The three-color Ramsey number of the 10-vertex path

P. Graham
https://orcid.org/0009-0007-8493-1505

November 4, 2026

Abstract

We prove $R(P_{10}, P_{10}, P_{10}) = 18$: every three-coloring of the edges of K_{18} contains a monochromatic path on ten vertices, and some three-coloring of K_{17} contains none. Faudree and Schelp suggested a formula for three-color path Ramsey numbers. Gyárfás, Ruszinkó, Sárközy and Szemerédi proved that the three-color Ramsey number of the path on n vertices is $2n-2+(n \bmod 2)$ for all sufficiently large n. Dybizbański, Dzido and Radziszowski established the eight- and nine-vertex cases, completing the cases through nine vertices recorded by Radziszowski's survey. The lower-bound coloring is the even-case K_4 blow-up given by Gyárfás, Ruszinkó, Sárközy and Szemerédi and by Dybizbański, Dzido and Radziszowski. The new upper bound follows from Boolean satisfiability refutations after fixing a monochromatic nine-vertex path and imposing simultaneous symmetry constraints. A separate refutation proves that the cases considered cover every coloring. The reduction and the certificate checks are formalized in Lean.

This work has not yet been reviewed by experts. The proofs, the Lean formalization, the search code, the certificate checkers and the text were produced by AI models (from OpenAI and Anthropic), working under the author's direction.

1 Introduction

For graphs G_1, G_2, G_3, the Ramsey number $R(G_1, G_2, G_3)$ is the least N such that every coloring of the edges of K_N with colors $1, 2, 3$ contains, for some i, a copy of G_i all of whose edges have color i. We write $R_3(G) = R(G, G, G)$, and P_n for the path on n vertices (with $n-1$ edges). Paths are not required to be induced.

Theorem 1.1. $R_3(P_{10}) = 18$. *That is, every coloring of the edges of K_{18} with three colors contains a monochromatic path on 10 vertices, and some 3-coloring of the edges of K_{17} contains none.*

The advance is the upper bound $R_3(P_{10}) \le 18$. Gyárfás, Ruszinkó, Sárközy and Szemerédi proved $R_3(P_n) = 2n - 2 + (n \bmod 2)$ for all sufficiently large n [18, Theorem 1], with its corrigendum. Dybizbański, Dzido and Radziszowski proved $R_3(P_8) = 14$ (Theorem 10) and $R_3(P_9) \le 17$ (Theorem 8), which with their lower bound gives $R_3(P_9) = 17$ [8]. Radziszowski's survey [31, Section 6.4, p. 65] records the formula through $n = 9$ and P_{10} as the first open case. Our K_{17} lower bound is the even-case blow-up in [18], also given in [8]. In Conjecture 2 of [8], Dybizbański, Dzido and Radziszowski attribute the suggestion to Faudree and Schelp [15].

Conjecture 1.2 (suggested by Faudree–Schelp [15]; stated as a conjecture in [8]; see [31])**.** *For all $n \ge 1$, $R_3(P_n) = 2n - 2 + (n \bmod 2)$.*

This paper settles $n = 10$ of Conjecture 1.2; the general conjecture remains open. Starting from their nine-vertex bound, our upper-bound computation combines simultaneous symmetry constraints with a separately verified cover of the remaining cases. The difficult step is obtaining a tractable case split without losing a coloring: fixing a guaranteed path and imposing symmetry constraints must be jointly sound, and the case division must be proved complete. A lexicographic minimum and a separate refutation excluding all assignments outside the selected cases discharge these obligations. Dybizbański, Dzido and Radziszowski reported that their attempts at the ten- and eleven-vertex cases remained incomplete [8].

Shao, Zhu, Xu and Kosari have independently proved $R_3(P_{10}) = 18$ by a different route [32]. Their reproducibility materials, posted on November 2, 2026, accompany their manuscript (revision 7), which we describe from the package's reproduction guide: the proof is structural, and its three finite verifications (its Lemmas 13, 14 and 28) are exhaustive enumerations by C++ programs that need no SAT solver. The package records a verification run from September 8, 2026.

The method combines cube-and-conquer with symmetry breaking and verified certificate checking. Heule, Kullmann, Wieringa and Biere's cube-and-conquer method divides a satisfiability problem into subproblems [20]. For certificate import we use Böving et al.'s verified LRAT checker [2]. LRAT certificates are propositional refutations whose steps refer to earlier clauses by number. Szeider's streaming tool `lrat-catcher` imports certificates into Lean theorems [35]; our import uses the same checker with its own compact encoding. Kirchweger, Manrique and Szeider combine SAT modulo symmetries and Lean for formally verified graph generation [23].

