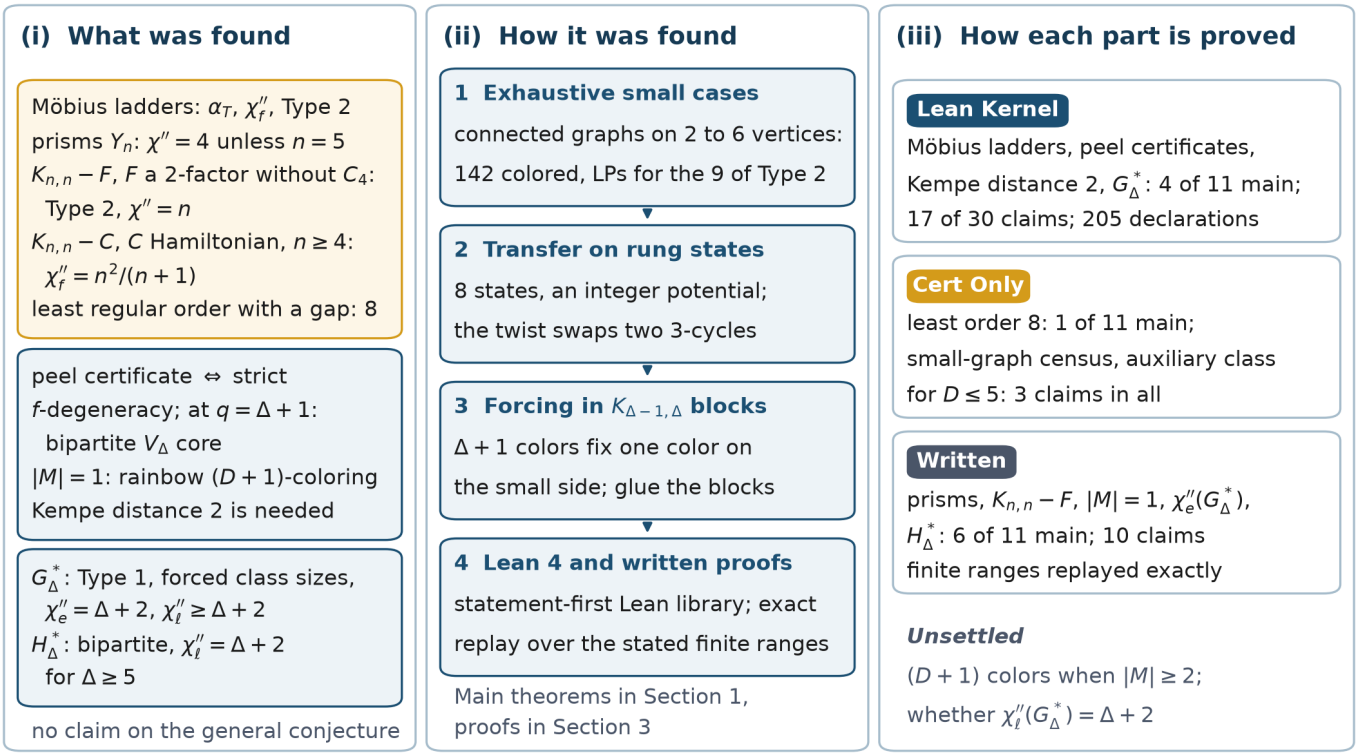


Figure 1. Discovery and proof architecture. Column (i) lists the results, column (ii) how they were found, column (iii) the proof tier of each result. The tiers and counts are those of the claim ledger of Section 5; the declaration count is read from the axiom receipt of the Lean library.

Peel certificates. Let A be an independent set and $q \in \mathbb{N}$. Following Chen and Shan [9], a *peel certificate* for (G, A, q) (in Lean `IndependentSeedPeelCertificate G A q`) is a repetition-free list $u_1, \dots, u_t$ of exactly the vertices outside A such that, for every i, the number of $j > i$ with $u_i u_j \in E$ is less than $q \dot{-} d_G(u_i)$.

The auxiliary class. A pair (H, J), with J a set of edges of H, is in $\mathcal{A}_D$ with center v_0 (`IsAuxiliaryClassMember`) if $\Delta(H) \leq D$, $|J| = D$, $2 \leq d_H(v_0) \leq D$, every edge at v_0 is in J, and every vertex $v \neq v_0$ lies on exactly one edge of J. Then $M = J \setminus \delta_H(v_0)$ (`auxMatching`) is a matching. A proper k-edge coloring c of H (`IsProperEdgeColoring`) is *rainbow on* J if the edges of J get distinct colors (`IsRainbowOn`). For colors a, b, a set S of edges is *Kempe-closed* if every edge of S has color a or b and every edge of color a or b that shares an endpoint with an edge of S is in S (`IsKempeClosed`); a *Kempe change* swaps a and b on S (`kempeFlip`). A Kempe-closed set is a union of *Kempe chains*, the components of the subgraph of edges colored a or b, and a Kempe change keeps the coloring proper.

3. PROOFS OF THE MAIN RESULTS

Each Lean statement is cited by its declaration name; the declarations gathered for this paper are in `VizingTotal/PaperDraft1.lean` of the finding (Section 7). Five arguments below are written for this draft and differ from the ones recorded with the finding: the proofs of Theorem 4 for $n \geq 13$, of Theorem 10 for even Δ and of Proposition 18 for odd n, the fan argument of Lemma 26 and Theorem 7, and the junction check of Theorem 2. The argument for the last clause of Theorem 11 is also ours. Section 7 lists them.

General bounds.

Proposition 13 (Cliques of the total graph; Lean). *Let G have at least one vertex. If $\Delta(G) = 1$ then the largest clique of $T(G)$ has 3 elements; otherwise it has $\Delta(G)+1$. Every fractional total coloring of G has weight at least the clique number of $T(G)$.*

Proof. This is `paper_clique`. A clique Q of $T(G)$ with at least three vertices has no edge, since an edge has two endpoints; with two vertices it has at most one edge; with one vertex u it lies in $\{u\} \cup \delta_G(u)$; with no vertex its edges pairwise meet, so they share a vertex or form a triangle. A closed star gives $\Delta + 1$, and for $\Delta = 1$ an edge with its endpoints gives 3. The last statement is weak duality with y the indicator of a largest clique. □

Proposition 14 (Deficiency bound; Lean). *Let G have $n \geq 1$ vertices and m edges. Every fractional total coloring of G has weight at least $1 + 2m/n$ and at least $\Delta(G) + 1 - \mathrm{def}(G)/n$. If G is r-regular, every fractional total coloring has weight at least $r + 1$, and it has weight exactly $r+1$ if and only if it covers every element with weight exactly 1 and every total matching of positive weight is star-saturated.*

The two lower bounds are equal, since $\mathrm{def}(G) + 2m = n\Delta(G)$ (`deficiency_add_two_mul_edges`).

Proof. This is `paper_deficiency_dual`. A total matching $S \sqcup M$ has $|S| + 2|M| \leq n$, because S and the vertices covered by M are disjoint. So $y_v = 1/n$ on vertices and $y_e = 2/n$ on edges is a dual weighting of value $(n + 2m)/n$, and $2m =$

$n\Delta - \mathrm{def}(G)$. For an r-regular graph the value is $r+1$. If a fractional total coloring w has weight $r+1$, then $r+1 = \sum_x y_x \le \sum_x y_x \sum_{I \ni x} w_I = \sum_I w_I(|S_I| + 2|M_I|)/n \le r+1$, so both inequalities are equalities; as every $y_x > 0$, every element is covered exactly once and every I with $w_I > 0$ is star-saturated. The converse is the same computation read backwards. □

Proposition 15 (Chetwynd and Hilton; Lean). *Every graph of Type* 1 *is conformable.*

Proof. This is `paper_conformable`; the statement is due to Chetwynd and Hilton [10]. In a proper $(\Delta+1)$-total coloring, the class $S_c \sqcup M_c$ of a color c leaves $d_c = |V| - |S_c| - 2|M_c| \ge 0$ vertices whose closed star misses c, and $|S_c| \equiv |V| - d_c \pmod 2$. A vertex v misses exactly $\Delta - d_G(v)$ colors on its closed star, so $\sum_c d_c = \mathrm{def}(G)$, and at most $\mathrm{def}(G)$ classes have d_c odd. The vertex coloring is the restriction to V. □

Dense bipartite graphs.

Proposition 16 (Regular bipartite graphs; Lean). *Let $0 \le k \le n$ and let G be an $(n-k)$-regular bipartite graph with sides of size n. Every total matching of G has at most $n+k$ elements, and if $n \ge 1$ every fractional total coloring has weight at least $n(n-k+2)/(n+k)$. For $n \ge 1$, $K_{n,n}$ is of Type 2 and $\chi''_f(K_{n,n}) = n+2$; the crown Cr_2 is of Type 2 and $\chi''_f(Cr_2) = 3$; for $n \ge 3$, Cr_n is of Type 1 and $\chi''_f(Cr_n) = n$.*

Proof. This is `paper_bipartite`. Let $S \sqcup M$ be a total matching with a vertices of S in A and b in B. If $a, b \ge 1$, a vertex of $S \cap A$ has $n-k$ neighbors in B, none in S, so $b \le k$; the matching covers at most $n-a$ vertices of A, so the size is at most $a+b+(n-a) \le n+k$. If $a=0$ or $b=0$ the size is at most n. The uniform weighting $1/(n+k)$ on the $2n+n(n-k)$ elements gives the fractional bound. For $K_{n,n}$ $(k=0)$ it is $n+2$, attained by an n-edge coloring (König [28]) and one more color on each side; $K_{n,n}$ has no proper $(n+1)$-total coloring because the first part with $k=0$ gives $\chi''_f(K_{n,n}) \ge n+2$ (the values are classical [4]). Cr_2 is two disjoint edges, so Proposition 13 gives 3. For $n \ge 3$ the $(n-1)$-regular Cr_n has $\chi''_f \ge n$ by Proposition 14, and an idempotent Latin square L of order n (with $L_{ii} = i$, which exists for $n \ne 2$) gives the n-total coloring $u_i, v_i \mapsto i$, $u_i v_j \mapsto L_{ij}$; the Lean proof builds L for both parities. □

Proposition 17 (Complete graphs; Lean). *For every $r \ge 0$ the complete graph K_{2r+1} is of Type 1, and for every $r \ge 1$ the complete graph K_{2r} is of Type 2.*

Proof. This is `paper_complete`, the classical result of Behzad, Chartrand and Cooper [4]. □

Proof of Theorem 3. G is $(n-2)$-regular and bipartite, so by König's theorem [28] its edges have a proper $(n-2)$-coloring; two more colors, one for each side, give a proper n-total coloring. Suppose G had a proper $(n-1)$-total coloring. The n^2 elements of $T(G)$ fall into $n-1$ classes, and summing $|S_c| + 2|M_c|$ over the classes gives $2n + 2n(n-2) = (n-1)\cdot 2n$; since each class has $|S_c| + 2|M_c| \le 2n$, every class is star-saturated. A matching covers equally many vertices of A and B, so $|S_c \cap A| = n - |M_c| = |S_c \cap B|$ for every c. The n vertices of A lie in $n-1$ classes, so some class has two vertices $u, u' \in A$ and then also two vertices $w, w' \in B$. These four vertices are pairwise non-adjacent in G, so $uw, uw', u'w, u'w'$ are edges of F and form a 4-cycle in F, a contradiction. Hence $\chi''(G) = n$.

The same argument bounds total matchings: if $S \sqcup M$ has two vertices on each side, F has a 4-cycle; so one side, say B, contains at most one vertex of S, and the size is at most $|S \cap A| + 1 + (n - |S \cap A|) = n+1$. The uniform weighting $1/(n+1)$ on the n^2 elements gives $\chi''_f(G) \ge n^2/(n+1)$. □

Proof of Theorem 4. All Hamiltonian cycles of $K_{n,n}$ are equivalent under automorphisms, so take C with the edges $u_i v_i$ and $u_i v_{i+1}$ (indices in $\mathbb{Z}_n$). Then $G = K_{n,n} - C$ has the edges $u_i v_{i+d}$ with *shift* $d \in D = \{2, \dots, n-1\}$; each shift class has n edges. The lower bound is Theorem 3, since C has no 4-cycle for $n \ge 3$. For $n = 4$, G is an 8-cycle and the exact linear program over all total matchings gives $16/5$ (`recompute.py`, block 5). Let $n \ge 5$. Two families of total matchings of size $n+1$ are used; the proof that follows differs from the argument recorded with the finding, which we could not check (Section 7).

For $d \in D$, the matching J_d has $S = \{u_0, v_0\}$, the $n-2$ shift-d edges that avoid S, and the edge $u_{-d}v_d$, of shift $2d$, which covers the two remaining vertices. The matching I_d has $S = \{u_0, v_0, v_1\}$, the $n-3$ shift-d edges that avoid S, and one edge at the remaining vertex v_d of B: $u_{-d}v_d$ (shift $2d$), or $u_{1-d}v_d$ (shift $2d-1$) if $2d \in \{0,1\}$ modulo n. In both cases S is independent, because the pairs u_0v_0 and u_0v_1 lie on C. An extra edge fails to exist only for one *special* shift d^*: $d^* = (n+1)/2$ for J and I when n is odd, and $d^* = n/2$ for J when n is even. There we drop the shift-d^* edge at u_a and add two edges: for odd n, $a = 1$ and the edges $u_{-d^*}v_{1+d^*}$ (shift 2) and $u_1 v_{d^*}$ (shift d^*-1); for even $n \ge 8$, $a = 2$ and the edges $u_{d^*}v_{2+d^*}$ (shift 2) and $u_2 v_{d^*}$ (shift d^*-2); for $n = 6$, $a = 4$ and the edges $u_3 v_1$ and $u_4 v_3$ (shifts 4 and 5). In each case the result is again a total matching of size $n+1$.

Let Γ be the group of order $2n$ generated by the maps $u_i \mapsto u_{i+1}, v_j \mapsto v_{j+1}$ and $u_i \mapsto v_{-i}, v_j \mapsto u_{-j}$; both maps preserve G and the shift of every edge, and Γ is transitive on V and on each shift class. Spreading a total matching of weight w evenly over its $2n$ images covers every vertex with weight $w|S|/(2n)$ and every edge of shift e with weight $w\,P_e/n$, where P_e counts its shift-e edges. Give every I_d the weight $\bar{x} = 2n/((n+1)(n-2))$ and every J_d a weight z_d. Exact covering of the edges is the system

$$\sum_{d \in D} z_d\, P^J_{d,e} = c_e := n - \bar{x} \sum_{d \in D} P^I_{d,e} \qquad (e \in D), \tag{1}$$

where $P^J_{d,e}, P^I_{d,e}$ count the shift-e edges of J_d, I_d. Summing (1) over e, with $n-1$ edges in each J_d and $n-2$ in each I_d, gives $Z = \sum_d z_d = n(n-2)/(n+1)$, and with $X = (n-2)\bar{x} = 2n/(n+1)$ the vertex coverage is $(2Z+3X)/(2n) = 1$. The total weight is $X + Z = n^2/(n+1)$. It remains to see that (1) has a solution with $z \ge 0$.

By construction $P^J_{e,e} \ge n-3$, $P^I_{e,e} \le n-3$, and $P^I_{e,e} \ge n-4$. Off the diagonal, the extra edges of J_d have shift $2d$, and $d \mapsto 2d$ takes each value at most twice, plus at most two edges of the special J_{d^*}; so $\sum_{d \ne e} P^J_{d,e} \le 4$, and similarly $\sum_{d \ne e} P^I_{d,e} \le 6$.

Hence $c_e \le n$, and $c_e \ge n-\bar{x}(n+3) \ge n-3$ once $n^2-9n-6 \ge 0$, that is $n \ge 10$. Write the matrix of (1) as a diagonal part with entries at least $n-3$ plus a nonnegative part with row sums at most 4. For $n \ge 8$ it is strictly diagonally dominant, so a solution exists, and $\|z\|_\infty \le (\max_e c_e + 4\|z\|_\infty)/(n-3)$ gives $\|z\|_\infty \le n/(n-7)$. Then every $z_e \ge (n-3-4n/(n-7))/P^J_{e,e}$, which is positive when $n^2-14n+21>0$, that is for $n \ge 13$. For $5 \le n \le 40$, `recompute.py` (block 5) builds every I_d and J_d, solves (1) in exact rational arithmetic, finds $z \ge 0$, and for $5 \le n \le 10$ also spreads the weighted matchings over Γ and checks that every vertex and edge is covered with weight exactly 1. This proves $\chi''_f(G) \le n^2/(n+1)$ for every $n \ge 4$. □

Proposition 18 ($K_{n,n}$ minus $C_4 \sqcup C_{2n-4}$; written proof and exact search). *Let F be a 2-factor of $K_{n,n}$ isomorphic to $C_4 \sqcup C_{2n-4}$ and $G = K_{n,n} - F$. For $n = 4$, G is of Type 2. For $n = 6$ and for every odd $n \ge 5$, G is of Type 1.*

Proof. For $n = 4$, G is two disjoint 4-cycles, and $\chi''(C_4) = 4$ (Proposition 16 with $n = 2$). For $n = 6$, an exact search finds a proper 5-total coloring (`recompute.py`, block 5). Let $n \ge 5$ be odd, $m = n-2 = 2q-1$, and index the vertices so that F is the 4-cycle on u_0, u_1, v_0, v_1 together with the cycle formed by the pairs $u_{2+a}v_{2+a}$ and $u_{2+a}v_{2+a+1}$ ($a \in \mathbb{Z}_m$, indices of u_{2+a}, v_{2+a} taken in $\mathbb{Z}_m$). The construction below is ours. Let $L_{a,b} = 1 + (q(a+b) \bmod m)$ for $a, b \in \mathbb{Z}_m$. As $2q \equiv 1$, L is a Latin square on $\{1, \dots, m\}$ with $L_{a,a} = a+1$. Color u_0, u_1, v_0, v_1 with 0 and u_{2+a}, v_{2+a} with $a+1$; color $u_0 v_{2+b}$ with $L_{b-1,b}$, $u_1 v_{2+b}$ with $L_{b+1,b}$, $u_{2+a}v_0$ with $L_{a,a+1}$, $u_{2+a}v_1$ with $L_{a,a-1}$, $u_{2+a}v_{2+a-1}$ with 0, and every other edge $u_{2+a}v_{2+b}$ with $L_{a,b}$. At u_{2+a} the edge colors are the entries $L_{a,b}$, $b \ne a$, of row a (the entry $L_{a,a+1}$ moved to v_0, the entry $L_{a,a-1}$ to v_1) together with 0; they are distinct and avoid $a+1$. Column b gives the same at v_{2+b}. At u_0 the colors $L_{b-1,b} = 1 + (b-q \bmod m)$ run over $\{1, \dots, m\}$ once, and likewise at u_1, v_0, v_1. Adjacent vertices get different colors, since $u_{2+a}v_{2+a}$ is not an edge. So this is a proper $(n-1)$-total coloring; `recompute.py` checks it for $5 \le n \le 21$. □

Möbius ladders and prisms. Write the vertices of M_{2k} as rungs $(u_j, v_j) = (j, j+k)$, $j \in \{0, \dots, k-1\}$; consecutive rungs are joined by $u_j u_{j+1}$ and $v_j v_{j+1}$, and the last rung is joined to the first with the rails exchanged, $u_{k-1}v_0$ and $v_{k-1}u_0$. A total matching I is read rung by rung. The state of rung j records, for each of u_j and v_j, whether it is in S (value 1), covered by an edge of M that arrives from the previous rung or by the spoke (value 2), or neither (value 0); $(1,1)$ is excluded, so there are 8 states. A *step* $(\sigma, \sigma'; e_u, e_v, i_u, i_v, s)$ from the state $\sigma = (a,b)$ of rung j to the state $\sigma' = (a', b')$ of rung $j+1$ carries five bits, which say whether $u_j u_{j+1} \in M$, $v_j v_{j+1} \in M$, $u_{j+1} \in S$, $v_{j+1} \in S$ and $u_{j+1}v_{j+1} \in M$. It is *legal* when: e_u only if $a = 0$; e_v only if $b = 0$; not both i_u and i_v; s only if the other four bits are 0; i_u only if $e_u = 0$ and $a \ne 1$; i_v only if $e_v = 0$ and $b \ne 1$; $a' = 1$ if i_u, else $a' = 2$ if e_u or s, else $a' = 0$; and b' is given by the same rule with i_v, e_v. The step adds $\delta = e_u + e_v + i_u + i_v + s$ elements. Let ϕ be the potential with $\phi = 0$ on $(0,0), (0,2), (2,0)$, $\phi = 1$ on $(0,1), (1,0)$ and $\phi = 2$ on $(1,2), (2,1), (2,2)$, and let a step that adds δ elements have reduced weight $\tilde{w} = 3\delta - 4 + \phi(\text{source}) - \phi(\text{target})$.