Computational precedents include Heule, Kullmann and Marek's Boolean Pythagorean triples proof [19, Theorem 1], Codish, Frank, Itzhakov and Miller's $R(4,3,3) = 30$ [5], and Heule and Scheucher's empty-hexagon theorem [21], with Subercaseaux et al.'s formal verification [34]. In related Ramsey computations, Wesley proves $R(C_3, C_6, C_6) = R(C_5, C_6, C_6) = 15$ [37, Theorems 3 and 4]; Galindez's draft reports $R(C_{10}, C_{10}, C_{10}) \leq 26$ [16, Theorem 1], using cube-and-conquer and Kirchweger–Szeider's SAT modulo symmetries [25]; the draft explicitly reports no DRAT/LRAT proof certificate. Shao, Jiang, Xu and Wang provide a reproduction package for $R(C_4, C_6, C_8) = 12$ [33], and Leo-Y-Zhang provides *CycleRamsey* [42]. Zhang's tree-bound corollary $R_k(T) \leq k(n-2)+2$ for trees on $n \geq 2$ vertices is `multicolourRamsey_le_tight` in `Erdos557/Tight.lean` [43]. It uses an AI-found Erdős–Sós proof packaged by T. Adamczewski and gives $R_3(P_{10}) \leq 26$.

Earlier path and cycle literature includes Dzido [10], Dzido–Fidytek [9], Omidi–Raeisi [30], Maherani et al. [27, Theorem 1.1], Benevides–Skokan [4], Figaj–Łuczak [14] and Letzter [26]. For small cases, see Irving, Yang–Rowlinson and Yang [22, 40, 41, 39], and Radziszowski's survey [31]. The extremal background is due to Erdős–Gallai [11], Faudree–Schelp [15], whose theorem is recalled by Dybizbański, Dzido and Radziszowski [8, Theorem 4]. See also Kopylov's work on paths and cycles [24]. Allen et al. study paths and cycles with few color changes [1]. We use de Moura and Ullrich's Lean 4 [29], the mathlib Community's library [28], Wetzler, Heule and Hunt's `drat-trim` [38] for conversion to Cruz-Filipe et al.'s LRAT format [7], and Tan, Heule and Myreen's `cake_lpr` [36] for the separate checks.

The proof uses Boolean satisfiability certificates and Lean; Section 5 describes this dependence.

2 Proof overview

The lower bound is the K_4 blow-up in Figure 1: every monochromatic component has at most nine vertices. For the upper bound, we first re-derive Dybizbański, Dzido and Radziszowski's bound [8, Theorem 8] by successive Boolean satisfiability refutations, starting with P_3 on K_5 and ending with P_9 on K_{17}. Each formula fixes only the monochromatic path guaranteed by the preceding bound.

A hypothetical coloring of K_{18} without a monochromatic P_{10} therefore contains a P_9. After fixing that path, we choose a least coloring among those obtained by the permitted vertex and color permutations. It satisfies all the imposed lexicographic inequalities simultaneously. To make the final refutations tractable, we divide the remaining assignments into cubes, conjunctions of edge-color assignments. One refutation excludes assignments outside all the cubes; a refutation for each cube then completes the upper bound. Lean checks the reductions and these finite certificates, as described in Section 5.

3 The lower bound

The following K_4 blow-up gives a coloring of K_{17} without a monochromatic ten-vertex path. In the construction and the upper-bound encoding we use the color labels $0, 1, 2$ in place of $1, 2, 3$. Partition $\{0, \dots, 16\}$ into $A = \{0, \dots, 4\}$, $B = \{5, \dots, 8\}$, $C = \{9, \dots, 12\}$ and $D = \{13, \dots, 16\}$. Color an edge with color 0 if both ends lie in $A \cup B$ or both lie in $C \cup D$; with color 1 if it joins A to C or B to D; and with color 2 if it joins A to D or B to C. This is the blow-up of the factorization of K_4 into the matchings $\{AB, CD\}$, $\{AC, BD\}$, $\{AD, BC\}$, with the edges inside the parts given color 0. This is the construction of Gyárfás, Ruszinkó, Sárközy and Szemerédi [18], also presented by Dybizbański, Dzido and Radziszowski [8]; both specify the odd and even part sizes.