Lemma 19 (The transition table; Lean). *There are exactly 39 legal steps; they join exactly 38 distinct ordered pairs (σ, σ') of states; the legal steps of reduced weight 0 join exactly 11 distinct ordered pairs; and every legal step has reduced weight at most 0.*

Proof. This is `paper_chiral_table`, checked by evaluation of the finite table (`allLegalTransitions`). □

Proof of Theorem 1. This is `paper_mobius`; we outline the argument. Summing the reduced weights around the closed walk of states of a total matching I gives $3|I| - 4k$, because the potential telescopes: the exchange of rails at the last step does not change ϕ. By Lemma 19, $3|I| \le 4k$, with equality only if every step has reduced weight 0. No step of weight 0 enters $(0,0)$, and no step of weight 0 from a state other than $(0,0)$ enters $(2,2)$; so a closed walk of weight-0 steps avoids both. Every step of weight 0 keeps each of the classes $C_+ = \{(0,1), (2,0), (1,2)\}$ and $C_- = \{(1,0), (0,2), (2,1)\}$, and the exchange of rails swaps C_+ and C_-. Around M_{2k} the walk returns to the exchanged state of rung 0, so it cannot consist of weight-0 steps, and $3|I| \le 4k-1$. Conversely, write $k = 3t+s$ with $s \in \{1,2,3\}$: t blocks of three rungs, each with $u_a, v_{a+2} \in S$ and $v_a v_{a+1}, u_{a+1}u_{a+2} \in M$, followed by s spokes, form a total matching of size $4t+s = \lfloor (4k-1)/3 \rfloor$. This gives (i).

For (ii), weak duality with the uniform weighting gives $\chi''_f(M_{2k}) \ge 5k/\alpha_T(M_{2k})$, and $\chi''_f \ge 4$ always. The dihedral symmetry acts transitively on the $2k$ vertices, the $2k$ rim edges and the k spokes. If $1 \le s \le t$, the matching above spread over its images with total weight k/t, plus the matching of all k spokes with weight $(t-s)/t$, covers every element exactly once with weight 4. The condition $s \le t$ holds exactly when $k = 4a+3b$ for some integers $a \ge 1$ and $b \ge 0$: if $s \le t$ take $a = s$ and $b = t-s$; conversely $a \equiv s \pmod 3$ and $1 \le s \le 3$ give $a = s+3j$ with $j \ge 0$, so $t - s = b + 4j \ge 0$. This is (iii): it fails for $k = 2, 3$, and for $k \ge 4$ it fails exactly when $k-4$ is one of $1, 2, 5$, the gaps of the semigroup generated by 3 and 4. For $k \in \{2, 3, 5, 6, 9\}$ the Lean proof exhibits weightings of weight $5k/\alpha_T(M_{2k})$. By profile $(|S|, \text{rim edges}, \text{spokes})$, spread over the dihedral images with the weights after the colons, they are $(0,0,2)$: 1, $(1,1,0)$: 4 for $k = 2$; $(0,0,3)$: $\frac{1}{2}$, $(0,2,1)$: $\frac{3}{2}$, $(2,1,0)$: 3 for $k = 3$; $(2,2,2)$: $\frac{5}{2}$, $(3,3,0)$: $\frac{5}{3}$ for $k = 5$; $(2,2,3)$: $\frac{6}{7}$, $(3,3,1)$: $\frac{24}{7}$ for $k = 6$; and $(4,4,3)$: $\frac{27}{11}$, $(5,5,1)$: $\frac{18}{11}$ for $k = 9$ (`recompute.py`, block 2, which checks $2 \le k \le 10$). For (iv), the Lean proof reads a 4-total coloring at the boundary between consecutive rungs as the colors of u_j, v_j, $u_j u_{j+1}$, $v_j v_{j+1}$ and sorts these quadruples into five classes: two classes L and R, each kept by every step and swapped by the exchange of rails; a class T with no possible next step; a class B whose next step is in T; and a class A whose next step is in B, L or R. Around the ladder some boundary is therefore in L or R, and the class is kept until the exchange of rails swaps it, a contradiction. The Lean proof also gives an explicit 5-total coloring. □

Table 1 lists the values; they are recomputed in exact arithmetic by `recompute.py` (blocks 2 and 3), which also finds $\alpha_T(M_{2k})$ by exhaustive search for $2 \le k \le 5$.

Table 1. The Möbius ladders M_{2k} for $2 \le k \le 10$ (Theorem 1).

k	$\alpha_T(M_{2k})$	$5k/\alpha_T(M_{2k})$	$\chi''_f(M_{2k})$
2	2	5	5
3	3	5	5
4	5	4	4
5	6	25/6	25/6
6	7	30/7	30/7
7	9	35/9	4
8	10	4	4
9	11	45/11	45/11
10	13	50/13	4

Proof of Theorem 2. The rung encoding applies to Y_n with rungs (u_j, v_j), $j \in \mathbb{Z}_n$, and no exchange of rails, so the telescoping sum and Lemma 19 give $3|I| \le 4n$. For the lower bound write $n = 3t + s$ with $s \in \{0, 1, 2\}$ and take t blocks of three rungs as above followed by s spokes; consecutive blocks are compatible because a block ends with $v_{a+2} \in S$ and u_{a+2} covered by M and begins with $u_a \in S$ and v_a covered by M. This total matching has $4t + s = \lfloor 4n/3 \rfloor$ elements (`recompute.py`, block 6, checks it for $3 \le n \le 60$, and block 3 finds $\alpha_T(Y_n)$ by exhaustive search for $3 \le n \le 6$).

Now let $n \ne 5$; then $n = 3a + 4b$ with $b \in \{0, 1, 2\}$ the remainder of n modulo 3 and $a \ge 0$. Describe a 4-total coloring rung by rung by the colors of $(u_j, v_j, u_jv_j, u_ju_{j+1}, v_jv_{j+1})$, and let

$$T_3 = [(0,1,3,2,0), (1,2,3,0,1), (2,0,3,1,2)],$$
$$T_4 = [(0,1,3,2,0), (3,2,1,0,3), (1,0,2,3,1), (2,3,0,1,2)].$$

Every constraint of a total coloring of Y_n involves two consecutive rungs: the closed star of u_{j+1} is u_{j+1}, u_ju_{j+1}, $u_{j+1}u_{j+2}$ and the spoke at $j+1$, and the adjacent vertex pairs are u_ju_{j+1}, v_jv_{j+1} and u_jv_j. So a cyclic sequence of rung colorings is proper exactly when each pair of consecutive rungs is. Inside T_3 and T_4 and at the junctions T_3T_3, T_3T_4, T_4T_3, T_4T_4 every pair is proper (checked by hand and in `recompute.py`, block 6, the junction check being the part written for this paper). Hence a copies of T_3 followed by b copies of T_4 give a proper 4-total coloring of Y_n, and $4 = \Delta + 1 \le \chi''_f(Y_n) \le \chi''(Y_n) \le 4$.

For $n = 5$, Proposition 21 gives $\chi''_f(Y_5) = 25/6$ and that Y_5 is of Type 2 (Lean); a 5-total coloring is $u_j \mapsto j$, $v_j \mapsto j + 2$, $u_jv_j \mapsto j + 1$, $u_ju_{j+1} \mapsto j + 3$, $v_jv_{j+1} \mapsto j + 4$ modulo 5. □

Small orders.

Proposition 20 (Census of small graphs; certificate). *Among the* 207 *unlabeled graphs on* 2 *to* 6 *vertices,* 142 *are connected; of these* 133 *are of Type* 1 *and* 9 *of Type* 2*, and each of the* 9 *satisfies* $\lceil \chi''_f \rceil = \chi''$*: six have* $\chi''_f > \Delta+1$ *and are not conformable* (K_2, K_4, C_5, $K_{3,3}$, $K_6 - e$, K_6*, with* $\chi''_f = 3, 5, 10/3, 5, 19/3, 7$*), and three have* $\chi''_f > \Delta+1$ *and are conformable* (C_4, $K_{3,3} - e$ *and* $K_6 - (K_{1,3} \sqcup K_2)$*, with* $\chi''_f = 4, 13/3, 16/3$*). Of the* 5 *connected cubic graphs on* 8 *vertices exactly one,* M_8*, is of Type* 2*; it is conformable and* $\chi''_f(M_8) = 4$*. The Petersen graph, a cubic graph* B_{10} *on* 10 *vertices with a bridge, and the Tietze graph have chromatic index* 4 *and* $\chi'' = 4$*, and* Y_5 *has* $\chi'' = 5$*.*

Proof. `checks/census.py` enumerates the graphs, computes χ'' by an exact coloring search, decides conformability, and computes χ''_f by an exact linear program over the maximal total matchings whose primal and dual certificates are reverified in rational arithmetic. Its output `checks/census.out` ends with `CENSUS CHECK: PASS`; the unlabeled counts match the sequences A000088 and A001349 of the OEIS [35]. □

Hence $\lceil \chi''_f(G) \rceil = \chi''(G)$ for every graph G with at most 6 vertices, connected or not. Both numbers are the largest of their values on the components of G, and rounding up commutes with a finite maximum. The empty graph has $\chi''_f = \chi'' = 0$ and a single vertex $\chi''_f = \chi'' = 1$; a connected graph of Type 1 has $\Delta + 1 \le \chi''_f \le \chi'' = \Delta + 1$ by Proposition 13; and the 9 connected graphs of Type 2 are in Proposition 20.

Proposition 21 (Small regular witnesses; Lean). *M_8 has* 12 *edges, is cubic and conformable,* $\alpha_T(M_8) = 5$*,* $\chi''_f(M_8) = 4$*, and M_8 is of Type* 2*. Y_5 has* 15 *edges, is cubic and conformable,* $\alpha_T(Y_5) = 6$*,* $\chi''_f(Y_5) = 25/6$*, and Y_5 is of Type* 2*. Every regular graph on* 7 *vertices has degree* 0*,* 2*,* 4 *or* 6*. The edgeless graph on* 7 *vertices and K_7 are of Type* 1*. Every fractional total coloring has weight at least* 7/2 *for C_7, at least* 4 *for $C_3 \sqcup C_4$, and at least* 21/4 *for the complement $\overline{C_7}$ of C_7; $\overline{C_7}$ has no proper* 5*-total coloring; and the complement $\overline{C_3 \sqcup C_4}$ is of Type* 1*.*

Proof. This is `paper_small_regular`. For $\overline{C_7}$: a total matching has at most two vertices, since $\overline{C_7}$ has independence number 2, and removing an independent pair leaves five vertices that carry at most two matching edges; so $\alpha_T \le 4$ and the uniform weighting gives 21/4 ($\alpha_T = 4$: on the vertices $0, \dots, 6$ of C_7 take the vertices 0, 1 and the edges 24, 35). □

The prism Y_5 is not bipartite, since its rim is a 5-cycle; so it is a cubic graph that is not bipartite, is conformable, and has $\chi''_f(Y_5) > \Delta + 1$.

Proof of Theorem 5. Let G be regular on $n \le 7$ vertices. If $n = 0$ then $\chi''(G) = \chi''_f(G) = 0$. Otherwise the components of G are regular of the same degree, $\chi''(G)$ is the largest χ'' of a component and $\chi''_f(G)$ the largest χ''_f of a component, so it suffices to treat connected G. A single vertex has $\chi'' = \chi''_f = 1$, and for $2 \le n \le 6$ this is Proposition 20. For $n = 7$ the degree is even (Proposition 21); the 2-regular graphs are C_7 and $C_3 \sqcup C_4$ and the 4-regular ones their complements, so the regular graphs are these four, the edgeless graph and K_7. Proposition 21 gives $\lceil \chi''_f \rceil \ge 4, 4, 6, 5, 1, 7$ for them, and $\chi''(C_7) = 4$, $\chi''(C_3 \sqcup C_4) = 4$, $\chi''(\overline{C_7}) = 6$ by exact search (`recompute.py`, block 10, which also computes the six values $\chi''_f = 7/2, 4, 21/4, 5, 1, 7$ by linear programming), while the other three are of Type 1. Hence $\lceil \chi''_f(G) \rceil = \chi''(G)$ for all of them. At order 8, M_8 is cubic with $\chi''_f(M_8) = 4$ and $\chi''(M_8) = 5$ (Proposition 21), and it is the only connected cubic graph on 8 vertices of Type 2 (Proposition 20). □

Of the six regular graphs on 7 vertices the three of Type 2 are C_7, $C_3 \sqcup C_4$ and $\overline{C_7}$, and all have $\chi''_f > \Delta + 1$. The first two are conformable: color C_7 cyclically by $0, 1, 0, 1, 0, 1, 2$,

and $C_3 \sqcup C_4$ by $0,1,2$ on the triangle and $0,1,0,1$ around the square; the classes have sizes $3,3,1$. The third is not: its color classes have at most 2 vertices, so five classes of odd size would cover 5 of the 7 vertices (`recompute.py`, block 10, decides the three cases).

Peel certificates and the auxiliary class.

Proof of Theorem 6. This is `paper_peel`. For (i): given a certificate and a nonempty $F \subseteq V \setminus A$, the earliest vertex of F in the list has all its F-neighbors later in the list, so fewer than f of them. Conversely, if $G[V \setminus A]$ is strictly f-degenerate, repeatedly removing a vertex with fewer than f neighbors among those left builds the list. For (ii): at $q = \Delta + 1$ every vertex of $V_\Delta \setminus A$ has $f = 1$, so two adjacent vertices u, w of $V_\Delta \setminus A$ would give $F = \{u, w\}$ with no vertex of degree 0 in $G[F]$. So $V_\Delta \setminus A$ is independent, and with $A \cap V_\Delta$ it splits V_Δ into two independent sets. (iii) follows from (ii), since a bipartite graph has no closed walk of odd length. □

Proposition 22 (Regular graphs; Lean). *Let G be r-regular with at least one vertex. Some set A has a peel certificate for $(G, A, r+1)$ if and only if G has a proper vertex 2-coloring. If moreover G is connected, then for every proper 2-coloring of G and every set A, a peel certificate for $(G, A, r+1)$ exists if and only if A is one of the two color classes.*

Proof. This is `paper_peel_regular`. Here $V_\Delta = V$ and every vertex has $f = 1$, so a certificate for A exists exactly when A and $V \setminus A$ are both independent. □

Corollary 23 (Odd cycles and complete graphs; Lean). *For $m \geq 1$ no set A of vertices of the cycle C_{2m+1} has a peel certificate for $(C_{2m+1}, A, \Delta(C_{2m+1}) + 1)$, and for $n \geq 3$ no set A of vertices of K_n has a peel certificate for $(K_n, A, \Delta(K_n) + 1)$.*

Here $\Delta(C_{2m+1}) + 1 = 3$ and $\Delta(K_n) + 1 = n$.

Proof. Every vertex of C_{2m+1} and of K_n has maximum degree, so this follows from Theorem 6(iii); in Lean it is `paper_peel_odd_cycle_complete`. □

Proposition 24 (Counting in $\mathcal{A}_D$; Lean). *Let $(H, J) \in \mathcal{A}_D$ with center v_0 and $M = J \setminus \delta_H(v_0)$. Then $d_H(v_0) + |M| = D$ and H has $1 + d_H(v_0) + 2|M|$ vertices. If $|M| = 0$, every proper k-edge coloring of H, for any k, is rainbow on J.*

Proof. This is `paper_aux_counts`. Every vertex other than v_0 lies on exactly one edge of J: the $d_H(v_0)$ neighbors of v_0 on an edge at v_0, the others on an edge of M. If $M = \emptyset$, J is the star at v_0, whose edges pairwise meet. □

By Vizing's theorem [40] [to check] every member with $|M| = 0$ therefore has a proper $(D+1)$-edge coloring rainbow on J; this existence step is not part of the Lean statement.

Proposition 25 (Adding edges; Lean). *Let $H_1 \subseteq H_2$ be graphs on the same vertex set, let $(H_1, J) \in \mathcal{A}_D$ with center v_0, and suppose $\Delta(H_2) \leq D$ and every neighbor of v_0 in H_2 is a neighbor of v_0 in H_1. Then $(H_2, J) \in \mathcal{A}_D$ with center v_0, and if H_2 has a proper k-edge coloring rainbow on J, so has H_1.*

Proof. This is `paper_aux_monotone`: the axioms of $\mathcal{A}_D$ do not mention the edges away from v_0 beyond the degree bound, and a coloring of H_2 restricts to H_1. □

So for the $(D+1)$-color question it suffices to treat the edge-maximal members. For $|M| = 1$ we use the fan argument of Vizing [40], in the following form, written out here for completeness. In a partial edge coloring, a color is *missing* at x if no colored edge at x has it.

Lemma 26 (Fan lemma; written proof). *Let φ be a proper coloring, with colors $\{0, \dots, k-1\}$, of a set F of edges of a graph, and let vw be an edge not in F. If a color is missing at v, and each vertex of $\{w\} \cup \{y : vy \in F\}$ has a missing color, which may differ from vertex to vertex, then the colors of F can be changed so that $F \cup \{vw\}$ is properly colored with the same k colors.*

Proof. Let α be a color missing at v, and build a sequence $y_0 = w, y_1, \dots, y_m$ of distinct neighbors of v with $\varphi(vy_i)$ missing at y_{i-1} for $i \geq 1$, as long as possible. *Shifting to y_i* means recoloring vy_r with $\varphi(vy_{r+1})$ for $r < i$; it keeps the coloring proper and leaves vy_i to be colored. If some y_i misses a color γ that is missing at v, shift to y_i and color vy_i with γ. Otherwise let β be missing at y_m. As β is not missing at v, some edge vu has color β; by maximality $u = y_j$ with $1 \leq j < m$, so β is missing at y_{j-1} and at y_m, and neither misses α. Each of v, y_{j-1}, y_m is an end of its chain of colors α, β, so the chain P from y_{j-1} and the chain Q from y_m cannot both end at v. If $v \notin P$, swap α, β on P: the edges at v keep their colors, the colors $\varphi(vy_r)$ with $r \leq j-1$ are neither α nor β, and now y_{j-1} misses α; shift to y_{j-1} and color vy_{j-1} with α. Otherwise P joins y_{j-1} to v, so Q avoids v and y_{j-1}; swap α, β on Q, after which y_m misses α and the sequence keeps its property; shift to y_m and color vy_m with α. □

Proof of Theorem 7. In $H - M_c$ the vertices v_0, x, y and those covered by M' have degree at most $D-1$. Every other vertex is uncovered by the maximal matching M' of $H - \{v_0, x, y\}$, so no two of them are adjacent; hence the set W of vertices of degree D in $H - M_c$ is independent. Color the edges of $H - M_c$ with $\{0, \dots, D-1\}$ one at a time by Lemma 26, first the edges with no end in W and then the edges with an end in W, applying the lemma with v the end in W when there is one. In the first phase a vertex of W has no colored edge and every other vertex has degree at most $D-1$, so every vertex misses a color at the moment its edge is added. In the second phase $v \in W$ misses a color because vw is not yet colored, and w and every neighbor of v lie outside W and have degree at most $D-1$. The recolorings of the lemma color no new edge, so these properties persist. This gives a proper D-edge coloring of $H - M_c$, and the color D on the matching M_c completes a proper $(D+1)$-edge coloring of H. The edges at v_0 are not in M_c, so they have distinct colors below D, and xy has color D. For $(H, J) \in \mathcal{A}_D$ with $M = \{xy\}$, the vertices x, y lie on no edge at v_0, so they are not neighbors of v_0, and $d_H(v_0) = D - 1$ by Proposition 24. □

The script `recompute.py` runs this procedure (block 10) on 150 seeded random instances for each D with $3 \leq D \leq 8$, on all 71 labeled maximal members at $(D, s, p) = (5, 4, 1)$ of Proposition 28, which form 8 symmetry classes, and on one

dense maximal member with $D+2$ vertices for each $3 \le D \le 10$, and checks the resulting colorings. For two edges in M the same peeling leaves a weaker conclusion.

Lemma 27 (Peeling with $|M| = 2$; written proof). *Let $(H, J) \in \mathcal{A}_D$ with center v_0 and $M = \{x_1y_1, e_2\}$. Let M_1' be a maximal matching of the graph $H - \{v_0, x_1, y_1\}$ with the edge e_2 deleted, and $M_D = \{x_1y_1\} \cup M_1'$. Then the only edge of $H - M_D$ that can join two vertices of degree D in $H - M_D$ is e_2.*

Proof. Vertices v_0, x_1, y_1 and those covered by M_1' have degree at most $D-1$ in $H - M_D$. Two other vertices joined by an edge other than e_2 would let that edge extend M_1'. □

So the vertices of degree D in $H - M_D$ induce a forest, and by Fournier's theorem [15] [to check] $H - M_D$ has a proper D-edge coloring; this alone does not make the coloring rainbow on J, since e_2 is colored below D.

Proof of Theorem 8. Both 01 and 34 have color 0. A Kempe change makes them different only if it moves exactly one of them and involves color 0; the Lean proof of `kempe_dist_eq_two_counterexample` checks every choice of two colors and Kempe-closed set and finds a repeated color on J in each case. Swapping 1, 2 on the chain 0–2–4 and then 0, 1 on the edge 01 gives the colors 1, 2, 0, 2, 3, 3, 1, which are proper and rainbow on J. □

Proposition 28 (The auxiliary class for $D \le 5$; certificate). *For a triple (D, s, p) with $s + p = D$ and $2 \le s \le D$, fix the vertex set $\{0, \dots, s + 2p\}$, the center $v_0 = 0$ and the edges $J = \{01, \dots, 0s\} \cup \{(s+1)(s+2), \dots, (s+2p-1)(s+2p)\}$; a* labeled member *is a graph H on this vertex set with $(H, J) \in \mathcal{A}_D$, and a coloring c has* the star fixed *if $c(0i) = i - 1$ for $1 \le i \le s$. For each of the ten triples with $D \le 5$, every edge-maximal labeled member has a proper $(D+1)$-edge coloring rainbow on J; there are 702 such graphs up to the symmetries of J and 44647 labeled ones, of which 675 and 44484 have $D = 5$. Measuring by single Kempe chain changes, every proper 4-edge coloring of a maximal member with $(D, s, p) = (3, 2, 1)$ is within distance 2 of a coloring rainbow on J: of the 76 such colorings with the star fixed, the distances 0, 1, 2 occur 40, 32 and 4 times (480, 384, 48 of all 912). With the star fixed, the distances 0, 1, 2 occur 1008, 1752 and 516 times among the 3276 colorings of the 17 labeled maximal members at $(4, 3, 1)$ (196560 colorings without fixing the star), and the distances 0, 1, 2, 3 occur 145824, 258192, 68832 and 240 times among the 473088 colorings of the 71 labeled maximal members at $(5, 4, 1)$. At $(4, 2, 2)$ there are 17 maximal members up to symmetry (138 labeled). For either edge of M taken as x_1y_1 and every maximal M_1', the peeling of Lemma 27 leaves the vertices of degree 4 independent or joined only by e_2. A matching of $H - v_0$ that contains x_1y_1 and not e_2 and covers every vertex of degree 4 in H exists for at least one choice of x_1y_1 in 14 of the 17, and for both choices in 9. Of the 728 connected graphs on the vertex set $\{0, \dots, 4\}$, 565 have a peel certificate at $q = \Delta + 1$.*

Proof. The script `checks/auxiliary.py` repeats the enumeration recorded with the finding, which also gives the rainbow colorings, and checks the labeled members with $p = 1$ against its own enumeration. For every coloring with the star fixed it runs a breadth-first search over all single Kempe chain changes, every component of every pair of colors, and requires that a coloring rainbow on J is reached. Its output, `checks/auxiliary.out`, ends with `AUXILIARY CHECK: PASS`. The four-edge coloring counts at $(3, 2, 1)$ are 12 times the star-fixed counts, one for each ordered pair of colors on the two star edges. Kempe distances at $(5, 3, 2)$ and $(5, 2, 3)$ were not searched. □

The families G_Δ^* and H_Δ^*.

Lemma 29 (Forcing in $K_{\Delta-1,\Delta}$; Lean). *Let $\Delta \ge 2$ and let ψ be a proper total coloring of $K_{\Delta-1,\Delta}$, with sides P ($\Delta-1$ vertices) and T (Δ vertices), by a set C of $\Delta+1$ colors. Then some color c^* is given to every vertex of P and to no edge; the vertices of T get distinct colors, which are exactly the colors other than c^*; at each $t \in T$ the edges carry every color other than c^* and $\psi(t)$; and every color other than c^* is on exactly $\Delta - 1$ edges.*

Proof. This is `k_pred_succ_monochromatic_forcing`. Each $p \in P$ has degree Δ, so its closed star uses all $\Delta+1$ colors. A color γ not used on P therefore lies on an edge at every p, hence on $\Delta - 1$ edges covering $\Delta - 1$ vertices of T, and on at most one vertex of T; no color used on P can be used on T. Summing over T, $\Delta \le \Delta + 1 - |\psi(P)|$, so $|\psi(P)| = 1$ and T takes each other color once. □

Proposition 30 (The graph G_3^*; Lean). *G_3^* has 20 vertices and 30 edges, and every proper 4-total coloring has one class of 14 elements and three of 12.*

Proof. This is `gDeltaStar_three_profile`, the case $\Delta = 3$ of Theorem 9. □

Proof of Theorem 9. In Lean this is `paper_gDeltaStar`. A triangle would lie in one block, which is bipartite, or use cross edges, which form a matching; a block contains a 4-cycle as $\Delta - 1 \ge 2$. For Type 1, fix an array $A = (A_{i,j})_{i \ne j}$ with entries in $\mathbb{Z}_\Delta$, each row a bijection onto $\mathbb{Z}_\Delta$, and $A_{i,j} \ne A_{j,i}$. With $m = \Delta + 1$ and indices mod m, for even Δ put $A_{i,i+t} = t - 1$ ($1 \le t \le \Delta$); for odd Δ, $m = 2h$, put the same except that the shifts $t = h$ and $t = h + 1$ get $h - 1, h$ when $i < h$ and $h, h - 1$ when $i \ge h$. Color all private vertices and cross edges with Δ, $v_{i,j}$ with $A_{i,j}$, and $p_{i,r}v_{i,j}$ with $A_{i,j} + r + 1$ mod Δ. Hence $\chi_f'' = \Delta + 1$ by Proposition 13. In any proper $(\Delta+1)$-total coloring, Lemma 29 on block B_i gives a color c_i on P_i; the cross edge at a terminal of B_i cannot carry a color other than c_i, since the block edges at that terminal carry all others but its own vertex color. So the cross edge $v_{i,j}v_{j,i}$ has color $c_i = c_j$, and all c_i equal one color c^*. Its class has the $(\Delta + 1)(\Delta - 1)$ private vertices and the $\binom{\Delta+1}{2}$ cross edges, and every other color has one terminal and $\Delta - 1$ edges in each block. The difference $(\Delta + 1)(\Delta - 2)/2$ is at least 2. For the lists, let $\Omega = \{0, \dots, \Delta + 1\}$, give all of B_i the list $\Omega \setminus \{i\}$, give the cross edge $v_{t+1,t+2}v_{t+2,t+1}$ the list $\Omega \setminus \{t\}$ (indices mod $\Delta + 1$), the cross edge between B_0 and B_2 the list $\Omega \setminus \{\Delta + 1\}$, and the other cross edges $\Omega \setminus \{0\}$. By Lemma 29, a cross edge leaving B_i must take a color in $S_i = \{i, c_i\}$, so the sets S_i pairwise intersect; as their union contains $\{0, \dots, \Delta\}$, they share a color ω, and the lists above exclude every possible ω. □

Proof of Theorem 10. A $(\Delta + 1)$-total coloring is never equitable (Theorem 9), and fewer than $\Delta + 1$ colors are impossible, so it suffices to recolor a total matching I^* of the coloring

ψ of the previous proof with a new color $\Delta+1$. Every class of ψ minus I^* stays a total matching, and I^* becomes a class, so the result is proper whenever I^* is a total matching. In ψ the class Δ has $(\Delta+1)(3\Delta-2)/2$ elements and each class $c<\Delta$ has $\Delta(\Delta+1)$. In block B_i the edges from $p_{i,r}$ to the terminal of color $x+1+r$, for $0\le r\le\Delta-2$, form a matching that avoids the terminal of color x, and their colors are $x+2+2r$ mod Δ.

Odd $\Delta=2h-1$, the construction recorded with the finding: I^* consists of the private vertices of $B_0,\dots,B_{h-1}$ and, for each $i\ge h$, the terminal $v_{i,0}$, of color $x=A_{i,0}$, with the matching above. As Δ is odd, the colors $x+2+2r$ are all colors of $\mathbb{Z}_\Delta$ but x, so each block B_i with $i\ge h$ removes one element of every color $c<\Delta$. The class Δ loses $h(\Delta-1)$ elements and each class $c<\Delta$ loses h, so all $\Delta+2$ classes have $h(2\Delta-1)$ elements. I^* is a total matching: the neighbor $v_{0,i}$ of $v_{i,0}$ outside B_i is not in I^*, and the private vertices of B_i for $i\ge h$ are not in I^*.

Even $\Delta=2q$, a construction written for this paper. The argument recorded with the finding uses, in each block, two terminals and a matching whose colors run over the other $2q-2$ colors, and does not show that such a matching exists (Section 7). Here I^* consists of the private vertices of $B_0,\dots,B_{q-1}$; the $q-1$ cross edges $v_{i,q}v_{q,i}$ with $i\le q-2$; in B_q, the q edges $p_{q,a+\sigma}v_{q,q+a+1}$ for $0\le a\le q-1$, where $\sigma=0$ if q is even and $\sigma=1$ if q is odd; and in each block B_i with $q+1\le i\le 2q$, one terminal $v_{i,j}$ with $j\le q-1$ of a color x with prescribed parity, together with the matching above. Since $A_{q,q+a+1}=a$, the B_q edges have colors $2a+1+\sigma$: all q odd colors when q is even, all q even colors when q is odd. In a parity block with terminal color x, the colors $x+2+2r$, $0\le r\le 2q-2$, cover every color of the parity of x twice except x, which they cover once, and the terminal itself has color x; so the block removes two elements of each color of that parity. Choose odd x in the first $q/2$ of these blocks if q is even, or the first $(q+1)/2$ if q is odd, and even x in the others; the terminals $v_{i,0},\dots,v_{i,q-1}$ have $q\ge 2$ consecutive colors, so both parities are available. Then every odd color loses $q+1$ elements and every even color q; the class Δ loses $q(2q-1)+q-1$. The class Δ and the even classes end with $4q^2+q$ elements, the new class and the odd classes with $4q^2+q-1$. I^* is a total matching: the chosen cross edges are disjoint and avoid the private vertices; the B_q edges use terminals $v_{q,j}$ with $j\ge q+1$, which are not on the chosen cross edges; a chosen terminal $v_{i,j}$ with $i>q$ has its outside neighbor $v_{j,i}$ in a block B_j with $j<q$, whose terminals are not in I^*; and the matchings avoid the chosen terminals and the private vertices in I^*. `recompute.py` (block 7) carries out both constructions and checks properness and class sizes for $3\le\Delta\le 20$. □

Proof of Theorem 11. H^*_Δ is bipartite with sides $P_0\cup T_1$ and $T_0\cup P_1$ and has the stated orders; a block contains a 4-cycle. Color the private vertices and the cross edges with Δ, and, reducing the following colors modulo Δ into $\{0,\dots,\Delta-1\}$, $t_{0,j}$ with j, $t_{1,j}$ with $j+1$, $p_{0,r}t_{0,j}$ with $j+r+1$ and $p_{1,r}t_{1,j}$ with $j+r+2$; this is a proper $(\Delta+1)$-total coloring. In any proper $(\Delta+1)$-total coloring, Lemma 29 on both blocks forces the color c^* of P_0 and of P_1 onto every cross edge, so the class of c^* has $2(\Delta-1)+\Delta=3\Delta-2$ elements and every other class 2Δ. For $\Delta=3$ the classes $7,6,6,6$ form an equitable coloring, so $\chi''_e(H^*_3)=4$. For $\Delta\ge 4$ the spread $\Delta-2\ge 2$ excludes $\Delta+1$ colors, and recoloring the side $P_0\cup T_1$, which is independent, with a new color leaves every class with $2\Delta-1$ elements; so $\chi''_e(H^*_\Delta)=\Delta+2$.

For the list total chromatic number, every bipartite graph G has $\chi''_\ell(G)\le\Delta(G)+2$: color the vertices greedily from their lists, which is possible as each vertex has at most Δ neighbors, delete from each edge list the colors of its two ends, and complete the edges by Galvin's theorem [17] [to check], since every edge list keeps at least Δ colors. For the lower bound, let $\Delta\ge 5$, let B and A be disjoint sets of $\Delta+1$ colors, and give block B_0 the list B and block B_1 the list A. By Lemma 29, the cross edge $t_{0,j}t_{1,j}$ can only take $c^*_0\in B$ or $c^*_1\in A$, so it has no color when its list omits both. Give each cross edge the complement of an *omission set* of $\Delta+1$ colors, so that every pair in $B\times A$ lies in some omission set; at most Δ omission sets are needed. For odd Δ split $A=A_1\sqcup A_2$ and $B=B_1\sqcup B_2$ into halves and use the four sets $A_i\cup B_j$. For even $\Delta=2h$, split B into parts of sizes $h+1$ and h and A into parts A_1,A_2,A_3 of sizes h, $\lfloor(h+1)/2\rfloor$, $\lceil(h+1)/2\rceil$, and use the five sets $A_1\cup B_1$, $A_1\cup B_2$, $A_2\cup A_3\cup B_2$, $A_2\cup B_1$, $A_3\cup B_1$, padded to $\Delta+1$ colors; $5\le\Delta$ as $\Delta\ge 6$. `recompute.py` (block 8) checks the colorings for $3\le\Delta\le 8$ and the covering for $5\le\Delta\le 12$.

For $\Delta=4$ this argument fails, and so does every choice of one list per block, which is the last clause. Give every element of B_0 a list C_0 and every element of B_1 a list C_1, with $|C_0|=|C_1|=5$, and put $k=|C_0\setminus C_1|$. By Lemma 29 a cross edge $t_{0,j}t_{1,j}$ can take a color x only if $x\notin C_0\setminus\{c^*_0\}$ and $x\notin C_1\setminus\{c^*_1\}$. A color outside $C_0\cup C_1$ is always available, since the cross edges form a matching, so a cross-edge list that leaves no color lies in $C_0\cup C_1$ and is the complement there of an omission set of k colors. If $c^*_0=c^*_1=m\in C_0\cap C_1$ the only available color is m, and if $c^*_0=b\notin C_1$ and $c^*_1=a\notin C_0$ the available colors are a and b. So the lists admit no coloring only if every color of $C_0\cap C_1$ and every pair in $(C_0\setminus C_1)\times(C_1\setminus C_0)$ lie in one of the four omission sets. A set of k colors contains at most $\lfloor k/2\rfloor\lceil k/2\rceil$ of these k^2 pairs. For odd k, $4\lfloor k/2\rfloor\lceil k/2\rceil<k^2$. For even k, a set that meets $C_0\cap C_1$ contains fewer than $k^2/4$ pairs, so four sets covering all pairs miss $C_0\cap C_1$, which has $5-k\ge 1$ colors. Hence some choice of c^*_0,c^*_1 leaves every cross edge an available color. Relabel a proper 5-total coloring of $K_{3,4}$ on each block so that P_i gets c^*_i; by Lemma 29 the terminals of B_i get the four colors of $C_i\setminus\{c^*_i\}$, and the colors available to a cross edge do not depend on which terminal gets which color. Permuting the terminals of a block keeps its coloring proper, so we may put each color common to the two sets of terminal colors on $t_{0,j}$ and $t_{1,j}$ for the same j, and then move the coloring of B_1 by the automorphism that shifts $t_{1,0},\dots,t_{1,3}$ cyclically by one index. Now $t_{0,j}$ and $t_{1,j}$ get different colors, since a common color at index j in B_0 is at index $j+1$ in B_1. The cross edges form a matching, so each takes an available color from its list independently. The script `recompute.py` (block 8) checks only the inequalities between $4\lfloor k/2\rfloor\lceil k/2\rceil$ and k^2 for $0\le k\le 5$. □

Proposition 31 uses four blocks instead.

Proposition 31 (A 4-regular bipartite graph; written proof and exhaustive check). *Let $H^{(4)}_4$ consist of four copies $B_0,\dots,B_3$ of $K_{3,4}$, with private vertices $p_{i,r}$ and terminals $t_{i,j}$, and the cross edges*

$t_{i,2}t_{i+1,0}$ *and* $t_{i,3}t_{i+1,1}$ *for* $i \in \mathbb{Z}_4$. *It has* 28 *vertices and* 56 *edges, is* 4*-regular and bipartite, is of Type* 1*, and* $\chi''_\ell(H_4^{(4)}) = 6$.

Proof. Each terminal has three block edges and one cross edge; the sides are the private vertices of B_0, B_2 with the terminals of B_1, B_3, and the rest. An exact search finds a proper 5-total coloring (`recompute.py`, block 9), and $\chi''_\ell \le 6$ by the bipartite bound in the proof of Theorem 11. Let $\Omega = \{0, \dots, 5\}$, give all of B_i the list $\Omega \setminus \{i\}$, and give the two cross edges between B_i and B_{i+1} the lists $\Omega \setminus \{f_i\}$ and $\Omega \setminus \{f'_i\}$ with $(f_i, f'_i) = (0,1), (1,3), (4,5), (0,2)$ for $i = 0, 1, 2, 3$. By Lemma 29 with $C = \Omega \setminus \{i\}$, every cross edge at B_i takes a color in $S_i = \{i, c_i\}$ with $c_i \ne i$. So the two cross edges between B_i and B_{i+1} need $S_i \cap S_{i+1} \not\subseteq \{f_i\}$ and $S_i \cap S_{i+1} \not\subseteq \{f'_i\}$. Of the $5^4 = 625$ choices of (c_0, c_1, c_2, c_3), 36 have all four intersections $S_i \cap S_{i+1}$ nonempty, and none meets all eight conditions (`recompute.py`, block 9). □

4. COMPUTATIONAL VERIFICATION

The Lean build. The library `VizingTotal` of the finding was built with Lean 4 version `4.35.0` and Mathlib `v4.35.0`, with no `native_decide` and no `sorry`. The module `VizingTotal.PaperDraft1` gathers the statements cited in Section 3 as conjunctions of theorems of the library, defines `FracTotalEq`, and adds three short proofs: the clique bound of Proposition 13, $\chi''_f(Cr_2) = 3$, and the 4-cycle of G^*_Δ. The file `VerifyDraft1.lean` prints the axioms of every public theorem in its import closure, 205 declarations, and every one depends on a subset of `propext`, `Classical.choice` and `Quot.sound`. The printout is a committed receipt from the author's machine; no independent replay has been run.

Certificates. Two scripts in `checks/` use only the Python standard library and exact integer and rational arithmetic. `census.py` supports Proposition 20 and Theorem 5; `auxiliary.py` supports Proposition 28 and calls the enumerator recorded in the evidence folder of the finding. Each writes the output bound in Section 5 by its SHA-256 hash.

Exact replay. `recompute.py` runs the Python box of Section 1 verbatim and recomputes, without floating point, the values of Table 1, the exhaustive values of α_T for small Möbius ladders and prisms, the columns and weights of the proof of Theorem 4 for $5 \le n \le 40$, the colorings of Proposition 18 and Theorems 2, 10 and 11, the omission sets, the 625 cases of Proposition 31, the six regular graphs on 7 vertices, and the peeling of Theorem 7 on random and dense instances; it also checks that both certificate outputs end in `PASS`. One run takes under a minute.