For each color, its *label classes* are the two parts of the corresponding partition: $(A \cup B, C \cup D)$ for color 0, $(A \cup C, B \cup D)$ for color 1, and $(A \cup D, B \cup C)$ for color 2. The *edge invariant* says that an edge of a given color has both ends in one of its label classes. Figure 1 displays the four constituent parts.

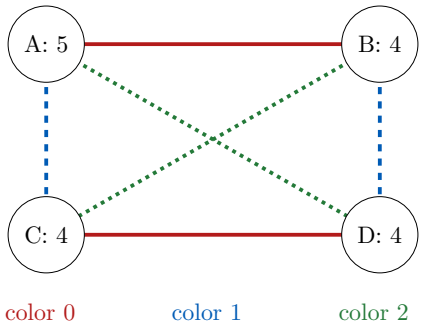


Figure 1: The K_{17} coloring, compressed into four parts. Each line represents all edges between its parts; all edges within a part have color 0. In each color, the two label classes have sizes 9 and 8.

Proposition 3.1. *This coloring of K_{17} has no monochromatic P_{10}. Hence $R_3(P_{10}) \geq 18$.*

Proof. Each color has two label classes: $A \cup B$ and $C \cup D$ for color 0, $A \cup C$ and $B \cup D$ for color 1, $A \cup D$ and $B \cup C$ for color 2, of sizes 9 and 8 in each case. The edge invariant is that every edge of a color lies inside one of that color's two label classes; no edge of that color joins them. So every connected piece of a color class lies inside one label class and has at most 9 vertices, and a monochromatic path, being connected, has at most 9 vertices. □

4 The upper bound

We encode a hypothetical coloring without a monochromatic path and refute the resulting finite formulas. Throughout, a *coloring* of K_n is a map X from the edges to $\{0,1,2\}$. For a fixed path order m, such a coloring is *good* if it has no monochromatic P_m; goodness refers to this same m throughout each formula and its associated lemmas. We order the edges $\{i,j\}$, $i<j$, row by row: $\{0,1\},\{0,2\},\dots,\{0,n-1\},\{1,2\},\dots,\{n-2,n-1\}$, and number them $e=0,1,\dots,\binom{n}{2}-1$ in this order.

4.1 Variables and the coloring clauses

We express the coloring and path conditions as Boolean clauses. A Boolean literal is a variable or its negation; a clause is a disjunction of literals. A model assigns truth values satisfying every clause; a refutation certifies that no such assignment exists.

The variable $x_{e,c}=3e+c+1$ means "edge e has color c". An unordered pair in the edge subscript denotes its number in the preceding edge order. For each edge there is one clause $x_{e,0}\vee x_{e,1}\vee x_{e,2}$ and three clauses $\neg x_{e,c}\vee\neg x_{e,d}$, $c<d$, so that models correspond to colorings. For a color c and distinct vertices $p_0,\dots,p_{m-1}$, the *path clause*

$$\bigvee_{i=0}^{m-2}\neg x_{\{p_i,p_{i+1}\},c}$$

says that this path is not monochromatic in color c. A coloring is good exactly when its assignment satisfies every path clause. Only the path clauses used in a refutation need to be listed: a good coloring satisfies every path clause, hence any listed subset. Lean rebuilds each listed clause from its color and distinct vertices, so the solver's path-finding procedure need not be assumed complete.

4.2 The fixed path

Vertex permutations $\pi\in S_n$ and color permutations $\sigma\in S_3$ act on colorings by $(g\cdot X)(\{\pi a,\pi b\})=\sigma(X(\{a,b\}))$ for $g=(\pi,\sigma)$. On edges, $\pi\{a,b\}=\{\pi(a),\pi(b)\}$. They map good colorings to good colorings.

The following lemma fixes only the edges of the guaranteed path.

Lemma 4.1 (fixed path)**.** *Fix a path order m, and call a coloring good if it has no monochromatic P_m. Let $k\le n$ and suppose every 3-coloring of K_n contains a monochromatic P_k. Then every good coloring of K_n is equivalent under $S_n\times S_3$ to a good coloring X with $X(\{t,t+1\})=0$ for $0\le t\le k-2$.*

Proof. Take a monochromatic path $v_0,\dots,v_{k-1}$ in color c; apply a vertex permutation sending v_t to t and the transposition of colors c and 0. □

The formula for P_m on K_n with a fixed P_k contains the $k-1$ unit clauses $x_{\{t,t+1\},0}$. Only the path edges are constrained; the chords of the fixed path are free.