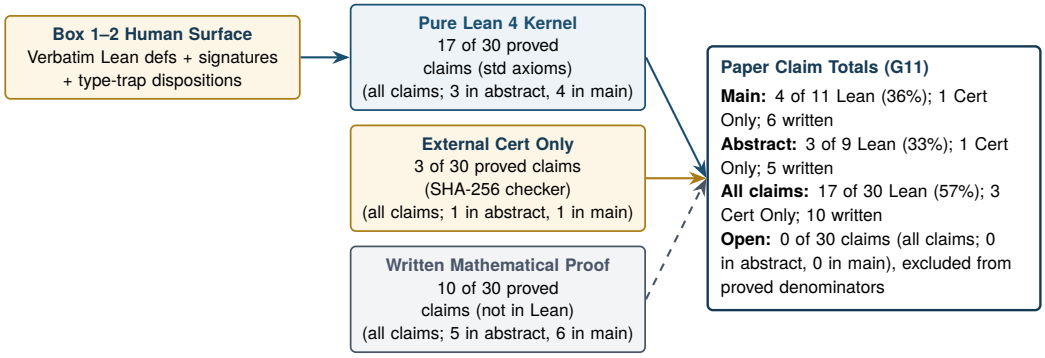
Reading Section 5. The next section is generated by the project's checking tool from the Lean sources and the manifest of this draft. Its tables restate every numbered result without its context and give its tier: Lean kernel, certificate only, or written proof. A written proof that cites a computation over a finite range keeps the tier *written*. Section 7 explains the terms the generated section uses.

5. FORMAL VERIFICATION & READER'S TRUST BOUNDARY

What You Must Trust to Accept This Paper:

1. **Lean 4 Kernel & Toolchain:** leanprover/lean4:v4.35.0; committed axiom receipt anchored to git commit 0ddf107833cb (independent gate-box kernel replay pending); standard axioms only (propext, Classical.choice, Quot.sound).
2. **Core Definitions & Fidelity (Boxes 1–2):** Verify that the verbatim Lean definitions in Box 1 match Section 2 and that each Lean signature in Box 2 matches its stated theorem.
3. **External Certificates & Written Proofs:** 3 SHA-256-pinned external artifact(s) plus any claims marked Written or Unsettled in Table 3 below.

Reader's Trust-Flow Architecture (Distinct Evidence Paths to Claim Totals)



Anti-Shortcut Audit Strip — Source-Level Approx (Gate Replay Pending)

WARN: Committed receipt (gate replay pending) | PASS: sorry/admit: 0 (per committed receipt) | PASS: native_decide: 0 (per committed receipt) | PASS: Custom axioms & meta: 0 (per committed receipt) | PASS: Coverage: 205/205 public theorems (source-level approx) | WARN: Fidelity: 17/17 exact (draft; auditor/formalizer relationship unknown) | WARN: Witnesses: pending gate replay | WARN: Human referee: pending

Box 1 — Verbatim Lean 4 Custom Definitions (Human Audit Surface)

VizingTotal.TotalElem (VizingTotal/Basic.lean:62-62) — Total graph T(G) (Section 2)

Context binders in scope: `variable {V : Type*}`

```
abbrev TotalElem (G : SimpleGraph V) : Type _ := V ⊕ G.edgeSet
```

VizingTotal.totalAdj (VizingTotal/Basic.lean:65-69) — Total graph T(G) (Section 1)

Context binders in scope: `variable {V : Type*}`

```
def totalAdj (G : SimpleGraph V) : TotalElem G → TotalElem G → Prop
  | Sum.inl u, Sum.inl v => G.Adj u v
  | Sum.inl u, Sum.inr e => u ∈ e.1
  | Sum.inr e, Sum.inl u => u ∈ e.1
  | Sum.inr e₁, Sum.inr e₂ => e₁ ≠ e₂ ∧ ∃ v : V, v ∈ e₁.1 ∧ v ∈ e₂.1
```

VizingTotal.TotalGraph (VizingTotal/Basic.lean:86-89) — Total graph T(G) (Section 1)

Context binders in scope: `variable {V : Type*}`

```
def TotalGraph (G : SimpleGraph V) : SimpleGraph (TotalElem G) where
  Adj := totalAdj G
  symm := totalAdj_symm G
  loopless := totalAdj_loopless G
```

VizingTotal.IsTotalIndep (VizingTotal/Basic.lean:145-146) — Total matchings (Section 1)

Context binders in scope: `variable {V : Type*} ; variable [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
def IsTotalIndep (I : Finset (TotalElem G)) : Prop :=
  ∀ x ∈ I, ∀ y ∈ I, ¬ (TotalGraph G).Adj x y
```

VizingTotal.alphaTotal (VizingTotal/Basic.lean:222-223) — Total matchings (Section 1)

Context binders in scope: `variable {V : Type*} ; variable [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
def alphaTotal : ℕ :=
  (Finset.univ.filter (IsTotalIndep G)).sup Finset.card
```

VizingTotal.IsTotalClique (VizingTotal/Basic.lean:256-257) — Cliques of T(G) (Proposition on cliques)

Context binders in scope: `variable {V : Type*} ; variable [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
def IsTotalClique (Q : Finset (TotalElem G)) : Prop :=
  ∀ x ∈ Q, ∀ y ∈ Q, x ≠ y → (TotalGraph G).Adj x y
```

VizingTotal.omegaTotal (VizingTotal/Basic.lean:265-266) — Cliques of T(G) (Proposition on cliques)

Context binders in scope: `variable {V : Type*} ; variable [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
def omegaTotal : ℕ :=
  (Finset.univ.filter (IsTotalClique G)).sup Finset.card
```

VizingTotal.IsProperTotalColoring (VizingTotal/Basic.lean:280-281) — Total colorings (Section 2)

Context binders in scope: `variable {V : Type*} ; variable [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
def IsProperTotalColoring (k : ℕ) (f : TotalElem G → Fin k) : Prop :=
  ∀ x y : TotalElem G, (TotalGraph G).Adj x y → f x ≠ f y
```

VizingTotal.HasProperTotalColoring (VizingTotal/Basic.lean:284-285) — Total colorings (Section 2)

Context binders in scope: `variable {V : Type*} ; variable [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
def HasProperTotalColoring (k : ℕ) : Prop :=
  ∃ f : TotalElem G → Fin k, IsProperTotalColoring G k f
```

VizingTotal.IsType1 (VizingTotal/Basic.lean:361-362) — Total colorings (Section 2)

Context binders in scope: `variable {V : Type*} ; variable [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
def IsType1 : Prop :=
  HasProperTotalColoring G (G.maxDegree + 1)
```

VizingTotal.IsType2 (VizingTotal/Basic.lean:365-366) — Total colorings (Section 2)

Context binders in scope: `variable {V : Type*} ; variable [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
def IsType2 : Prop :=
  ¬ HasProperTotalColoring G (G.maxDegree + 1) ∧ HasProperTotalColoring G
    ↪ (G.maxDegree + 2)
```

VizingTotal.deficiency (VizingTotal/Basic.lean:369-370) — Total colorings (Section 2)

Context binders in scope: `variable {V : Type*} ; variable [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
def deficiency : ℕ :=
  ∑ v : V, (G.maxDegree - G.degree v)
```

VizingTotal.IsStarSaturated (VizingTotal/Basic.lean:382-383) — Total colorings (Section 2)

Context binders in scope: `variable {V : Type*} ; variable [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
def IsStarSaturated (I : Finset (TotalElem G)) : Prop :=
  IsTotalIndep G I ∧ (indepVerts G I).card + 2 * (indepEdges G I).card = Fintype.card
    ↪ V
```

VizingTotal.IsConformable (VizingTotal/Basic.lean:387-392) — Total colorings (Section 2)

Context binders in scope: `variable {V : Type*} ; variable [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
def IsConformable : Prop :=
  ∃ c : V → Fin (G.maxDegree + 1),
    (∀ u v : V, G.Adj u v → c u ≠ c v) ∧
    (Finset.univ.filter (fun col : Fin (G.maxDegree + 1) =>
      (Finset.univ.filter (fun v : V => c v = col)).card % 2 ≠ Fintype.card V %
        ↪ 2)).card ≤
      deficiency G
```

VizingTotal.FractionalTotalColoring (VizingTotal/Basic.lean:397-401) — Fractional total colorings (Section 2)

Context binders in scope: `variable {V : Type*} ; variable [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
structure FractionalTotalColoring where
  w : Finset (TotalElem G) → ℚ
  nonneg : ∀ I, 0 ≤ w I
  support_indep : ∀ I, w I ≠ 0 → IsTotalIndep G I
  cover : ∀ x : TotalElem G, 1 ≤ ∑ I : Finset (TotalElem G), if x ∈ I then w I else 0
```

VizingTotal.FractionalTotalColoring.totalWeight (VizingTotal/Basic.lean:404-405) — Fractional total colorings (Section 2)

Context binders in scope: `variable {V : Type*} ; variable [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
def FractionalTotalColoring.totalWeight (P : FractionalTotalColoring G) : ℚ :=
  ∑ I : Finset (TotalElem G), P.w I
```

VizingTotal.FracTotalEq (VizingTotal/PaperDraft1.lean:30-32) — Fractional total colorings (Section 2)

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
def FracTotalEq (q : ℚ) : Prop :=
  (∃ P : FractionalTotalColoring G, P.totalWeight = q) ∧
  ∀ P : FractionalTotalColoring G, q ≤ P.totalWeight
```

VizingTotal.MobiusLadder (VizingTotal/MobiusFrobenius.lean:59-73) — Graph families (Section 2)

```
def MobiusLadder (k : ℕ) : SimpleGraph (Fin (2 * k)) where
  Adj x y := x ≠ y ∧
    ((x.val + 1) % (2 * k) = y.val ∨ (y.val + 1) % (2 * k) = x.val ∨
     (x.val + k) % (2 * k) = y.val)
  symm := ⟨by
    intro x y ⟨hne, h⟩
    refine ⟨hne.symm, ?_⟩
    rcases h with h1 | h2 | h3
    · exact Or.inr (Or.inl h1)
    · exact Or.inl h2
    · right; right
      have hx := x.isLt; have hy := y.isLt
      rw [mod_lt_two_mul (by omega)] at h3 ⊢
      split_ifs at h3 ⊢ <;> omega⟩
  loopless := ⟨fun _ ⟨h, _⟩ => h rfl⟩
```

VizingTotal.mobiusChiF (VizingTotal/MobiusFrobenius.lean:2168-2173) — Theorem on Mobius ladders

```
def mobiusChiF (k : ℕ) : ℚ :=
  if k = 2 ∨ k = 3 then 5
  else if k = 5 then 25 / 6
  else if k = 6 then 30 / 7
  else if k = 9 then 45 / 11
  else 4
```

VizingTotal.RungState (VizingTotal/MobiusFrobenius.lean:857-859) — Mobius ladders and prisms (Section 3)

```
inductive RungState : Type
  | s00 | s01 | s02 | s10 | s12 | s20 | s21 | s22
  deriving DecidableEq, Repr
```

VizingTotal.RungState.phi (VizingTotal/MobiusFrobenius.lean:885-888) — Mobius ladders and prisms (Section 3)

```
def RungState.phi : RungState → ℤ
  | .s00 | .s02 | .s20 => 0
  | .s01 | .s10 => 1
  | .s12 | .s21 | .s22 => 2
```

VizingTotal.RungTransition (VizingTotal/MobiusFrobenius.lean:919-927) — Mobius ladders and prisms (Section 3)

```
structure RungTransition where
  src : RungState
  dst : RungState
  eu : Bool
  ev : Bool
  uin : Bool
  vin : Bool
  spin : Bool
  deriving DecidableEq, Repr
```

VizingTotal.RungTransition.isLegal (VizingTotal/MobiusFrobenius.lean:930-938) — Mobius ladders and prisms (Section 3)

```
def RungTransition.isLegal (t : RungTransition) : Bool :=
  (!t.eu || t.src.top == 0) &&
  (!t.ev || t.src.bot == 0) &&
  !(t.uin && t.vin) &&
  (!t.spin || (!t.eu && !t.ev && !t.uin && !t.vin)) &&
  (!t.uin || (!t.eu && t.src.top != 1)) &&
  (!t.vin || (!t.ev && t.src.bot != 1)) &&
  (t.dst.top == if t.uin then 1 else if t.eu || t.spin then 2 else 0) &&
  (t.dst.bot == if t.vin then 1 else if t.ev || t.spin then 2 else 0)
```

VizingTotal.RungTransition.delta (VizingTotal/MobiusFrobenius.lean:941-943) — Mobius ladders and prisms (Section 3)

```
def RungTransition.delta (t : RungTransition) : ℕ :=
  (if t.eu then 1 else 0) + (if t.ev then 1 else 0) +
  (if t.uin then 1 else 0) + (if t.vin then 1 else 0) + (if t.spin then 1 else 0)
```

VizingTotal.RungTransition.reducedWeight (VizingTotal/MobiusFrobenius.lean:946-947) — Mobius ladders and prisms (Section 3)

```
def RungTransition.reducedWeight (t : RungTransition) : ℤ :=
  3 * (t.delta : ℤ) - 4 + t.src.phi - t.dst.phi
```

VizingTotal.allLegalTransitions (VizingTotal/MobiusFrobenius.lean:950-959) — Mobius ladders and prisms (Section 3)

```
def allLegalTransitions : List RungTransition :=
  allRungStates.flatMap fun s =>
  allRungStates.flatMap fun d =>
  [false, true].flatMap fun eu =>
  [false, true].flatMap fun ev =>
  [false, true].flatMap fun uin =>
  [false, true].flatMap fun vin =>
  [false, true].filterMap fun spin =>
    let t : RungTransition := ⟨s, d, eu, ev, uin, vin, spin⟩
    if t.isLegal then some t else none
```

VizingTotal.PrismL10 (VizingTotal/AuxiliaryKempe.lean:523-530) — Graph families (Section 2)

```
def PrismL10 : SimpleGraph (Fin 10) where
  Adj x y := x ≠ y ∧
    ((x.val < 5 ∧ y.val < 5 ∧ ((x.val + 1) % 5 = y.val ∨ (y.val + 1) % 5 = x.val)) ∨
     (5 ≤ x.val ∧ 5 ≤ y.val ∧ ((x.val - 5 + 1) % 5 = y.val - 5 ∨ (y.val - 5 + 1) % 5
       ↪ = x.val - 5)) ∨
     (x.val < 5 ∧ y.val = x.val + 5) ∨
     (y.val < 5 ∧ x.val = y.val + 5))
  symm := ⟨by intro x y ⟨hne, h⟩; refine ⟨hne.symm, ?_⟩; tauto⟩
  loopless := ⟨fun _ ⟨h, _⟩ => h rfl⟩
```

VizingTotal.CompleteBipartite (VizingTotal/Basic.lean:1286-1289) — Graph families (Section 2)

```
def CompleteBipartite (n : ℕ) : SimpleGraph (Fin n ⊕ Fin n) where
  Adj u v := (u.isLeft ∧ v.isRight) ∨ (u.isRight ∧ v.isLeft)
  symm := ⟨by intro u v h; rcases h with ⟨h1, h2⟩ | ⟨h1, h2⟩ <;> [exact Or.inr ⟨h2,
    ↪ h1⟩; exact Or.inl ⟨h2, h1⟩]⟩
  loopless := ⟨by rintro (a | b) (⟨h1, h2⟩ | ⟨h1, h2⟩) <;> simp at h1 h2⟩
```

VizingTotal.CrownGraph (VizingTotal/Basic.lean:1515-1521) — Graph families (Section 2)

```
def CrownGraph (n : ℕ) : SimpleGraph (Fin n ⊕ Fin n) where
  Adj u v := match u, v with
    | Sum.inl i, Sum.inr j => i ≠ j
    | Sum.inr j, Sum.inl i => i ≠ j
    | _, _ => False
  symm := ⟨by rintro (i | j) (k | l) h <;> exact h⟩
  loopless := ⟨by rintro (i | j) h <;> exact h⟩
```

VizingTotal.IsBipartiteParts (VizingTotal/Basic.lean:1120-1121) — Proposition on regular bipartite graphs

```
def IsBipartiteParts {n : ℕ} (G : SimpleGraph (Fin n ⊕ Fin n)) : Prop :=
  ∀ u v : Fin n ⊕ Fin n, G.Adj u v → (u.isLeft ∧ v.isRight) ∨ (u.isRight ∧ v.isLeft)
```

VizingTotal.OddCycle (VizingTotal/IndependentSeedBarrier.lean:505-508) — Proposition on small regular witnesses

```
def OddCycle (m : ℕ) : SimpleGraph (Fin (2 * m + 1)) where
  Adj u v := u ≠ v ∧ ((u.val + 1) % (2 * m + 1) = v.val ∨ (v.val + 1) % (2 * m + 1) =
    ↪ u.val)
  symm := ⟨fun _ _ ⟨hne, h⟩ => ⟨hne.symm, h.symm⟩⟩
  loopless := ⟨fun _ ⟨hne, _⟩ => hne rfl⟩
```

VizingTotal.C3uC4 (VizingTotal/AuxiliaryKempe.lean:1360-1368) — Proposition on small regular witnesses

```
def C3uC4 : SimpleGraph (Fin 7) where
  Adj u v :=
    (u.val < 3 ∧ v.val < 3 ∧ u ≠ v) ∨
    (u.val = 3 ∧ (v.val = 4 ∨ v.val = 6)) ∨
    (u.val = 4 ∧ (v.val = 3 ∨ v.val = 5)) ∨
    (u.val = 5 ∧ (v.val = 4 ∨ v.val = 6)) ∨
    (u.val = 6 ∧ (v.val = 5 ∨ v.val = 3))
  symm := ⟨by intro u v h; rcases h with ⟨h1, h2, h3⟩ | h; exact Or.inl ⟨h2, h1,
    ↪ h3.symm⟩; tauto⟩
  loopless := ⟨by intro u h; omega⟩
```

VizingTotal.CompC7 (VizingTotal/AuxiliaryKempe.lean:1193-1196) — Proposition on small regular witnesses

```
def CompC7 : SimpleGraph (Fin 7) where
  Adj u v := u ≠ v ∧ ¬ (OddCycle 3).Adj u v
  symm := ⟨fun u v ⟨hne, h⟩ => ⟨hne.symm, fun h' => h (((OddCycle 3).adj_comm v u).mp
    ↪ h')⟩⟩
  loopless := ⟨fun _ ⟨h, _⟩ => h rfl⟩
```

VizingTotal.CompC3uC4 (VizingTotal/AuxiliaryKempe.lean:1264-1270) — Proposition on small regular witnesses

```
def CompC3uC4 : SimpleGraph (Fin 7) where
  Adj u v :=
    (u.val < 3 ∧ 3 ≤ v.val) ∨ (v.val < 3 ∧ 3 ≤ u.val) ∨
    (u.val = 3 ∧ v.val = 5) ∨ (u.val = 5 ∧ v.val = 3) ∨
    (u.val = 4 ∧ v.val = 6) ∨ (u.val = 6 ∧ v.val = 4)
  symm := ⟨by intro u v h; tauto⟩
  loopless := ⟨by intro u h; omega⟩
```

VizingTotal.GDeltaVert (VizingTotal/DeltaSeparation.lean:314-317) — Graph families (Section 2)

```
inductive GDeltaVert (Δ : ℕ) : Type
  | priv (i : Fin (Δ + 1)) (r : Fin (Δ - 1))
  | term (i j : Fin (Δ + 1)) (hne : i ≠ j)
  deriving DecidableEq
```

VizingTotal.gDeltaAdj (VizingTotal/DeltaSeparation.lean:367-371) — Graph families (Section 2)

```
def gDeltaAdj {Δ : ℕ} : GDeltaVert Δ → GDeltaVert Δ → Prop
  | .priv i _, .term i' _ _ => i = i'
  | .term i' _ _, .priv i _ => i = i'
  | .term i j _, .term i' j' _ => i = j' ∧ j = i'
  | .priv _ _, .priv _ _ => False
```