4.3 Symmetry constraints

The following constraints retain a lexicographically least coloring. Compare colorings by the lexicographic order $\le_{\mathrm{lex}}$ of their color vectors $(X(e))_e$ in edge order, with $0<1<2$. Let Γ be a list of generators $g=(\pi,\sigma)$. With a fixed P_k on $0,\dots,k-1$ and $W=\{k,\dots,n-1\}$, Γ consists of

the transpositions of pairs of vertices of W (in lexicographic order), the reversal $t \mapsto k-1-t$ of $\{0,\dots,k-1\}$, and the color transposition $1 \leftrightarrow 2$. For K_5, nothing is fixed and Γ consists of all vertex transpositions and the color transpositions $0 \leftrightarrow 1$ and $1 \leftrightarrow 2$.

The orbit of a coloring consists of the colorings reachable by finitely many moves from Γ. A least member of an orbit gives simultaneous symmetry inequalities, as follows.

Lemma 4.2 (lex-leader)**.** *Fix a path order m. Let S be the set of good colorings of K_n (those with no monochromatic P_m) with color 0 on the fixed path. Each $g \in \Gamma$ maps S to itself. Consequently, if S is nonempty it contains a coloring X^* with $X^* \le_{\text{lex}} g \cdot X^*$ for every $g \in \Gamma$.*

Proof. When a path is fixed, each vertex permutation in Γ maps the path's edge set to itself and each color permutation in Γ fixes color 0, so Γ preserves the fixed path; every g preserves goodness. (With nothing fixed, S is the set of all good colorings.) Take X^* to be the $\le_{\text{lex}}$-least coloring reachable from a member of S by finitely many moves from Γ; it lies in S, and $g \cdot X^*$ is reachable too, so $X^* \le_{\text{lex}} g \cdot X^*$. □

The point of Lemma 4.2 is that the lex-leader constraints for *every* generator hold simultaneously for one coloring that also has the fixed path; the two reductions are therefore jointly sound. The generators do not need to generate the full stabilizer of the fixed path: a weaker set of constraints only leaves more symmetry in the search.

The base formula consists of the coloring clauses, the fixed-path unit clauses, and the clauses encoding these symmetry inequalities.

4.4 Encoding the symmetry inequalities

The constraint $X \le_{\text{lex}} g \cdot X$ is encoded in the standard way [6, Proposition 3.1]. The clauses below adapt that Boolean-variable construction to three colors. The positions compared are the edges f in edge order, except those with $\pi^{-1}(f) = f$ and σ the identity (where $X(f) = (g \cdot X)(f)$ identically). For the j-th compared position f_j, let $Y_j = \sigma(X(\pi^{-1} f_j))$ be the color of $g \cdot X$ there, and let q_j be an auxiliary variable meaning "X and $g \cdot X$ agree on the first j compared positions" (q_0 is true and omitted). The clauses are, for colors a, b:

$$q_{j-1} \to \neg\big(X(f_j) = a \wedge Y_j = b\big) \quad (a > b), \qquad q_{j-1} \wedge X(f_j) = a \wedge Y_j = a \to q_j,$$
$$q_j \to q_{j-1}, \qquad q_j \wedge X(f_j) = a \to Y_j = a.$$

Under the intended meaning of the q_j, every clause holds exactly when $X \le_{\text{lex}} g \cdot X$. Its soundness is proved in Lean: assigning the auxiliaries their intended values extends every coloring satisfying the inequalities to a model of the encoded clauses.

4.5 Successive path bounds

The formula for P_m on K_n with fixed P_k is the formula of Sections 4.1–4.4 with the fixed path on $0,\dots,k-1$, together with all path clauses for P_m. If it is unsatisfiable and every 3-coloring of K_n contains a monochromatic P_k, then by Lemmas 4.1 and 4.2 every 3-coloring of K_n contains a monochromatic P_m. Every 3-coloring of a larger complete graph then does too, by restriction.

The sequence of bounds is

$$R_3(P_3) \le 5, \quad R_3(P_4) \le 6, \quad R_3(P_5) \le 9, \quad R_3(P_6) \le 10,$$
$$R_3(P_7) \le 13, \quad R_3(P_8) \le 14, \quad R_3(P_9) \le 17.$$

The first formula fixes nothing; each later formula fixes the path obtained at the preceding stage. Each is refuted as a whole. The final formula, for P_{10} on K_{18}, fixes P_9 and is divided into cubes. The formula sizes and smaller-order checks are in Appendix A.