VizingTotal.GDeltaStar (VizingTotal/DeltaSeparation.lean:374-385) — Graph families (Section 2)

```
def GDeltaStar (Δ : ℕ) : SimpleGraph (GDeltaVert Δ) where
  Adj := gDeltaAdj
  symm := ⟨by
    rintro (⟨i, r⟩ | ⟨i, j, h⟩) (⟨i', r'⟩ | ⟨i', j', h'⟩) hadj
    · exact hadj
    · exact hadj
    · exact hadj
    · exact ⟨hadj.2.symm, hadj.1.symm⟩⟩
  loopless := ⟨by
    rintro (⟨i, r⟩ | ⟨i, j, h⟩) hadj
    · exact hadj
    · exact h hadj.1⟩
```

VizingTotal.crossEdge (VizingTotal/DeltaSeparation.lean:932-933) — Graph families (Section 2)

```
def crossEdge {Δ : ℕ} (i j : Fin (Δ + 1)) (hne : i ≠ j) : (GDeltaStar Δ).edgeSet :=
  ⟨s(.term i j hne, .term j i hne.symm), ⟨rfl, rfl⟩⟩
```

VizingTotal.colorClass (VizingTotal/Basic.lean:288-289) — Theorem on the family G*_Delta

Context binders in scope: `variable {V : Type*} ; variable [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
def colorClass {k : ℕ} (f : TotalElem G → Fin k) (c : Fin k) : Finset (TotalElem G) :=
  Finset.univ.filter (fun x => f x = c)
```

VizingTotal.IsEquitableTotalColoring (VizingTotal/DeltaSeparation.lean:936-939) — Equitable and list colorings (Section 2)

```
def IsEquitableTotalColoring {V : Type*} [Fintype V] [DecidableEq V]
    (G : SimpleGraph V) [DecidableRel G.Adj] (k : ℕ) (f : TotalElem G → Fin k) : Prop
      ↪ :=
  IsProperTotalColoring G k f ∧
    ∀ c₁ c₂ : Fin k, (colorClass G f c₁).card ≤ (colorClass G f c₂).card + 1
```

VizingTotal.IsProperListTotalColoring (VizingTotal/DeltaSeparation.lean:1278-1280) — Equitable and list colorings (Section 2)

```
def IsProperListTotalColoring {V C : Type*} (G : SimpleGraph V)
    (L : TotalElem G → Finset C) (f : TotalElem G → C) : Prop :=
  (∀ x, f x ∈ L x) ∧ ∀ x y : TotalElem G, (TotalGraph G).Adj x y → f x ≠ f y
```

VizingTotal.HasListTotalColoring (VizingTotal/DeltaSeparation.lean:1283-1285) — Equitable and list colorings (Section 2)

```
def HasListTotalColoring {V : Type*} (G : SimpleGraph V) (k : ℕ) : Prop :=
  ∀ (C : Type) [Fintype C] [DecidableEq C] (L : TotalElem G → Finset C),
    (∀ x, (L x).card = k) → ∃ f : TotalElem G → C, IsProperListTotalColoring G L f
```

VizingTotal.KBi (VizingTotal/DeltaSeparation.lean:52-55) — Lemma on forcing

```
def KBi (a b : ℕ) : SimpleGraph (Fin a ⊕ Fin b) where
  Adj u v := (u.isLeft ∧ v.isRight) ∨ (u.isRight ∧ v.isLeft)
  symm := ⟨by intro u v h; rcases h with ⟨h1, h2⟩ | ⟨h1, h2⟩ <;> [exact Or.inr ⟨h2,
    ↪ h1⟩; exact Or.inl ⟨h2, h1⟩]⟩
  loopless := ⟨by rintro (p | t) (⟨h1, h2⟩ | ⟨h1, h2⟩) <;> simp at h1 h2⟩
```

VizingTotal.IsProperTotalColoringIn (VizingTotal/DeltaSeparation.lean:89-90) — Lemma on forcing

```
def IsProperTotalColoringIn {V C : Type*} (G : SimpleGraph V) (ψ : TotalElem G → C) :
  ↪ Prop :=
  ∀ x y : TotalElem G, (TotalGraph G).Adj x y → ψ x ≠ ψ y
```

VizingTotal.IndependentSeedPeelCertificate (VizingTotal/IndependentSeedBarrier.lean:46-54) — Peel certificates; the auxiliary class (Section 2)

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
structure IndependentSeedPeelCertificate (A : Finset V) (q : ℕ) where
  seed_indep : ∀ u ∈ A, ∀ v ∈ A, ¬ G.Adj u v
  order : List V
  nodup : order.Nodup
  covers_compl : ∀ v : V, v ∈ order ↔ v ∉ A
  peel_bound : ∀ i : Fin order.length,
    (Finset.univ.filter (fun j : Fin order.length =>
      i.val < j.val ∧ G.Adj (order.get i) (order.get j))).card <
      q - G.degree (order.get i)
```

VizingTotal.IsStrictFDegenerate (VizingTotal/IndependentSeedBarrier.lean:58-59) — Peel certificates and the auxiliary class (Section 1)

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
def IsStrictFDegenerate (U : Finset V) (f : V → ℕ) : Prop :=
  ∀ F : Finset V, F ⊆ U → F.Nonempty → ∃ v ∈ F, (F.filter (G.Adj v)).card < f v
```

VizingTotal.maxDegVerts (VizingTotal/IndependentSeedBarrier.lean:221-222) — Peel certificates (Section 1)

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
def maxDegVerts : Finset V :=
  Finset.univ.filter (fun v => G.degree v = G.maxDegree)
```

VizingTotal.CoreIsBipartite (VizingTotal/IndependentSeedBarrier.lean:225-226) — Theorem on peel certificates

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
def CoreIsBipartite : Prop :=
  ∃ c : V → Fin 2, ∀ u ∈ maxDegVerts G, ∀ v ∈ maxDegVerts G, G.Adj u v → c u ≠ c v
```

VizingTotal.IsAuxiliaryClassMember (VizingTotal/AuxiliaryKempe.lean:48-55) — Peel certificates; the auxiliary class (Section 2)

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
structure IsAuxiliaryClassMember (H : SimpleGraph V) [DecidableRel H.Adj]
    (D : ℕ) (v₀ : V) (J : Finset H.edgeSet) : Prop where
  maxDegree_bound : H.maxDegree ≤ D
  card_distinguished : J.card = D
  center_degree_lower : 2 ≤ H.degree v₀
  center_degree_upper : H.degree v₀ ≤ D
  star_subset : ∀ e : H.edgeSet, v₀ ∈ e.1 → e ∈ J
  exact_coverage : ∀ v : V, v ≠ v₀ → ∃! e : H.edgeSet, e ∈ J ∧ v ∈ e.1
```

VizingTotal.auxMatching (VizingTotal/AuxiliaryKempe.lean:58-60) — Peel certificates; the auxiliary class (Section 2)

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
def auxMatching (H : SimpleGraph V) [DecidableRel H.Adj] (v₀ : V) (J : Finset
  ↪ H.edgeSet) :
    Finset H.edgeSet :=
  J.filter (fun e => v₀ ∉ e.1)
```

VizingTotal.IsProperEdgeColoring (VizingTotal/AuxiliaryKempe.lean:158-159) — Peel certificates; the auxiliary class (Section 2)

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
def IsProperEdgeColoring (H : SimpleGraph V) (k : ℕ) (c : H.edgeSet → Fin k) : Prop :=
  ∀ e₁ e₂ : H.edgeSet, e₁ ≠ e₂ → (∃ v : V, v ∈ e₁.1 ∧ v ∈ e₂.1) → c e₁ ≠ c e₂
```

VizingTotal.IsRainbowOn (VizingTotal/AuxiliaryKempe.lean:162-163) — Peel certificates; the auxiliary class (Section 2)

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
def IsRainbowOn (H : SimpleGraph V) (k : ℕ) (J : Finset H.edgeSet) (c : H.edgeSet →
  ↪ Fin k) : Prop :=
  ∀ e₁ ∈ J, ∀ e₂ ∈ J, c e₁ = c e₂ → e₁ = e₂
```

VizingTotal.IsKempeClosed (VizingTotal/AuxiliaryKempe.lean:250-254) — Peel certificates; the auxiliary class (Section 2)

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
def IsKempeClosed (H : SimpleGraph V) (k : ℕ) (c : H.edgeSet → Fin k)
    (a b : Fin k) (S : Finset H.edgeSet) : Prop :=
  (∀ e ∈ S, c e = a ∨ c e = b) ∧
  ∀ e₁ ∈ S, ∀ e₂ : H.edgeSet, e₁ ≠ e₂ → (∃ v : V, v ∈ e₁.1 ∧ v ∈ e₂.1) →
    (c e₂ = a ∨ c e₂ = b) → e₂ ∈ S
```

VizingTotal.kempeFlip (VizingTotal/AuxiliaryKempe.lean:257-259) — Peel certificates; the auxiliary class (Section 2)

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
def kempeFlip {H : SimpleGraph V} {k : ℕ} [DecidableEq H.edgeSet]
    (c : H.edgeSet → Fin k) (a b : Fin k) (S : Finset H.edgeSet) (e : H.edgeSet) :
      ↪ Fin k :=
  if e ∈ S then (if c e = a then b else a) else c e
```

VizingTotal.auxCexGraph (VizingTotal/AuxiliaryKempe.lean:295-303) — Theorem on Kempe distance two

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
def auxCexGraph : SimpleGraph (Fin 5) where
  Adj u v :=
    (u.val = 0 ∧ (v.val = 1 ∨ v.val = 2)) ∨
    (v.val = 0 ∧ (u.val = 1 ∨ u.val = 2)) ∨
    (u.val = 3 ∧ v.val = 4) ∨ (u.val = 4 ∧ v.val = 3) ∨
    ((u.val = 1 ∨ u.val = 2) ∧ (v.val = 3 ∨ v.val = 4)) ∨
    ((v.val = 1 ∨ v.val = 2) ∧ (u.val = 3 ∨ u.val = 4))
  symm := ⟨by intro u v h; tauto⟩
  loopless := ⟨by intro u h; omega⟩
```

VizingTotal.auxCexJ (VizingTotal/AuxiliaryKempe.lean:323-323) — Theorem on Kempe distance two

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
def auxCexJ : Finset auxCexGraph.edgeSet := {cexE01, cexE02, cexE34}
```

VizingTotal.auxCexPsi (VizingTotal/AuxiliaryKempe.lean:337-344) — Theorem on Kempe distance two

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
def auxCexPsi (e : auxCexGraph.edgeSet) : Fin 4 :=
  if e = cexE01 then 0
  else if e = cexE02 then 1
  else if e = cexE34 then 0
  else if e = cexE13 then 2
  else if e = cexE14 then 3
  else if e = cexE23 then 3
  else 2
```

VizingTotal.kbiEdge (VizingTotal/DeltaSeparation.lean:61-62) — Lemma on forcing

```
def kbiEdge {a b : ℕ} (p : Fin a) (t : Fin b) : (KBi a b).edgeSet :=
  ⟨s(Sum.inl p, Sum.inr t), Or.inl ⟨rfl, rfl⟩⟩
```

VizingTotal.cexE01 (VizingTotal/AuxiliaryKempe.lean:314-314) — Theorem on Kempe distance two

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
def cexE01 : auxCexGraph.edgeSet := ⟨s(0, 1), by decide⟩
```

VizingTotal.cexE02 (VizingTotal/AuxiliaryKempe.lean:315-315) — Theorem on Kempe distance two

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
def cexE02 : auxCexGraph.edgeSet := ⟨s(0, 2), by decide⟩
```

VizingTotal.cexE13 (VizingTotal/AuxiliaryKempe.lean:317-317) — Theorem on Kempe distance two

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
def cexE13 : auxCexGraph.edgeSet := ⟨s(1, 3), by decide⟩
```

VizingTotal.cexE14 (VizingTotal/AuxiliaryKempe.lean:318-318) — Theorem on Kempe distance two

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
def cexE14 : auxCexGraph.edgeSet := ⟨s(1, 4), by decide⟩
```

VizingTotal.cexE23 (VizingTotal/AuxiliaryKempe.lean:319-319) — Theorem on Kempe distance two

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
def cexE23 : auxCexGraph.edgeSet := ⟨s(2, 3), by decide⟩
```

VizingTotal.cexE34 (VizingTotal/AuxiliaryKempe.lean:316-316) — Theorem on Kempe distance two

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
def cexE34 : auxCexGraph.edgeSet := ⟨s(3, 4), by decide⟩
```

VizingTotal.RungState.top (VizingTotal/MobiusFrobenius.lean:862-865) — Mobius ladders and prisms (Section 3)

```
def RungState.top : RungState → Nat
  | .s00 | .s01 | .s02 => 0
  | .s10 | .s12 => 1
  | .s20 | .s21 | .s22 => 2
```

VizingTotal.RungState.bot (VizingTotal/MobiusFrobenius.lean:868-871) — Mobius ladders and prisms (Section 3)

```
def RungState.bot : RungState → Nat
  | .s00 | .s10 | .s20 => 0
  | .s01 | .s21 => 1
  | .s02 | .s12 | .s22 => 2
```

VizingTotal.indepVerts (VizingTotal/Basic.lean:159-160) — Total colorings (Section 2)

Context binders in scope: `variable {V : Type*} ; variable [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
def indepVerts (I : Finset (TotalElem G)) : Finset V :=
  Finset.univ.filter (fun v : V => Sum.inl v ∈ I)
```

VizingTotal.indepEdges (VizingTotal/Basic.lean:163-164) — Total colorings (Section 2)

Context binders in scope: `variable {V : Type*} ; variable [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
def indepEdges (I : Finset (TotalElem G)) : Finset G.edgeSet :=
  Finset.univ.filter (fun e : G.edgeSet => Sum.inr e ∈ I)
```

Selected Concrete Sanity Examples (checked in Lean; examples only, not global identity proofs): VizingTotal.sanity_mobiusChiF_five (mobiusChiF 5 = 25 / 6); VizingTotal.sanity_mobiusChiF_ten (mobiusChiF 10 = 4); VizingTotal.sanity_card_gDeltaVert_three (Fintype.card (GDeltaVert 3) = 20); VizingTotal.sanity_alphaTotal_mobius_four (alphaTotal (MobiusLadder 4) = 5).

Box 2 — Verbatim Main Theorem Signatures, Context Binders & Applicability

VizingTotal.paper_mobius (VizingTotal/PaperDraft1.lean:169-178) src:c2b17b7b7001a7

```
theorem paper_mobius {k : ℕ} (hk : 2 ≤ k) :
    alphaTotal (MobiusLadder k) = (4 * k - 1) / 3 ∧
    ((∃ P : FractionalTotalColoring (MobiusLadder k), P.totalWeight = mobiusChiF k) ∧
      (∀ P : FractionalTotalColoring (MobiusLadder k), mobiusChiF k ≤ P.totalWeight))
        ↪ ∧
    (4 < mobiusChiF k ↔ k ∈ ({2, 3, 5, 6, 9} : Finset ℕ)) ∧
    ((∃ a b : ℕ, 1 ≤ a ∧ k = 4 * a + 3 * b) ↔ k ∉ ({2, 3, 5, 6, 9} : Finset ℕ)) ∧
    IsType2 (MobiusLadder k)
```

Assumptions: k is a natural number with 2 <= k; MobiusLadder k is the graph on Fin (2k) of Section 2, and mobiusChiF k is the function m(k) of the statement. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

VizingTotal.paper_peel (VizingTotal/PaperDraft1.lean:210-223) src:f95fc1af2dded1

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
theorem paper_peel :
    (∀ A : Finset V, (∀ u ∈ A, ∀ v ∈ A, ¬ G.Adj u v) → ∀ q : ℕ,
      (Nonempty (IndependentSeedPeelCertificate G A q) ↔
        IsStrictFDegenerate G Aᶜ (fun v => q - G.degree v))) ∧
    (∀ A : Finset V, Nonempty (IndependentSeedPeelCertificate G A (G.maxDegree + 1)) →
      CoreIsBipartite G) ∧
    (∀ (m : ℕ) (cyc : Fin (2 * m + 1) → V),
      (∀ i, cyc i ∈ maxDegVerts G) →
      (∀ i : Fin (2 * m + 1), G.Adj (cyc i)
        (cyc ⟨(i.val + 1) % (2 * m + 1), Nat.mod_lt _ (by omega)⟩)) →
      ∀ A : Finset V, IsEmpty (IndependentSeedPeelCertificate G A (G.maxDegree + 1)))
```

Assumptions: G is a simple graph on a finite vertex type with decidable adjacency. Part (i): A is a finset of pairwise non-adjacent vertices and q is any natural number. Part (ii) assumes a certificate for a finset A at Delta(G) + 1. Part (iii): m is any natural number and cyc maps Fin (2m + 1) to vertices of maximum degree, each adjacent to the next modulo 2m + 1; the conclusion ranges over every finset A. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

VizingTotal.kempe_dist_eq_two_counterexample (VizingTotal/AuxiliaryKempe.lean:463-480) src:bcad0c6a19f63a

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
theorem kempe_dist_eq_two_counterexample :
    Fintype.card (Fin 5) = 5 ∧
    auxCexGraph.edgeFinset.card = 7 ∧
    IsAuxiliaryClassMember auxCexGraph 3 0 auxCexJ ∧
    IsProperEdgeColoring auxCexGraph 4 auxCexPsi ∧
    ¬ IsRainbowOn auxCexGraph 4 auxCexJ auxCexPsi ∧
    (∀ (a b : Fin 4) (S : Finset auxCexGraph.edgeSet),
      IsKempeClosed auxCexGraph 4 auxCexPsi a b S →
      ¬ IsRainbowOn auxCexGraph 4 auxCexJ (kempeFlip auxCexPsi a b S)) ∧
    (∃ (a₁ b₁ a₂ b₂ : Fin 4) (S₁ S₂ : Finset auxCexGraph.edgeSet),
      IsKempeClosed auxCexGraph 4 auxCexPsi a₁ b₁ S₁ ∧
      let ψ₁ := kempeFlip auxCexPsi a₁ b₁ S₁
      IsKempeClosed auxCexGraph 4 ψ₁ a₂ b₂ S₂ ∧
      let ψ₂ := kempeFlip ψ₁ a₂ b₂ S₂
      IsProperEdgeColoring auxCexGraph 4 ψ₂ ∧
      IsRainbowOn auxCexGraph 4 auxCexJ ψ₂)
```

Assumptions: No binders: the graph auxCexGraph, the set auxCexJ and the coloring auxCexPsi are fixed. In the one-step clause a, b range over Fin 4 and S must be Kempe-closed for psi and colors a, b, and each of the two witnessed swaps is closed for its own colors in the coloring immediately before it. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

VizingTotal.paper_gDeltaStar (VizingTotal/PaperDraft1.lean:293-321) src:190c2e8d033c16

```
theorem paper_gDeltaStar {Δ : ℕ} (hΔ : 3 ≤ Δ) :
    Fintype.card (GDeltaVert Δ) = (Δ + 1) * (2 * Δ - 1) ∧
    (GDeltaStar Δ).IsRegularOfDegree Δ ∧
    (GDeltaStar Δ).Connected ∧
    (∀ u v w : GDeltaVert Δ, ¬ ((GDeltaStar Δ).Adj u v ∧ (GDeltaStar Δ).Adj v w ∧
```

```
      (GDeltaStar Δ).Adj u w)) ∧
    (∃ a b c d : GDeltaVert Δ, a ≠ c ∧ b ≠ d ∧
      (GDeltaStar Δ).Adj a b ∧ (GDeltaStar Δ).Adj b c ∧
      (GDeltaStar Δ).Adj c d ∧ (GDeltaStar Δ).Adj d a) ∧
    IsType1 (GDeltaStar Δ) ∧
    FracTotalEq (GDeltaStar Δ) ((Δ : ℚ) + 1) ∧
    (∀ φ : TotalElem (GDeltaStar Δ) → Fin (Δ + 1),
      IsProperTotalColoring (GDeltaStar Δ) (Δ + 1) φ →
      ∃ cStar : Fin (Δ + 1),
        (∀ i r, φ (Sum.inl (.priv i r)) = cStar) ∧
        (∀ i j hne, φ (Sum.inr (crossEdge i j hne)) = cStar) ∧
        2 * (colorClass (GDeltaStar Δ) φ cStar).card = (Δ + 1) * (3 * Δ - 2) ∧
        (∀ γ : Fin (Δ + 1), γ ≠ cStar →
          (colorClass (GDeltaStar Δ) φ γ).card = Δ * (Δ + 1)) ∧
        (∀ γ : Fin (Δ + 1), γ ≠ cStar →
          (colorClass (GDeltaStar Δ) φ γ).card + 2 ≤
            (colorClass (GDeltaStar Δ) φ cStar).card)) ∧
    (¬ ∃ φ : TotalElem (GDeltaStar Δ) → Fin (Δ + 1),
      IsEquitableTotalColoring (GDeltaStar Δ) (Δ + 1) φ) ∧
    ¬ HasListTotalColoring (GDeltaStar Δ) (Δ + 1)
```

Assumptions: Delta is a natural number with 3 <= Delta; phi is any total coloring with values in Fin (Delta + 1) that is proper. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

VizingTotal.paper_clique (VizingTotal/PaperDraft1.lean:74-79) `src:d5fbbf75b2dd0f`

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
theorem paper_clique [Nonempty V] :
    (G.maxDegree = 1 → omegaTotal G = 3) ∧
    (G.maxDegree ≠ 1 → omegaTotal G = G.maxDegree + 1) ∧
    ∀ P : FractionalTotalColoring G, (omegaTotal G : ℚ) ≤ P.totalWeight
```

Assumptions: G is a simple graph on a finite, nonempty vertex type; P is any fractional total coloring of G. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

VizingTotal.paper_deficiency_dual (VizingTotal/PaperDraft1.lean:82-93) `src:71b0f2f290f3cf`

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
theorem paper_deficiency_dual [Nonempty V] :
    (∀ P : FractionalTotalColoring G,
      1 + (2 * (G.edgeFinset.card : ℚ)) / (Fintype.card V : ℚ) ≤ P.totalWeight ∧
      (G.maxDegree : ℚ) + 1 - (deficiency G : ℚ) / (Fintype.card V : ℚ) ≤
        ↪ P.totalWeight) ∧
    ∀ r : ℕ, G.IsRegularOfDegree r → ∀ P : FractionalTotalColoring G,
      ((r : ℚ) + 1 ≤ P.totalWeight) ∧
      (P.totalWeight = (r : ℚ) + 1 ↔
        (∀ x : TotalElem G,
          (∑ I : Finset (TotalElem G), if x ∈ I then P.w I else 0) = 1) ∧
        (∀ I : Finset (TotalElem G), P.w I ≠ 0 → IsStarSaturated G I))
```

Assumptions: G is a simple graph on a finite, nonempty vertex type; P is any fractional total coloring; in the second part r is a natural number and G is r-regular. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

VizingTotal.paper_conformable (VizingTotal/PaperDraft1.lean:96-97) `src:52a4a479bad70d`

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
theorem paper_conformable (ht1 : IsType1 G) : IsConformable G
```

Assumptions: G is a simple graph on a finite vertex type that has a proper (Delta + 1)-total coloring. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

VizingTotal.paper_bipartite (VizingTotal/PaperDraft1.lean:135-150) `src:dc42259bdfcf08`

```
theorem paper_bipartite :
    (∀ (n k : ℕ) (_ : k ≤ n) (G : SimpleGraph (Fin n ⊕ Fin n)) [DecidableRel G.Adj],
      IsBipartiteParts G → G.IsRegularOfDegree (n - k) →
      (∀ I : Finset (TotalElem G), IsTotalIndep G I → I.card ≤ n + k) ∧
      (1 ≤ n → ∀ P : FractionalTotalColoring G,
        ((n * (n - k + 2) : ℕ) : ℚ) / ((n + k : ℕ) : ℚ) ≤ P.totalWeight)) ∧
    (∀ n : ℕ, 1 ≤ n →
      IsType2 (CompleteBipartite n) ∧ FracTotalEq (CompleteBipartite n) ((n : ℚ) +
        ↪ 2)) ∧
    (IsType2 (CrownGraph 2) ∧ FracTotalEq (CrownGraph 2) 3) ∧
    (∀ n : ℕ, 3 ≤ n → IsType1 (CrownGraph n) ∧ FracTotalEq (CrownGraph n) (n : ℚ))
```

Assumptions: n and k are natural numbers with k <= n; G is a simple graph on Fin n + Fin n whose edges join the two copies, and G is (n - k)-regular. I is any total matching, and the bound for P assumes n >= 1. The complete bipartite and crown clauses have independent parameters, respectively n >= 1, n = 2, and n >= 3. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

VizingTotal.paper_complete (VizingTotal/PaperDraft1.lean:153-156) src:522d396294fc87

```
theorem paper_complete :
    (∀ r : ℕ, IsType1 (⊤ : SimpleGraph (Fin (2 * r + 1)))) ∧
    (∀ r : ℕ, 1 ≤ r → IsType2 (⊤ : SimpleGraph (Fin (2 * r))))
```

Assumptions: r is a natural number, with 1 <= r in the second part. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

VizingTotal.paper_chiral_table (VizingTotal/PaperDraft1.lean:159-166) src:e43ed2462679ba

```
theorem paper_chiral_table :
    allLegalTransitions.length = 39 ∧
    (allLegalTransitions.map (fun t => (t.src, t.dst))).dedup.length = 38 ∧
    ((allLegalTransitions.filter (fun t => t.reducedWeight == 0)).map
      (fun t => (t.src, t.dst))).dedup.length = 11 ∧
    ∀ t : RungTransition, t.isLegal = true → t.reducedWeight ≤ 0
```

Assumptions: No binders except t, any transition datum satisfying t.isLegal = true in the weight bound; the table allLegalTransitions is a fixed finite list. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

VizingTotal.paper_small_regular (VizingTotal/PaperDraft1.lean:181-203) src:4f5a448de3b0dd

```
theorem paper_small_regular :
    ((MobiusLadder 4).edgeFinset.card = 12 ∧
      (MobiusLadder 4).IsRegularOfDegree 3 ∧ IsConformable (MobiusLadder 4) ∧
      alphaTotal (MobiusLadder 4) = 5 ∧ FracTotalEq (MobiusLadder 4) 4 ∧
      IsType2 (MobiusLadder 4)) ∧
    (PrismL10.edgeFinset.card = 15 ∧ PrismL10.IsRegularOfDegree 3 ∧
      IsConformable PrismL10 ∧ alphaTotal PrismL10 = 6 ∧ FracTotalEq PrismL10 (25 /
        ↪ 6) ∧
      IsType2 PrismL10) ∧
    (∀ (G : SimpleGraph (Fin 7)) [DecidableRel G.Adj] {r : ℕ},
      G.IsRegularOfDegree r → r = 0 ∨ r = 2 ∨ r = 4 ∨ r = 6) ∧
    (IsType1 (⊥ : SimpleGraph (Fin 7)) ∧
      (∀ P : FractionalTotalColoring (OddCycle 3), (7 / 2 : ℚ) ≤ P.totalWeight) ∧
      (∀ P : FractionalTotalColoring C3uC4, (4 : ℚ) ≤ P.totalWeight) ∧
      (∀ P : FractionalTotalColoring CompC7, (21 / 4 : ℚ) ≤ P.totalWeight) ∧
      ¬ HasProperTotalColoring CompC7 5 ∧
      IsType1 CompC3uC4 ∧ IsType1 (⊤ : SimpleGraph (Fin 7)))
```

Assumptions: The degree clause quantifies over G on Fin 7 and an implicit natural number r with G r-regular. Each fractional bound quantifies over P on the indicated graph. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

VizingTotal.paper_peel_regular (VizingTotal/PaperDraft1.lean:226-234) src:ad129d477fcb80

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V] (G : SimpleGraph V) [DecidableRel G.Adj]`

```
theorem paper_peel_regular [Nonempty V] {r : ℕ} (hreg : G.IsRegularOfDegree r) :
    ((∃ A : Finset V, Nonempty (IndependentSeedPeelCertificate G A (r + 1))) ↔
      Nonempty (G.Coloring (Fin 2))) ∧
    (G.Connected → ∀ (col : G.Coloring (Fin 2)) (A : Finset V),
      (Nonempty (IndependentSeedPeelCertificate G A (r + 1)) ↔
        (A = Finset.univ.filter (fun v => col v = ⟨0, by omega⟩) ∨
         A = Finset.univ.filter (fun v => col v = ⟨1, by omega⟩)))))
```

Assumptions: G is a simple graph on a finite, nonempty vertex type that is r-regular. The second equivalence additionally assumes G is connected and ranges over every proper 2-coloring col and every finset A. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

VizingTotal.paper_peel_odd_cycle_complete (VizingTotal/PaperDraft1.lean:239-246) src:167e7e26713fff

```
theorem paper_peel_odd_cycle_complete :
    (∀ m : ℕ, 1 ≤ m → ∀ A : Finset (Fin (2 * m + 1)),
      IsEmpty (IndependentSeedPeelCertificate (OddCycle m) A ((OddCycle m).maxDegree
        ↪ + 1))) ∧
    (∀ n : ℕ, 3 ≤ n → ∀ A : Finset (Fin n),
      IsEmpty (IndependentSeedPeelCertificate (⊤ : SimpleGraph (Fin n)) A
        ((⊤ : SimpleGraph (Fin n)).maxDegree + 1)))
```

Assumptions: m is a natural number with 1 <= m, and n a natural number with 3 <= n; A is any finset of vertices. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

VizingTotal.paper_aux_counts (VizingTotal/PaperDraft1.lean:253-260) src:1ce32d97b4bad1

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
theorem paper_aux_counts {H : SimpleGraph V} [DecidableRel H.Adj] {D : ℕ} {v₀ : V}
    {J : Finset H.edgeSet} (hmem : IsAuxiliaryClassMember H D v₀ J) :
    H.degree v₀ + (auxMatching H v₀ J).card = D ∧
    Fintype.card V = 1 + H.degree v₀ + 2 * (auxMatching H v₀ J).card ∧
    ((auxMatching H v₀ J).card = 0 → ∀ (k : ℕ) (c : H.edgeSet → Fin k),
      IsProperEdgeColoring H k c → IsRainbowOn H k J c)
```

Assumptions: H is a simple graph on a finite vertex type; (H, J) is a member of the auxiliary class with parameter D and center v_0. In the rainbow clause |M| = 0, k is any natural number, and c is any proper k-edge coloring. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

VizingTotal.paper_aux_monotone (VizingTotal/PaperDraft1.lean:263-272) src:4f0711f5c73719

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
theorem paper_aux_monotone {H₁ H₂ : SimpleGraph V} [DecidableRel H₁.Adj]
    [DecidableRel H₂.Adj] {D k : ℕ} {v₀ : V} {J₁ : Finset H₁.edgeSet}
    (hle : H₁ ≤ H₂) (hmem₁ : IsAuxiliaryClassMember H₁ D v₀ J₁)
    (hΔ₂ : H₂.maxDegree ≤ D) (hstar_eq : ∀ w : V, H₂.Adj v₀ w → H₁.Adj v₀ w) :
    let liftE : H₁.edgeSet → H₂.edgeSet := fun e => ⟨e.1, SimpleGraph.edgeSet_mono
      ↪ hle e.2⟩
    let J₂ : Finset H₂.edgeSet := J₁.image liftE
    IsAuxiliaryClassMember H₂ D v₀ J₂ ∧
      ((∃ c₂ : H₂.edgeSet → Fin k, IsProperEdgeColoring H₂ k c₂ ∧ IsRainbowOn H₂ k J₂
        ↪ c₂) →
       (∃ c₁ : H₁.edgeSet → Fin k, IsProperEdgeColoring H₁ k c₁ ∧ IsRainbowOn H₁ k J₁
         ↪ c₁))
```

Assumptions: H_1 <= H_2 are simple graphs on one finite vertex type; (H_1, J_1) is a member with parameter D and center v_0; the maximum degree of H_2 is at most D; every H_2-neighbor of v_0 is an H_1-neighbor; k is any natural number. The coloring implication assumes that H_2 has a proper k-edge coloring rainbow on the lifted J. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

VizingTotal.k_pred_succ_monochromatic_forcing (VizingTotal/DeltaSeparation.lean:94-308) src:8f3af1e5965aed

```
theorem k_pred_succ_monochromatic_forcing {Δ : ℕ} (hΔ : 2 ≤ Δ)
    {C : Type*} [Fintype C] [DecidableEq C] (hC : Fintype.card C = Δ + 1)
    {ψ : TotalElem (KBi (Δ - 1) Δ) → C} (hψ : IsProperTotalColoringIn (KBi (Δ - 1) Δ)
      ↪ ψ) :
    ∃ cStar : C,
      (∀ p : Fin (Δ - 1), ψ (Sum.inl (Sum.inl p)) = cStar) ∧
      (∀ t : Fin Δ, ψ (Sum.inl (Sum.inr t)) ≠ cStar) ∧
      Function.Injective (fun t : Fin Δ => ψ (Sum.inl (Sum.inr t))) ∧
      (∀ γ : C, γ ≠ cStar → ∃ t : Fin Δ, ψ (Sum.inl (Sum.inr t)) = γ) ∧
      (∀ e : (KBi (Δ - 1) Δ).edgeSet, ψ (Sum.inr e) ≠ cStar) ∧
      (∀ (t : Fin Δ) (γ : C), γ ≠ cStar → γ ≠ ψ (Sum.inl (Sum.inr t)) →
        ∃ p : Fin (Δ - 1), ψ (Sum.inr (kbiEdge p t)) = γ) ∧
      (∀ γ : C, γ ≠ cStar →
        (Finset.univ.filter (fun e : (KBi (Δ - 1) Δ).edgeSet => ψ (Sum.inr e) =
          ↪ γ)).card = Δ - 1)
```

Assumptions: Delta is a natural number with 2 <= Delta; C is a finite type with Delta + 1 elements; psi is a proper total coloring of KBi (Delta - 1) Delta with values in C. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

VizingTotal.gDeltaStar_three_profile (VizingTotal/DeltaSeparation.lean:1261-1273) src:eb9e8edd01a287