4.6 Covering the cases

We split the possible colorings into cases and certify completeness separately. A cube is a conjunction of literals fixing the colors of some edges. Unit propagation forces the last unassigned literal of a clause when all its other literals are false. Walking through the edges in order, the CaDiCaL solver of Biere et al. [3] tries each color and propagates the consequences, including path clauses, through Fazekas et al.'s IPASIR-UP interface [12, 13]. Choices leading to a conflict are discarded; an edge with only one surviving color is fixed without counting as a split. The walk stops after ten edges with at least two surviving colors. This gives 15,316 cubes. Continuing the same procedure, with conflict rejection and unit propagation, splits sixteen of them on four further split edges, giving 41 surviving children each and 656 in total. The final list has $15{,}316 - 16 + 656 = 15{,}956$ cubes.

The discarded branches are accounted for by the cover formula: it consists of the base formula, the 162 path clauses learned during generation, and the negation of every final cube (one clause per cube). The computational facts used below are that this cover formula and the formula for each cube, augmented by the path clauses used in its refutation, are unsatisfiable. For each formula, Lean's verified LRAT checker accepts a propositional refutation. The resulting 15,957 certificates, together with the seven smaller-formula certificates, give 15,964 in all. The following lemma combines the cover refutation with the cube refutations. In this lemma Q denotes a cube, distinct from the color symbol c, and $\mathcal{P}$ includes every path clause for the given path order.

Lemma 4.3. *Let F be a formula, $\mathcal{C}$ a list of cubes, and $\mathcal{P}$ the set of all path clauses. If $F \cup \mathcal{P} \cup \{\neg Q : Q \in \mathcal{C}\}$ is unsatisfiable and $F \cup \mathcal{P} \cup Q$ is unsatisfiable for every $Q \in \mathcal{C}$, then $F \cup \mathcal{P}$ is unsatisfiable.*

Proof. A model of $F \cup \mathcal{P}$ satisfies some cube $Q \in \mathcal{C}$, since otherwise it falsifies a literal of every cube and so satisfies every $\neg Q$. It is then a model of $F \cup \mathcal{P} \cup Q$. □

Proof of Theorem 1.1. Suppose a 3-coloring X of K_{18} has no monochromatic P_{10}. Since $R_3(P_9) \leq 17$ by the successive bounds above, X contains a monochromatic P_9. By Lemma 4.1 a vertex permutation and a color permutation carry X to a good coloring with color 0 on the path $0, 1, \ldots, 8$, and by Lemma 4.2 its orbit under the generators contains a good coloring X^* that keeps the fixed path and satisfies every lex-leader inequality.

With the auxiliaries set to their meaning, X^* satisfies the base formula and every path clause. By the cover certificate, X^* satisfies some cube of the final list (otherwise it would satisfy the cover formula), and that cube's formula is unsatisfiable by its own certificate: a contradiction. Thus $R_3(P_{10}) \leq 18$, while Proposition 3.1 gives $R_3(P_{10}) \geq 18$. □

5 Computational verification

The Lean development proves the following formal statement:

```
R3P10.R3_P10_eq_18 :  ramsey3Path 10 = 18.
```

Its colorings assign one of the colors $0, 1, 2$ to every unordered pair of vertices, including unused diagonal pairs. Relabeling the colors as $1, 2, 3$ gives the convention of Theorem 1.1. A formal monochromatic path is an injective map from $\{0, \ldots, m-1\}$ whose consecutive pairs all have one

color, with no condition on the other pairs. Thus it is the same non-induced path on m vertices used here. The formal Ramsey number is the natural-number infimum of the orders at which every coloring contains such a path. The proof shows that this set contains 18 and excludes every smaller order by restricting the countercoloring on 17 vertices. The final theorem has no certificate hypotheses; the finite coloring in its lower bound is exactly the construction of Section 3.

Lean proves the reductions, the lower bound and the soundness of the certificate checker. The finite certificate checks are evaluated by compiled code, so their validity also relies on faithful compiled evaluation. The checking notes specify the logical assumptions and reproduction instructions. The refutations were checked twice. The archived certificates [17] permit checking without repeating the search.

Appendix A Computational details

The details below specify the finite formulas, case division and checks used in Section 4.

A.1 Formula sizes

For K_{18} with a fixed P_9 there are 38 generators (36 transpositions of the nine vertices of W, the reversal, and the color swap), and the base formula has 1,868 variables (459 color variables and 1,409 auxiliaries) and 14,672 clauses.