```
theorem gDeltaStar_three_profile :
    Fintype.card (GDeltaVert 3) = 20 ∧
    (GDeltaStar 3).edgeFinset.card = 30 ∧
    ∀ {φ : TotalElem (GDeltaStar 3) → Fin 4}, IsProperTotalColoring (GDeltaStar 3) 4
      ↪ φ →
      ∃ cStar : Fin 4, (colorClass (GDeltaStar 3) φ cStar).card = 14 ∧
        ∀ γ ≠ cStar, (colorClass (GDeltaStar 3) φ γ).card = 12
```

Assumptions: No binders except phi, any proper total coloring of G*_3 with values in Fin 4. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

Table 2. Lean 4 Type-Trap Audit Checklist — Source-Level Approximation (Gate Replay Pending).[a]

Declaration	Hazard	Operand	Category	Audited Disposition & Guard
`GDeltaVert`	`nat_sub`	`- 1`	intended_convention	The private vertices of a block are indexed by Fin (Delta - 1). Every theorem about G*_Delta assumes 3 <= Delta, so Delta - 1 is the ordinary difference and each block has Delta - 1 private vertices, as in Section 2.
`IndependentSeedPeel Certificate`	`nat_sub`	`- G.degree`	intended_convention	The bound q - d_G(u_i) is subtraction truncated at 0, exactly as the definition in Section 2 and Theorem on peel certificates state it: when d_G(u_i) >= q no vertex can be peeled at u_i.
`IsConformable`	`nat_mod`	`% 2`	intended_convention	Parity of a class size and of |V|; the modulus 2 is not zero.
`MobiusLadder`	`nat_mod`	`% (2 * k)`	intended_convention	Indices are taken modulo 2k on Fin (2k). For k = 0 the vertex type is empty; every theorem assumes 2 <= k, so the modulus is positive.

Declaration	Hazard	Operand	Category	Audited Disposition & Guard
OddCycle	nat_mod	% (2 * m + 1)	intended_convention	The modulus 2m + 1 is positive for every m.
PrismL10	nat_mod	% 5	intended_convention	Indices along each 5-cycle of the prism Y_5 are taken modulo 5.
PrismL10	nat_sub	- 5	intended_convention	The differences x - 5 and y - 5 occur only in the conjunct that first requires 5 <= x and 5 <= y, so they are ordinary differences (the second 5-cycle).
deficiency	nat_sub	- G.degree	intended_convention	Every degree is at most the maximum degree, so Delta(G) - d_G(v) is the ordinary difference (degree_le_max Degree).
k_pred_succ_monochromatic_forcing	nat_sub	- 1	intended_convention	The graph is K_{Delta-1,Delta} with the hypothesis 2 <= Delta, so Delta - 1 is the ordinary difference.
mobiusChiF	nat_div	/ 6	intended_convention	25 / 6 is a literal of type Q (mobiusChiF has values in Q); this is exact rational division, not natural-number division.
mobiusChiF	nat_div	/ 7	intended_convention	30 / 7 is a literal of type Q; exact rational division.
mobiusChiF	nat_div	/ 11	intended_convention	45 / 11 is a literal of type Q; exact rational division.
paper_bipartite	nat_sub	- k	intended_convention	The degree n - k is formed under the hypothesis k <= n of the same statement, so it is the ordinary difference.
paper_bipartite	nat_div	/ ((n + k : ℕ) : ℚ)	intended_convention	The division is in Q, after casting; it is used only under the hypothesis 1 <= n, so the divisor n + k is positive.
paper_deficiency_dual	nat_div	/ (Fintype.card V : ℚ)	intended_convention	The division is in Q; the statement assumes the vertex type is nonempty, so |V| >= 1.
paper_deficiency_dual	nat_sub	- (deficiency G : ℚ)	intended_convention	The subtraction Delta(G) + 1 - def(G)/n is taken in Q after casting, so nothing is truncated.
paper_gDeltaStar	nat_sub	- 2	intended_convention	The natural-number expressions 3 Delta - 2 and 2 Delta - 1 occur under the hypothesis 3 <= Delta, so both are ordinary differences.
paper_mobius	nat_sub	- 1	intended_convention	4k - 1 is formed under the hypothesis 2 <= k, so it is the ordinary difference.
paper_mobius	nat_div	/ 3	intended_convention	(4k - 1) / 3 is natural-number division, the floor of (4k - 1)/3, which is the value the theorem states.
paper_peel	nat_sub	- G.degree	intended_convention	q - d_G(v) is subtraction truncated at 0, as part (i) of the theorem states.
paper_peel	nat_mod	% (2 * m + 1)	intended_convention	The closed walk is indexed by Fin (2m + 1) and the successor of i is (i + 1) mod (2m + 1); the modulus is positive.
paper_small_regular	nat_div	/ 6	intended_convention	25 / 6 is a literal of type Q; exact rational division.
paper_small_regular	nat_div	/ 2	intended_convention	7 / 2 is a literal of type Q; exact rational division.
paper_small_regular	nat_div	/ 4	intended_convention	21 / 4 is a literal of type Q; exact rational division.
instDecidableRelAuxCexGraph	local_instance_or_notation	instDecidableRelAuxCexGraph	intended_convention	Decidability instance for adjacency of the graph H of the theorem on Kempe distance two, defined by unfolding the definition; a decidability instance is a subsingleton, so no statement depends on which instance is used.
instDecidableRelPrismL10	local_instance_or_notation	instDecidableRelPrismL10	intended_convention	Decidability instance for adjacency of the prism Y_5, defined by unfolding the definition; a decidability instance is a subsingleton, so no statement depends on which instance is used.
instDecidableRelCompC7	local_instance_or_notation	instDecidableRelCompC7	intended_convention	Decidability instance for adjacency of the complement of C_7, defined by unfolding the definition; a decidability instance is a subsingleton, so no statement depends on which instance is used.

Declaration	Hazard	Operand	Category	Audited Disposition & Guard
instDecidableRelCompC3uC4	local_instance_or_notation	instDecidableRelCompC3uC4	intended_convention	Decidability instance for adjacency of the complement of C_3 + C_4, defined by unfolding the definition; a decidability instance is a subsingleton, so no statement depends on which instance is used.
instDecidableRelC3uC4	local_instance_or_notation	instDecidableRelC3uC4	intended_convention	Decidability instance for adjacency of C_3 + C_4, defined by unfolding the definition; a decidability instance is a subsingleton, so no statement depends on which instance is used.
instDecidableRelTotalGraph	local_instance_or_notation	instDecidableRelTotalGraph	intended_convention	Decidability instance for adjacency of the total graph T(G), defined by unfolding the definition; a decidability instance is a subsingleton, so no statement depends on which instance is used.
instDecidablePredIsTotalIndep	local_instance_or_notation	instDecidablePredIsTotalIndep	intended_convention	Decidability instance for the predicate of being a total matching, defined by unfolding the definition; a decidability instance is a subsingleton, so no statement depends on which instance is used.
instDecidablePredIsTotalClique	local_instance_or_notation	instDecidablePredIsTotalClique	intended_convention	Decidability instance for the predicate of being a clique of T(G), defined by unfolding the definition; a decidability instance is a subsingleton, so no statement depends on which instance is used.
instDecidableRelCompleteBipartite	local_instance_or_notation	instDecidableRelCompleteBipartite	intended_convention	Decidability instance for adjacency of K_{n,n}, defined by unfolding the definition; a decidability instance is a subsingleton, so no statement depends on which instance is used.
instDecidableRelCrownGraph	local_instance_or_notation	instDecidableRelCrownGraph	intended_convention	Decidability instance for adjacency of the crown Cr_n, defined by unfolding the definition; a decidability instance is a subsingleton, so no statement depends on which instance is used.
instDecidableRelKBi	local_instance_or_notation	instDecidableRelKBi	intended_convention	Decidability instance for adjacency of K_{a,b}, defined by unfolding the definition; a decidability instance is a subsingleton, so no statement depends on which instance is used.
instDecidableRelGDeltaStar	local_instance_or_notation	instDecidableRelGDeltaStar	intended_convention	Decidability instance for adjacency of G*_Delta, defined by unfolding the definition; a decidability instance is a subsingleton, so no statement depends on which instance is used.
instDecidableRelOddCycle	local_instance_or_notation	instDecidableRelOddCycle	intended_convention	Decidability instance for adjacency of the cycle C_{2m+1}, defined by unfolding the definition; a decidability instance is a subsingleton, so no statement depends on which instance is used.
instDecidableRelMobiusLadder	local_instance_or_notation	instDecidableRelMobiusLadder	intended_convention	Decidability instance for adjacency of M_2k, defined by unfolding the definition; a decidability instance is a subsingleton, so no statement depends on which instance is used.
instFintypeGDeltaVert	local_instance_or_notation	instFintypeGDeltaVert	intended_convention	Fintype instance for the vertices of G*_Delta, built from an explicit equivalence; Fintype is a subsingleton, so no cardinality depends on the instance chosen.

[a] Covers the specified Lean 4 hazard set (`Nat` subtraction/division/mod, totalized real/rational division, `Real.log`/`rpow`, `Finset.range`, `Classical.choose`, local instances); no finite checklist certifies all semantics.

Table 3. Per-Claim Reader's Trust Boundary Ledger (G11 Denominators Derived Mechanically).[a]

Paper Claim	Proof Tier	Lean Decl / Artifact	Actual Axioms & Fidelity	Assumptions & What You Must Trust
Thm. 1: Let k >= 2 and let M_2k be the Mobius ladder on Z_2k, in which x is adjacent to x + 1, x - 1 and x + k. Then (i) the largest total matching of M_2k has floor((4k - 1)/3) elements; (ii) some fractional total coloring of M_2k has weight m(k) and every one has weight at least m(k), where m(k) = 5 for k in {2, 3}, m(5) = 25/6, m(6) = 30/7, m(9) = 45/11 and m(k) = 4 for every other k; and m(k) > 4 exactly when k is in {2, 3, 5, 6, 9}; (iii) k is not in {2, 3, 5, 6, 9} exactly when k = 4a + 3b for some integers a >= 1 and b >= 0; (iv) M_2k is of Type 2	Lean Kernel	`VizingTotal.paper_mobius`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel; axioms propext, Classical.choice, Quot.sound; definitions of Section 2; hypotheses as stated.
Thm. 2: Let n >= 3 and let Y_n be the prism C_n x K_2. Its largest total matching has floor(4n/3) elements. If n != 5 then the fractional total chromatic number and the total chromatic number of Y_n both equal 4, so Y_n is of Type 1; Y_5 has fractional total chromatic number 25/6 and total chromatic number 5. In particular the ceiling of the fractional total chromatic number equals the total chromatic number for every n >= 3	Written	recompute.py (7a9a48079da8) [regenerable; replay pending]	External cert only; fidelity: n/a	Written proof (rung-state bound, tiles T_3 and T_4); replay: tiles and the block matching for 3 <= n <= 60; Y_5 values from the Lean claim prop_smallregular
Thm. 3: Let n >= 3, let F be a 2-factor of K_{n,n} that contains no 4-cycle, and let G = K_{n,n} - F. Then G is of Type 2, its total chromatic number is n = Delta(G) + 2, every total matching of G has at most n + 1 elements, and its fractional total chromatic number is at least n^2/(n + 1)	Written	Written proof	Lean: none; fidelity: n/a	Written proof (counting and pigeonhole), with Konig's theorem for the upper bound
Thm. 4: For every n >= 4 and every Hamiltonian cycle C of K_{n,n}, the fractional total chromatic number of K_{n,n} - C is n^2/(n + 1)	Written	recompute.py (7a9a48079da8) [regenerable; replay pending]	External cert only; fidelity: n/a	Written proof for n >= 13 (explicit columns and a diagonally dominant system), written by the author of this draft; replay: exact LP for n = 4 and exact columns and weights for 5 <= n <= 40

Paper Claim	Proof Tier	Lean Decl / Artifact	Actual Axioms & Fidelity	Assumptions & What You Must Trust
Thm. 5: Every regular graph on at most 7 vertices has total chromatic number equal to the ceiling of its fractional total chromatic number. The Mobius ladder M_8 is cubic, has fractional total chromatic number 4 and total chromatic number 5, and is the only connected cubic graph on 8 vertices of Type 2. Hence 8 is the least order of a regular graph whose total chromatic number exceeds the ceiling of its fractional total chromatic number	Cert Only	census.py (ff42470cc907) [regenerable; replay pending]; recompute.py (7a9a48079da8) [regenerable; replay pending]	External cert only; fidelity: n/a	Exhaustive census for n <= 6; the six regular graphs on 7 vertices by exact search and LP (recompute.py block 10); M_8 values from the Lean claim prop_smallregular
Thm. 6: For every finite simple graph G: (i) for every independent set A and every natural number q, a peel certificate for (G, A, q) exists if and only if G[V \ A] is strictly f-degenerate for f(v) = q - d_G(v), subtraction truncated at 0; (ii) if a peel certificate for (G, A, Delta(G) + 1) exists for some A, then the vertices of degree Delta(G) induce a bipartite graph; (iii) if G has a closed walk of odd length all of whose vertices have degree Delta(G), then no set A has a peel certificate for (G, A, Delta(G) + 1)	Lean Kernel	`VizingTotal.paper_peel`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel; axioms propext, Classical.choice, Quot.sound; definitions of Section 2; hypotheses as stated.
Thm. 7: Let D >= 3, let H be a finite simple graph with Delta(H) <= D, let v_0 be a vertex of degree at most D - 1, and let xy be an edge with x, y outside N[v_0]. Let M' be a maximal matching of H - {v_0, x, y} and M_c = {xy} + M'. Then the vertices of degree D in H - M_c are pairwise non-adjacent, H - M_c has a proper D-edge coloring, and the color D on M_c completes a proper (D + 1)-edge coloring of H that is rainbow on the edges at v_0 together with xy. In particular every member (H, J) of the auxiliary class A_D with |M| = 1 has a proper (D + 1)-edge coloring rainbow on J	Written	recompute.py (7a9a48079da8) [regenerable; replay pending]	External cert only; fidelity: n/a	Written proof (Vizing fan lemma in two phases); replay: 150 seeded random instances for each 3 <= D <= 8, the 71 labeled maximal members at (5, 4, 1), and one dense maximal member for each 3 <= D <= 10

Paper Claim	Proof Tier	Lean Decl / Artifact	Actual Axioms & Fidelity	Assumptions & What You Must Trust
Thm. 8: Let H be the graph on {0, 1, 2, 3, 4} with the 7 edges 01, 02, 34, 13, 14, 23, 24 and J = {01, 02, 34}. Then (H, J) is in the auxiliary class A_3 with center 0, and the proper 4-edge coloring psi with the colors 0, 1, 0, 2, 3, 3, 2 on these edges in this order is not rainbow on J. For every two colors a, b and every Kempe-closed set S of psi for a, b, the coloring obtained by swapping a and b on S is not rainbow on J; and two successive Kempe changes give a proper 4-edge coloring that is rainbow on J	Lean Kernel	`VizingTotal.kempe_dist_eq_two_counterexample`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel; axioms propext, Classical.choice, Quot.sound; definitions of Section 2; hypotheses as stated.
Thm. 9: For every Delta >= 3 the graph G*_Delta has (Delta + 1)(2 Delta - 1) vertices, is Delta-regular and connected, has no triangle and contains a 4-cycle; it is of Type 1 and its fractional total chromatic number is Delta + 1. In every proper (Delta + 1)-total coloring some color c* is given to every private vertex and every cross edge, twice the size of its class is (Delta + 1)(3 Delta - 2), every other class has Delta(Delta + 1) elements, and every other class is smaller than the class of c* by at least 2. G*_Delta has no equitable (Delta + 1)-total coloring, and some assignment of lists of size Delta + 1 admits no proper total coloring from the lists	Lean Kernel	`VizingTotal.paper_gDeltaStar`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel; axioms propext, Classical.choice, Quot.sound; definitions of Section 2; hypotheses as stated.
Thm. 10: For every Delta >= 3 the equitable total chromatic number of G*_Delta is Delta + 2. For odd Delta there is an equitable (Delta + 2)-total coloring whose classes all have (Delta + 1)(2 Delta - 1)/2 elements; for even Delta = 2q there is one with q + 1 classes of 4q^2 + q elements and q + 1 classes of 4q^2 + q - 1 elements	Written	recompute.py (7a9a48079da8) [regenerable; replay pending]	External cert only; fidelity: n/a	Written proof; odd Delta: the construction recorded with the finding; even Delta: parity blocks written by the author of this draft; replay: both constructions for 3 <= Delta <= 20

Paper Claim	Proof Tier	Lean Decl / Artifact	Actual Axioms & Fidelity	Assumptions & What You Must Trust
Thm. 11: For every Delta >= 3 the graph H*_Delta is bipartite and Delta-regular, with 4 Delta - 2 vertices and Delta(2 Delta - 1) edges, and contains a 4-cycle. It is of Type 1, and in every proper (Delta + 1)-total coloring one class has 3 Delta - 2 elements and each of the other Delta classes has 2 Delta. Its equitable total chromatic number is 4 for Delta = 3 and Delta + 2 for Delta >= 4, and its list total chromatic number is Delta + 2 for Delta >= 5. For Delta = 4, every list assignment that gives all elements of B_0 one list of 5 colors, all elements of B_1 one list of 5 colors, and each cross edge any list of 5 colors admits a proper total coloring from the lists	Written	recompute.py (7a9a48079da8) [regenerable; replay pending]	External cert only; fidelity: n/a	Written proof, with Galvin's theorem for the list upper bound; replay: colorings for 3 <= Delta <= 8 and the omission sets for 5 <= Delta <= 12; the Delta = 4 clause is written by the author of this draft (block 8 checks only its count inequality)
Prop. 13: Let G have at least one vertex. If Delta(G) = 1, the largest clique of the total graph T(G) has 3 elements; otherwise it has Delta(G) + 1 elements. Every fractional total coloring of G has weight at least the clique number of T(G)	Lean Kernel	`VizingTotal.paper_clique`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel; axioms propext, Classical.choice, Quot.sound; definitions of Section 2; hypotheses as stated.
Prop. 14: Let G have n >= 1 vertices and m edges. Every fractional total coloring of G has weight at least 1 + 2m/n and at least Delta(G) + 1 - def(G)/n. If G is r-regular, every fractional total coloring has weight at least r + 1, and it has weight exactly r + 1 if and only if it covers every element with weight exactly 1 and every set of nonzero weight is a star-saturated total matching	Lean Kernel	`VizingTotal.paper_deficiency_dual`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel; axioms propext, Classical.choice, Quot.sound; definitions of Section 2; hypotheses as stated.
Prop. 15: Every graph of Type 1 is conformable	Lean Kernel	`VizingTotal.paper_conformable`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel; axioms propext, Classical.choice, Quot.sound; definitions of Section 2; hypotheses as stated.

Paper Claim	Proof Tier	Lean Decl / Artifact	Actual Axioms & Fidelity	Assumptions & What You Must Trust
Prop. 16: Let k <= n and let G be an (n - k)-regular graph on two disjoint copies of an n-element set whose edges all join the two copies. Every total matching of G has at most n + k elements, and if n >= 1 every fractional total coloring has weight at least n(n - k + 2)/(n + k). For n >= 1, K_{n,n} is of Type 2 and has fractional total chromatic number n + 2; the crown Cr_2 is of Type 2 and has fractional total chromatic number 3; for n >= 3 the crown Cr_n is of Type 1 and has fractional total chromatic number n	Lean Kernel	`VizingTotal.paper_bipartite`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel; axioms propext, Classical.choice, Quot.sound; definitions of Section 2; hypotheses as stated.
Prop. 17: For every r >= 0 the complete graph K_{2r+1} is of Type 1, and for every r >= 1 the complete graph K_{2r} is of Type 2	Lean Kernel	`VizingTotal.paper_complete`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel; axioms propext, Classical.choice, Quot.sound; definitions of Section 2; hypotheses as stated.
Prop. 18: Let F be a 2-factor of K_{n,n} isomorphic to C_4 + C_{2n-4} and G = K_{n,n} - F. For n = 4, G is of Type 2. For n = 6 and for every odd n >= 5, G is of Type 1	Written	recompute.py (7a9a48079da8) [regenerable; replay pending]	External cert only; fidelity: n/a	Written proof for odd n (explicit coloring written by the author of this draft), replayed for 5 <= n <= 21; n = 4 and n = 6 by exact search
Lem. 19: There are exactly 39 legal steps; they join exactly 38 distinct ordered pairs of states; the legal steps of reduced weight 0 join exactly 11 distinct ordered pairs; and every legal step has reduced weight at most 0	Lean Kernel	`VizingTotal.paper_chiral_table`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel; axioms propext, Classical.choice, Quot.sound; definitions of Section 2; hypotheses as stated.

Paper Claim	Proof Tier	Lean Decl / Artifact	Actual Axioms & Fidelity	Assumptions & What You Must Trust
Prop. 20: Among the 207 unlabeled graphs on 2 to 6 vertices, 142 are connected; of these 133 are of Type 1 and 9 of Type 2, and each of the 9 has total chromatic number equal to the ceiling of its fractional total chromatic number; six of the 9 are not conformable (K_2, K_4, C_5, K_{3,3}, K_6 - e, K_6, with fractional total chromatic numbers 3, 5, 10/3, 5, 19/3, 7) and three are conformable (C_4, K_{3,3} - e and K_6 - (K_{1,3} + K_2), with 4, 13/3, 16/3). Of the 5 connected cubic graphs on 8 vertices exactly one, M_8, is of Type 2; it is conformable and has fractional total chromatic number 4. The Petersen graph, a cubic graph B_10 on 10 vertices with a bridge, and the Tietze graph have chromatic index 4 and total chromatic number 4, and Y_5 has total chromatic number 5	Cert Only	census.py (ff42470cc907) [regenerable; replay pending]	External cert only; fidelity: n/a	Exhaustive computation by checks/census.py
Prop. 21: M_8 has 12 edges, is cubic and conformable, its largest total matching has 5 elements, its fractional total chromatic number is 4, and it is of Type 2. Y_5 has 15 edges, is cubic and conformable, its largest total matching has 6 elements, its fractional total chromatic number is 25/6, and it is of Type 2. Every regular graph on 7 vertices has degree 0, 2, 4 or 6. The edgeless graph on 7 vertices and K_7 are of Type 1. Every fractional total coloring has weight at least 7/2 for C_7, at least 4 for C_3 + C_4, and at least 21/4 for the complement of C_7; the complement of C_7 has no proper 5-total coloring; and the complement of C_3 + C_4 is of Type 1	Lean Kernel	`VizingTotal.paper_small_regular`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel; axioms propext, Classical.choice, Quot.sound; definitions of Section 2; hypotheses as stated.
Prop. 22: Let G be r-regular with at least one vertex. Some set A has a peel certificate for (G, A, r + 1) if and only if G has a proper vertex 2-coloring. If moreover G is connected, then for every proper 2-coloring of G and every set A, a peel certificate for (G, A, r + 1) exists if and only if A is one of the two color classes	Lean Kernel	`VizingTotal.paper_peel_regular`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel; axioms propext, Classical.choice, Quot.sound; definitions of Section 2; hypotheses as stated.

Paper Claim	Proof Tier	Lean Decl / Artifact	Actual Axioms & Fidelity	Assumptions & What You Must Trust
Cor. 23: For m >= 1 no set A of vertices of the cycle C_{2m+1} has a peel certificate for (C_{2m+1}, A, Delta(C_{2m+1}) + 1), and for n >= 3 no set A of vertices of K_n has a peel certificate for (K_n, A, Delta(K_n) + 1)	Lean Kernel	`VizingTotal.paper_peel_odd_cycle_complete`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel; axioms propext, Classical.choice, Quot.sound; definitions of Section 2; hypotheses as stated.
Prop. 24: Let (H, J) be in A_D with center v_0 and M the set of edges of J not at v_0. Then d_H(v_0) + |M| = D and H has 1 + d_H(v_0) + 2|M| vertices. If |M| = 0, every proper k-edge coloring of H, for any k, is rainbow on J	Lean Kernel	`VizingTotal.paper_aux_counts`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel; axioms propext, Classical.choice, Quot.sound; definitions of Section 2; hypotheses as stated.
Prop. 25: Let H_1 be a subgraph of H_2 on the same vertex set, let (H_1, J) be in A_D with center v_0, and suppose Delta(H_2) <= D and every neighbor of v_0 in H_2 is a neighbor of v_0 in H_1. Then (H_2, J) is in A_D with center v_0, and if H_2 has a proper k-edge coloring rainbow on J, so has H_1	Lean Kernel	`VizingTotal.paper_aux_monotone`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel; axioms propext, Classical.choice, Quot.sound; definitions of Section 2; hypotheses as stated.
Lem. 26: Let phi be a proper coloring with colors {0, ..., k - 1} of a set F of edges of a graph, and let vw be an edge not in F. If a color is missing at v, and each vertex of {w} + {y : vy in F} has a missing color, which may differ from vertex to vertex, then the colors of F can be changed so that F together with vw is properly colored with the same k colors	Written	Written proof	Lean: none; fidelity: n/a	Written proof (standard Vizing fan argument)
Lem. 27: Let (H, J) be in A_D with center v_0 and M = {x_1 y_1, e_2}. Let M_1′ be a maximal matching of H - {v_0, x_1, y_1} with the edge e_2 deleted, and M_D = {x_1 y_1} + M_1′. Then the only edge of H - M_D that can join two vertices of degree D in H - M_D is e_2	Written	Written proof	Lean: none; fidelity: n/a	Written proof (maximality of the matching)

Paper Claim	Proof Tier	Lean Decl / Artifact	Actual Axioms & Fidelity	Assumptions & What You Must Trust
Prop. 28: For a triple (D, s, p) with s + p = D and 2 <= s <= D, fix the vertex set {0, ..., s + 2p}, the center v_0 = 0 and J = {01, ..., 0s} + {(s+1)(s+2), ..., (s+2p-1)(s+2p)}; a labeled member is a graph H on this vertex set with (H, J) in A_D, and a coloring c has the star fixed if c(0i) = i - 1 for 1 <= i <= s. For each of the ten triples with D <= 5, every edge-maximal labeled member has a proper (D + 1)-edge coloring rainbow on J; there are 702 such graphs up to the symmetries of J and 44647 labeled ones, of which 675 and 44484 have D = 5. Every proper 4-edge coloring of a maximal member at (3, 2, 1) is within 2 single Kempe changes of a rainbow one: of the 76 with the star fixed the distances 0, 1, 2 occur 40, 32 and 4 times (480, 384 and 48 of all 912). With the star fixed, the distances 0, 1, 2 occur 1008, 1752 and 516 times among the 3276 colorings of the 17 labeled maximal members at (4, 3, 1) (196560 colorings without fixing the star), and the distances 0, 1, 2, 3 occur 145824, 258192, 68832 and 240 times among the 473088 colorings of the 71 labeled maximal members at (5, 4, 1). At (4, 2, 2) there are 17 maximal members up to symmetry (138 labeled). For either edge of M taken as x_1 y_1 and every maximal M_1', the peeling of the lemma leaves the vertices of degree 4 independent or joined only by e_2. A matching of H - v_0 that contains x_1 y_1 and not e_2 and covers every vertex of degree 4 in H exists for at least one choice of x_1 y_1 in 14 of the 17, and for both choices in 9. Of the 728 connected graphs on the vertex set {0, ..., 4}, 565 have a peel certificate at q = Delta + 1	Cert Only	auxiliary.py (4b68c40a1eb5) [regenerable; replay pending]	External cert only; fidelity: n/a	Exhaustive computation by checks/auxiliary.py

Paper Claim	Proof Tier	Lean Decl / Artifact	Actual Axioms & Fidelity	Assumptions & What You Must Trust
Lem. 29: Let Delta >= 2 and let psi be a proper total coloring of K_ {Delta-1,Delta}, with sides P of Delta - 1 vertices and T of Delta vertices, by a set C of Delta + 1 colors. Then some color c* is given to every vertex of P and to no edge; the vertices of T get distinct colors, which are exactly the colors other than c*; at each vertex t of T the edges carry every color other than c* and psi(t); and every color other than c* is on exactly Delta - 1 edges	Lean Kernel	`VizingTotal.k_pred_succ_monochromatic_forcing`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel; axioms propext, Classical.choice, Quot.sound; definitions of Section 2; hypotheses as stated.
Prop. 30: G*_3 has 20 vertices and 30 edges, and every proper 4-total coloring has one class of 14 elements and three classes of 12	Lean Kernel	`VizingTotal.gDeltaStar_three_profile`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel; axioms propext, Classical.choice, Quot.sound; definitions of Section 2; hypotheses as stated.
Prop. 31: Let H_4^(4) consist of four copies B_0, ... , B_3 of K_{3,4}, with private vertices p_{i,r} and terminals t_{i,j}, and the cross edges t_{i,2} t_{i+1,0} and t_{i,3} t_{i+1,1} for i in Z_4. It has 28 vertices and 56 edges, is 4-regular and bipartite, is of Type 1, and its list total chromatic number is 6	Written	recompute.py (7a9a48079da8) [regenerable; replay pending]	External cert only; fidelity: n/a	Written proof; replay: the 625 cases and an exact 5-total coloring; upper bound by Galvin's theorem

[a] Allowed pure-kernel axioms: `propext`, `Classical.choice`, `Quot.sound`. Actual axioms printed above are read directly from verified outputs. Provenance nonce: e5871405e0f9 | Lean mode: committed_receipt | Closure SHA-256: f07d54ccedfc | Git anchor: 0ddf107833cb | Axioms SHA-256: 2e2d3d8099e5 | Fidelity SHA-256: 9e00a28681ba.

6. PRIOR WORK AND OPEN QUESTIONS

The conjecture. The Total Coloring Conjecture was posed by Behzad [3] and Vizing [40, 41]. It holds for $\Delta = 3$ [36, 39], $\Delta = 4$ [29] and $\Delta = 5$ [30], for planar graphs with $\Delta \geq 7$ [5, 37, 31], and, for each fixed $\varepsilon > 0$, for sufficiently large r-regular graphs with $r \geq (1+\varepsilon)|V|/2$ [11], a result extended to sufficiently large graphs of minimum degree at least $(1+\varepsilon)|V|/2$ [21]; Dalal, McDonald and Shan also prove $\chi''(G) \leq \Delta(G) + 2\lceil |V|/(\Delta(G)+1) \rceil$. Hind, Molloy and Reed [22] and Molloy and Reed [32] gave general upper bounds of the form $\Delta + o(\Delta)$ and $\Delta + C$. Kilakos and Reed [27] proved $\chi''_f \leq \Delta + 2$. For $\Delta = 3$ and for even $\Delta \geq 4$, Kaiser, King and Král' [24] showed $\chi''_f = \Delta + 1$ for graphs of maximum degree Δ and girth large enough in terms of Δ; for odd $\Delta \geq 5$ and every $\varepsilon > 0$, Kardoš, Král' and Sereni [25] showed $\chi''_f \leq \Delta + 1 + \varepsilon$ for girth large enough in terms of Δ and ε. Yap's monograph [43] and the survey of Geetha, Narayanan and Somasundaram [18] collect the classical values, among them those of complete graphs [4], of dense bipartite graphs [8, 20] and of cycles; Brinkmann, Preissmann and Sasaki [7] study the total chromatic number of snarks, Müßig [33] checks the conjecture for small orders by computer, and Zhu, Zhou and Xu [45] and Zhu [44] treat planar graphs of maximum degree 6. Ferrarini and Gualandi [14] study the polytope of total matchings. We read none of these sources in full; all carry [to check] in `references-checked.md`.

Sources compared with each result. For Möbius ladders and prisms: the Type 2 property of M_{2k} is credited to Chetwynd and Hilton [10] by Alves Junior and Sasaki [23]; Khennoufa and Togni [26] bound χ''_f of cubic circulants from above, and Hackmann and Kemnitz [19] study circular total colorings of cubic circulants; Dejter [13] constructs equitable total colorings of some prisms. We read only abstracts or snippets of these, so Theorem 1(ii) and Theorem 2 may overlap them [to check]. For G^*_Δ: Noel's paper [34], read in full, builds G^*_3 from four copies of $K_{2,3}$ joined by single cross edges, proves $\chi'' = 4$ and $\chi''_\ell = 5$, and gives its House of Graphs number 27005; Adauto, de Figueiredo, Sasaki and Schneider [2], of which we read the abstract, the introduction and the passages on order 20, draw the graph R of Dantas et al. [12] with $\chi'' = 4$ and $\chi''_e = 5$, and Abreu et al. [1] list the same graph [to check]. For the peeling results: the auxiliary class, the peel certificate and the $(D+2)$-color theorem are from the Lean library of Chen and Shan [9], whose statements we read; the $(D+1)$-color case is

not claimed there. Fu [16] conjectured $\chi''_e \le \Delta + 2$, proved for $\Delta \le 3$ by Wang [42]. The equitable total colorings of cubic graphs are studied by Dantas et al. [12], Adauto et al. [2] and Shimizu [38].

What was searched. The prior-work search is logged in `prior-work-search/searchlog.md`, and its gate passes. It ran keyword and phrase queries on arXiv, CORE, Google Scholar through a keyed provider, GitHub and the web. Not searched: OpenAlex and Semantic Scholar, both refused requests; Math StackExchange and MathOverflow; the full texts of the paywalled articles above; Yap's monograph [43] and books on fractional graph theory; HAL full texts; Zenodo, OSF Preprints and problem lists. Among the sources read, none states Theorem 1(i)–(iii), Theorem 4, Theorem 5, Theorem 10 for $\Delta \ge 4$ or the list bound of Theorem 11; the unread full texts above may.

Open questions. (i) Does every $(H, J) \in \mathcal{A}_D$ have a proper $(D+1)$-edge coloring rainbow on J when $|M| \ge 2$? (ii) Is $\chi''_\ell(G^*_\Delta) = \Delta + 2$? (iii) Is $K_{n,n} - (C_4 \sqcup C_{2n-4})$ of Type 1 for even $n \ge 8$, and which other 2-factors containing a 4-cycle leave a Type 1 graph? (iv) Is $\chi''_\ell(H^*_4) = 6$? (v) Can the written proofs be moved into Lean? (vi) Review by a human mathematician.

7. DATA AVAILABILITY, REPRODUCIBILITY, AND USE OF AI

In the `mathdisc` repository the draft folder `papers/vizing-total-coloring/draft1/` has the manuscript, `recompute.py` and `checks/` with their outputs, the figure scripts, the search log and the ledgers; its `README.md` gives the commands that re-run everything. The finding folder `findings/vizing-total-coloring/` has the Lean library in `lean/`, with `PaperDraft1.lean`, `VerifyDraft1.lean`, the axiom printout and receipt, the evidence, and the earlier manuscript in `paper/`, kept unchanged.

What changed from the earlier manuscript. The theorems of the earlier manuscript are carried, with the changes listed here and below. Five arguments are ours and differ from the recorded ones; the three that replace a recorded proof are these. (1) Theorem 4: the earlier text bounds a perturbation for $n \ge 10$ without stating its matrix, and its script prints the values for $n \le 10$ without computing them; Section 3 gives explicit matchings, a bound for $n \ge 13$ and an exact computation for $5 \le n \le 40$. (2) Theorem 10 for even Δ: the earlier construction needs, in each block B_i with $i > q$, $2q-2$ block edges that avoid $v_{i,0}, v_{i,1}$ and carry all the colors other than theirs, and does not show that they exist; the parity blocks avoid that step. (3) Proposition 18 for odd n: the earlier text gives the coloring in one sentence and cites $n = 6$ to a receipt that is not in the finding folder; above, the coloring is explicit with a proof, and $n = 6$ is an exact search. The other two are the fan argument behind Theorem 7 and the tile junctions in the proof of Theorem 2. The equality $\chi''_f(Cr_2) = 3$ and the 4-cycle of G^*_Δ, stated without Lean proofs earlier, are now proved in Lean. That G^*_3 is the graph R of Dantas et al. [12], stated earlier, is a belief here, as we have only compared figures. The barrier for one list per block at $\Delta = 4$, given earlier with a count of pairs, is now a clause of Theorem 11, and the argument is ours. The count 36 in the proof of Proposition 31 and the odd cycles of G^*_Δ and Y_5 are written observations, outside Lean. At $(4,2,2)$ the earlier count 11 fixed one edge of M of a chosen representative as x_1y_1 and is replaced by the counts 14 and 9 of Proposition 28, which do not depend on that choice; and at $(5,4,1)$ the earlier seeded search, along Kempe chains through three fixed vertices, found distance at most 2, while the exhaustive search finds 240 colorings at distance 3. The earlier table puts all three regular graphs of Type 2 on 7 vertices among the non-conformable ones; only $\overline{C_7}$ is, and the text after the proof of Theorem 5 gives the correct ones. The earlier claim table states $\chi''_f = n^2/(n+1)$ for every $K_{n,n}$ minus a 2-factor without 4-cycles; Theorem 3 has the lower bound for that class and Theorem 4 the equality for a Hamiltonian cycle and $n \ge 4$ only ($K_{3,3}$ minus a 6-cycle has $\chi''_f = 3$). The earlier runs of the fan procedure, on the 8 classes at $(5,4,1)$ and on dense graphs through $D = 10$, rest on a receipt that is not in the finding folder; block 10 runs it on the 71 labeled maximal members at $(5,4,1)$ and on dense members of our own choice.

What is stated elsewhere or left out. The fractional value $r+1$ of an r-regular graph in terms of a conic hull of star-saturated total matchings is the exact covering of Proposition 14. G^*_Δ, H^*_Δ and $H^{(4)}_4$ have girth 4 (a 4-cycle and no triangle). The table of optimal weightings of Möbius ladders is replaced by the weightings in the proof of Theorem 1, with the rows for $k = 4, 7, 8, 10$ in block 2. The numbers $36, 49, 64, 81$ of pairs covered for $\Delta = 5, \dots, 8$ are the $(\Delta+1)^2$ pairs of block 8, and the orbit weights of the prisms are in block 6. Left out are constants and orders quoted from other sources (additive bounds, orders 14, 16, 36, the hypothesis $|V| \le 2\Delta$) and the counts of the earlier edition's own checks (308 declarations, 10 smoke tests, 10 proved negations, 12 planted variants, claims split 21 and 6), replaced by Section 5.

Use of AI. AI research agents built and directed by Tyler B. Hughes found the results, wrote the proofs, the Lean formalization and the checkers, and drafted this manuscript. Statement fidelity of the Lean claims was audited by a model of another family than the author's, and a blind referee of another family reviewed the draft. No human mathematician has yet refereed the proof.

Terms of Section 5. The project's checking tool writes Section 5 and the first-page box. "G11": coverage as counts of claims with denominators. *Certificate*: a checker's output bound by hash. *Human audit surface*: the printed definitions, statements and type-trap dispositions. *Gate-box replay*: a Lean rebuild on a separate machine; none has been run.

Drafted: November 2026 **Updated:** November 25, 2026

REFERENCES

(1) Abreu, M.; Goedgebeur, J.; Labbate, D.; Mazzuoccolo, G. Colourings of cubic graphs inducing isomorphic monochromatic subgraphs. *J. Graph Theory* **2019**, *92*, 415–444. [to check]

(2) Adauto, M.; de Figueiredo, C.; Sasaki, D.; Schneider, R. Order 14 is the largest order for which every 4-total coloring of every cubic graph is equitable. arXiv:2609.05259, 2026. [to check]

(3) Behzad, M. Graphs and their chromatic numbers. Ph.D. thesis, Michigan State University, 1965. doi:10.25335/m5h41k22c. [to check]

(4) Behzad, M.; Chartrand, G.; Cooper, J. K., Jr. The colour numbers of complete graphs. *J. London Math. Soc.* **1967**, *42*, 226–228. [to check]

(5) Borodin, O. V. On the total coloring of planar graphs. *J. Reine Angew. Math.* **1989**, *394*, 180–185. [to check]

(6) Borodin, O. V.; Kostochka, A. V.; Woodall, D. R. List edge and list total colourings of multigraphs. *J. Combin. Theory Ser. B* **1997**, *71*, 184–204. [to check]

(7) Brinkmann, G.; Preissmann, M.; Sasaki, D. Snarks with total chromatic number 5. *Discrete Math. Theor. Comput. Sci.* **2015**, *17*, 369–382. [to check]

(8) Chen, B.-L.; Dong, L.; Liu, Q.-Z.; Huang, K.-C. Total colorings of equibipartite graphs. *Discrete Math.* **1999**, *194*, 59–65. [to check]

(9) Chen, L.; Shan, S. Total Coloring Lean (total-coloring-lean, release v0.2.0), Lean 4 library. Zenodo, 2026. doi:10.5281/zenodo.23452514. [to check]

(10) Chetwynd, A. G.; Hilton, A. J. W. Some refinements of the total chromatic number conjecture. *Congr. Numer.* **1988**, *66*, 195–216. [to check]

(11) Dalal, A.; McDonald, J.; Shan, S. Total coloring graphs with large maximum degree. arXiv:2405.07382, 2024. [to check]

(12) Dantas, S.; de Figueiredo, C. M. H.; Mazzuoccolo, G.; Preissmann, M.; dos Santos, V. F.; Sasaki, D. On the equitable total chromatic number of cubic graphs. *Discrete Appl. Math.* **2016**, *209*, 84–91. [to check]

(13) Dejter, I. J. From semi-total to equitable total colorings, and their induced partitions into perfect and total perfect codes. arXiv:2503.20055, 2025. [to check]

(14) Ferrarini, L.; Gualandi, S. Total coloring and total matching: polyhedra and facets. arXiv:2105.09827, 2021. [to check]

(15) Fournier, J.-C. Colorations des arêtes d'un graphe. *Cahiers Centre Études Rech. Opér.* **1973**, *15*, 311–314. [to check]

(16) Fu, H.-L. Some results on equalized total coloring. *Congr. Numer.* **1994**, *102*, 111–119. [to check]

(17) Galvin, F. The list chromatic index of a bipartite multigraph. *J. Combin. Theory Ser. B* **1995**, *63*, 153–158. [to check]

(18) Geetha, J.; Narayanan, N.; Somasundaram, K. Total colorings—a survey. *AKCE Int. J. Graphs Comb.* **2023**, *20*, 339–351. [to check]

(19) Hackmann, A.; Kemnitz, A. Circular total colorings of cubic circulant graphs. *J. Combin. Math. Combin. Comput.* **2004**, *49*, 65–72. [to check]

(20) Hamilton, G. M.; Hilton, A. J. W.; Hind, H. R. F. Totally critical even order graphs. *J. Combin. Theory Ser. B* **1999**, *76*, 262–279. [to check]

(21) Henderschedt, O.; McDonald, J.; Shan, S. Total coloring graphs with large minimum degree. arXiv:2507.05548, 2025. [to check]

(22) Hind, H.; Molloy, M.; Reed, B. Total coloring with Δ + poly$(\log \Delta)$ colors. *SIAM J. Comput.* **1998**, *28*, 816–821. [to check]

(23) Alves Junior, M. N.; Sasaki, D. A result on total coloring of circulant graphs. In *Anais do Encontro de Teoria da Computação*, SBC, 2020, 81–84. [to check]

(24) Kaiser, T.; King, A.; Král', D. Fractional total colourings of graphs of high girth. arXiv:0911.2808, 2009. [to check]

(25) Kardoš, F.; Král', D.; Sereni, J.-S. The last fraction of a fractional conjecture. *SIAM J. Discrete Math.* **2010**, *24*, 699–707. [to check]

(26) Khennoufa, R.; Togni, O. Total and fractional total colourings of circulant graphs. *Discrete Math.* **2008**, *308*, 6316–6329. [to check]

(27) Kilakos, K.; Reed, B. Fractionally colouring total graphs. *Combinatorica* **1993**, *13*, 435–440. [to check]

(28) König, D. Über Graphen und ihre Anwendung auf Determinantentheorie und Mengenlehre. *Math. Ann.* **1916**, *77*, 453–465. [to check]

(29) Kostochka, A. V. The total coloring of a multigraph with maximal degree 4. *Discrete Math.* **1977**, *17*, 161–163. [to check]

(30) Kostochka, A. V. The total chromatic number of any multigraph with maximum degree five is at most seven. *Discrete Math.* **1996**, *162*, 199–214. [to check]

(31) Kowalik, Ł.; Sereni, J.-S.; Škrekovski, R. Total-coloring of plane graphs with maximum degree nine. *SIAM J. Discrete Math.* **2008**, *22*, 1462–1479. [to check]

(32) Molloy, M.; Reed, B. A bound on the total chromatic number. *Combinatorica* **1998**, *18*, 241–280. [to check]

(33) Müßig, M. total-coloring-conjecture, GitHub repository, 2025. [to check]

(34) Noel, J. A. The list total colouring conjecture is false. arXiv:2609.38417, 2026.

(35) OEIS Foundation Inc. The On-Line Encyclopedia of Integer Sequences, sequences A000088, A001349 and A002851. https://oeis.org. [to check]

(36) Rosenfeld, M. On the total coloring of certain graphs. *Israel J. Math.* **1971**, *9*, 396–402. [to check]

(37) Sanders, D. P.; Zhao, Y. On total 9-coloring planar graphs of maximum degree seven. *J. Graph Theory* **1999**, *31*, 67–73. [to check]

(38) Shimizu, H. GitHub repository equitable-total-cubic-16, 2026. [to check]

(39) Vijayaditya, N. On total chromatic number of a graph. *J. London Math. Soc.* (2) **1971**, *3*, 405–408. [to check]

(40) Vizing, V. G. On an estimate of the chromatic class of a p-graph. *Diskret. Analiz* **1964**, *3*, 25–30. [to check]

(41) Vizing, V. G. Some unsolved problems in graph theory. *Russian Math. Surveys* **1968**, *23*; Russian original in *Uspekhi Mat. Nauk* **1968**, *23*, 117–134. [to check]

(42) Wang, W.-F. Equitable total coloring of graphs with maximum degree 3. *Graphs Combin.* **2002**, *18*, 677–685. [to check]

(43) Yap, H. P. *Total Colourings of Graphs*. Lecture Notes in Mathematics 1623, Springer, Berlin, 1996. [to check]

(44) Zhu, E. Total eight-colorings of maximal planar graphs of maximum degree six. arXiv:2610.32717, 2026. [to check]

(45) Zhu, E.; Zhou, Y.; Xu, J. Total-coloring of planar graphs with maximum degree 6 and without prescribed 4-cycles. arXiv:2609.19503, 2026. [to check]