A.2 Smaller path bounds

Table 1 lists the formulas. The first fixes nothing; each later formula fixes the path guaranteed by the preceding stage. A DRAT proof is a clause-addition and deletion refutation, converted to the line-indexed LRAT format for checking. Each formula up to P_9 on K_{17} is solved as one formula, with a DRAT proof. The formulas establish the upper bounds $R_3(P_m) \le n$ for $m = 3, \dots, 9$; the final formula, P_{10} on K_{18}, uses $R_3(P_9) \le 17$ as established by the formula below it. The earlier P_9 bound is Theorem 8 of Dybizbański, Dzido and Radziszowski [8].

Formula	fixed path	variables	base clauses	logged path clauses
P_3 on K_5	none	110	828	11
P_4 on K_6	P_3	92	527	31
P_5 on K_9	P_4	308	2,135	185
P_6 on K_{10}	P_5	368	2,502	1,683
P_7 on K_{13}	P_6	828	6,234	10,094
P_8 on K_{14}	P_7	928	6,897	60,734
P_9 on K_{17}	P_8	1,720	13,633	520,545
P_{10} on K_{18}	P_9	1,868	14,672	(per cube)

Table 1: Successive upper bounds. Each formula from P_3 to P_9 is refuted as a whole; the largest of these seven takes a few CPU-minutes. The K_{18} formula is split into cubes (Section 4.6).

The same code also finds colorings where they exist. For example it finds good colorings for P_5 on K_8, P_6 on K_9, P_7 on K_{12} and P_8 on K_{13}, and, with no fixed path, a coloring of K_{16} with no monochromatic P_9; each was confirmed by a separate exhaustive path search.

A.3 Certificate sizes

The initial cubes have 11 to 20 literals; the sixteen selected parents have 11 literals each. Generation of the initial cubes and their children uses 18 and 144 path clauses respectively, for a total of 162. The cover formula has 30,790 clauses: the 14,672 base clauses, these 162 path clauses, and one negated clause for each of the 15,956 cubes.

References

[1] P. Allen, J. Böttcher, D. Clemens, F. Hamann, J. Skokan and A. Taraz, *On Ramsey-type problems for paths and cycles with few colour changes*, preprint, 2026. arXiv:2607.03243v1.

[2] H. Böving, S. Bhat, L. Cicolini, A. Keizer, L. Frenot, A. Mohamed, L. Stefanesco, H. Khan, J. Clune, C. Barrett and T. Grosser, *Interactive bitvector reasoning using verified bit-blasting*, Proc. ACM Program. Lang. **9** (OOPSLA2) (2025), 3259–3285. doi:10.1145/3763167.

[3] A. Biere, T. Faller, K. Fazekas, M. Fleury, N. Froleyks and F. Pollitt, *CaDiCaL 2.0*, in: Computer Aided Verification (CAV 2024), Lecture Notes in Comput. Sci. **14681**, Springer, 2024, 133–152. doi:10.1007/978-3-031-65627-9_7.

[4] F. S. Benevides and J. Skokan, *The* 3*-colored Ramsey number of even cycles*, J. Combin. Theory Ser. B **99** (2009), 690–708. doi:10.1016/j.jctb.2008.12.002.

[5] M. Codish, M. Frank, A. Itzhakov and A. Miller, *Computing the Ramsey number* $R(4,3,3)$ *using abstraction and symmetry breaking*, Constraints **21** (2016), 375–393. doi:10.1007/s10601-016-9240-3; author preprint arXiv:1510.08266.

[6] J. M. Crawford, M. L. Ginsberg, E. M. Luks and A. Roy, *Symmetry-breaking predicates for search problems*, in: Proc. Fifth International Conference on Principles of Knowledge Representation and Reasoning (KR'96), Morgan Kaufmann, 1996, 148–159; technical report TR-1996-012.

[7] L. Cruz-Filipe, M. J. H. Heule, W. A. Hunt, M. Kaufmann and P. Schneider-Kamp, *Efficient certified RAT verification*, in: Automated Deduction – CADE 26, Lecture Notes in Artificial Intelligence **10395**, Springer, 2017, 220–236. doi:10.1007/978-3-319-63046-5_14.

[8] J. Dybizbański, T. Dzido and S. Radziszowski, *On some three-color Ramsey numbers for paths*, Discrete Appl. Math. **204** (2016), 133–141. doi:10.1016/j.dam.2015.11.002; preprint arXiv:1207.5851v3 (2015; first posted 2012).

[9] T. Dzido and R. Fidytek, *On some three color Ramsey numbers for paths and cycles*, Discrete Math. **309** (2009), 4955–4958. doi:10.1016/j.disc.2008.04.053.

[10] T. Dzido, *Multicolor Ramsey numbers for paths and cycles*, Discuss. Math. Graph Theory **25** (2005), 57–65. doi:10.7151/dmgt.1260.

[11] P. Erdős and T. Gallai, *On maximal paths and circuits of graphs*, Acta Math. Acad. Sci. Hungar. **10** (1959), 337–356. doi:10.1007/BF02024498.

[12] K. Fazekas, A. Niemetz, M. Preiner, M. Kirchweger, S. Szeider and A. Biere, *IPASIR-UP: User propagators for CDCL*, in: 26th International Conference on Theory and Applications of Satisfiability Testing (SAT 2023), LIPIcs **271**, 2023, 8:1–8:13. doi:10.4230/LIPIcs.SAT.2023.8.

[13] K. Fazekas, A. Niemetz, M. Preiner, M. Kirchweger, S. Szeider and A. Biere, *Satisfiability modulo user propagators*, J. Artificial Intelligence Res. **81** (2024), 989–1017. doi:10.1613/jair.1.16163.

[14] A. Figaj and T. Łuczak, *The Ramsey number for a triple of long even cycles*, J. Combin. Theory Ser. B **97** (2007), 584–596. doi:10.1016/j.jctb.2006.09.001.

[15] R. J. Faudree and R. H. Schelp, *Path Ramsey numbers in multicolorings*, J. Combin. Theory Ser. B **19** (1975), 150–160. doi:10.1016/0095-8956(75)90080-5.

[16] N. F. Galindez, *An upper bound for the three-colour Ramsey number of the ten-cycle:* $R(C_{10}, C_{10}, C_{10}) \leq 26$, draft note, August 18, 2026, Zenodo, version 1.0.0. doi:10.5281/zenodo.22000000.

[17] P. Graham, *Certificates for the three-color Ramsey number of the 10-vertex path*, dataset, Zenodo, 2026. doi:10.5281/zenodo.23562432.

[18] A. Gyárfás, M. Ruszinkó, G. N. Sárközy and E. Szemerédi, *Three-color Ramsey numbers for paths*, Combinatorica **27** (2007), 35–69. doi:10.1007/s00493-007-0043-4; author manuscript dated July 9, 2012, https://www.renyi.hu/~gyarfas/Cikkek/120_sharp3cfin.pdf; corrigendum, Combinatorica **28** (2008), 499–502. doi:10.1007/s00493-008-2395-9.

[19] M. J. H. Heule, O. Kullmann and V. W. Marek, *Solving and verifying the Boolean Pythagorean triples problem via cube-and-conquer*, in: Theory and Applications of Satisfiability Testing – SAT 2016, Lecture Notes in Comput. Sci. **9710**, Springer, 2016, 228–245; author preprint arXiv:1605.00723. doi:10.1007/978-3-319-40970-2_15.

[20] M. J. H. Heule, O. Kullmann, S. Wieringa and A. Biere, *Cube and conquer: Guiding CDCL SAT solvers by lookaheads*, in: Hardware and Software: Verification and Testing (HVC 2011), Lecture Notes in Comput. Sci. **7261**, Springer, 2012, 50–65; author manuscript https://www.cs.cmu.edu/~mheule/publications/cube.pdf. doi:10.1007/978-3-642-34188-5_8.

[21] M. J. H. Heule and M. Scheucher, *Happy ending: An empty hexagon in every set of 30 points*, in: Tools and Algorithms for the Construction and Analysis of Systems (TACAS 2024), Lecture Notes in Comput. Sci. **14570**, Springer, 2024, 61–80. doi:10.1007/978-3-031-57246-3_5.

[22] R. W. Irving, *Generalised Ramsey numbers for small graphs*, Discrete Math. **9** (1974), 251–264. doi:10.1016/0012-365X(74)90008-9.

[23] M. Kirchweger, P. Manrique and S. Szeider, *Formally verified graph generation with SAT modulo symmetries and Lean*, in: Automated Reasoning (IJCAR 2026), Lecture Notes in Artificial Intelligence **16688**, Springer, 2026, 117–135. doi:10.1007/978-3-032-32589-1_8.

[24] G. N. Kopylov, *Maximal paths and cycles in a graph*, Dokl. Akad. Nauk SSSR **234** (1977), 19–21; English translation: Soviet Math. Dokl. **18** (1977), 593–596.

[25] M. Kirchweger and S. Szeider, *SAT modulo symmetries for graph generation and enumeration*, ACM Trans. Comput. Log. **25**(3) (2024), Article 18. doi:10.1145/3670405.

[26] S. Letzter, *An improvement on Łuczak's connected matchings method*, Bull. Lond. Math. Soc. **54** (2022), 609–623. doi:10.1112/blms.12587.

[27] L. Maherani, G. R. Omidi, G. Raeisi and M. Shahsiah, *On three-color Ramsey number of paths*, Graphs Combin. **31** (2015), 2299–2308. doi:10.1007/s00373-014-1507-0; author preprint arXiv:1207.3771v1 (2012).

[28] The mathlib Community, *The Lean mathematical library*, in: Proc. 9th ACM SIGPLAN International Conference on Certified Programs and Proofs (CPP 2020), ACM, 2020, 367–381. doi:10.1145/3372885.3373824.

[29] L. de Moura and S. Ullrich, *The Lean 4 theorem prover and programming language*, in: Automated Deduction – CADE 28, Lecture Notes in Artificial Intelligence **12699**, Springer, 2021, 625–635. doi:10.1007/978-3-030-79876-5_37.

[30] G. R. Omidi and G. Raeisi, *On multicolor Ramsey number of paths versus cycles*, Electron. J. Combin. **18** (2011), #P24. doi:10.37236/511.

[31] S. P. Radziszowski, *Small Ramsey numbers*, Electron. J. Combin., Dynamic Survey DS1, revision #18, April 24, 2026. doi:10.37236/21.

[32] Z. Shao, E. Zhu, J. Xu and S. Kosari, *Reproducibility materials for The three-color Ramsey number of the ten-vertex path*, Zenodo, version 1.0.1, November 2, 2026. doi:10.5281/zenodo.23881576.

[33] Z. Shao, H. Jiang, X. Xu and S. Wang, *Native Windows reproduction package:* $R(C_4, C_6, C_8) = 12$, Zenodo, 2026. doi:10.5281/zenodo.23230189.

[34] B. Subercaseaux, W. Nawrocki, J. Gallicchio, C. Codel, M. Carneiro and M. J. H. Heule, *Formal verification of the empty hexagon number*, in: 15th International Conference on Interactive Theorem Proving (ITP 2024), LIPIcs **309**, 2024, 35:1–35:19. doi:10.4230/LIPIcs.ITP.2024.35.

[35] S. Szeider, *Streaming LRAT certificates into Lean theorems*, preprint, 2026. arXiv:2607.00815v2.

[36] Y. K. Tan, M. J. H. Heule and M. O. Myreen, *cake_lpr: Verified propagation redundancy checking in CakeML*, in: Tools and Algorithms for the Construction and Analysis of Systems (TACAS 2021), Lecture Notes in Comput. Sci. **12652**, Springer, 2021, 223–241. doi:10.1007/978-3-030-72013-1_12.

[37] W. J. Wesley, *New bounds for some small multicolor Ramsey numbers*, preprint, 2025. arXiv:2509.03784v1.

[38] N. Wetzler, M. J. H. Heule and W. A. Hunt, *DRAT-trim: Efficient checking and trimming using expressive clausal proofs*, in: Theory and Applications of Satisfiability Testing – SAT 2014, Lecture Notes in Comput. Sci. **8561**, Springer, 2014, 422–429. doi:10.1007/978-3-319-09284-3_31.

[39] Yang Yuansheng, *On the third Ramsey numbers of graphs with six edges*, J. Combin. Math. Combin. Comput. **17** (1995), 199–208.

[40] Yang Yuansheng and P. Rowlinson, *The third Ramsey numbers for graphs with at most four edges*, Discrete Math. **125** (1994), 399–406. doi:10.1016/0012-365X(94)90182-1.

[41] Yang Yuansheng and P. Rowlinson, *On the third Ramsey numbers of graphs with five edges*, J. Combin. Math. Combin. Comput. **11** (1992), 213–222.

[42] Leo-Y-Zhang, *CycleRamsey*, software repository on GitHub, 2026. `https://github.com/Leo-Y-Zhang/CycleRamsey`.

[43] J. Zhang, *erdos557*, `Erdos557/Tight.lean`, `multicolourRamsey_le_tight`, commit `9d26a79d9ad04ff605167d022e512984a76ea3fa`, October 10, 2026. `https://github.com/zhangjun725/erdos557/blob/9d26a79d9ad04ff605167d022e512984a76ea3fa/Erdos557/Tight.lean`. The pinned README credits an AI-found proof packaged by T. Adamczewski in *erdos548*, `https://github.com/tadamcz/erdos548`.