

THE COVERING NUMBER $C(16, 5, 3)$ IS AT LEAST 62

CHRISTOPHER B. CELAYA

ABSTRACT. A (v, k, t) covering is a family of k -subsets (blocks) of a v -set such that every t -subset lies in at least one block, and the covering number $C(v, k, t)$ is the least number of blocks in such a family. The Schönheim bound gives $C(16, 5, 3) \geq 61$, and a covering with 65 blocks has been known since 1997; these were the best recorded bounds. We prove that $C(16, 5, 3) \geq 62$. Counting shows that a 61-block covering would be very rigid: one point lies in 20 blocks and every other point in 19, every triple is covered once or twice, the pairs covered six times form a five-edge star plus a perfect matching of the remaining ten points, and the derived design at a point of degree 19 is a minimum $(15, 4, 2)$ covering. Minimum $(15, 4, 2)$ coverings fall into exactly four isomorphism classes, as Allston, Buskens and Stanton showed; we confirm this by two independent exhaustive computations. Fixing the derived design at a suitable point to each class in turn leaves four Boolean satisfiability instances, each with 144,553 variables and 512,424 clauses. All four are unsatisfiable: the SAT solver Lingeling refutes them with DRAT proofs of 13.0 to 15.6 million lines that the proof checker drat-trim verifies, and the solver CaDiCaL and a separately written constraint programming model agree. The encoder, the formulas, the proofs and the checking logs are publicly available.

1. INTRODUCTION

Let $v \geq k \geq t \geq 1$ be integers. A (v, k, t) *covering* is a family $\mathcal{B}$ of k -subsets, called *blocks*, of a v -set V such that every t -subset of V is contained in at least one block. The *covering number* $C(v, k, t)$ is the least size of a (v, k, t) covering. We refer to the survey of Gordon and Stinson [12] for background, and to the La Jolla Covering Repository [10], now continued by the Covering Repository [1], for tables of bounds. Taking complements, $C(v, k, t)$ equals the Turán number $T(v, v - t, v - k)$, the least number of $(v - k)$ -subsets of a v -set such that every $(v - t)$ -subset contains one of them [11, 12]; for Turán numbers in general see the survey [23].

The basic lower bound is due to Schönheim [22]:

$$C(v, k, t) \geq L(v, k, t) := \left\lceil \frac{v}{k} L(v - 1, k - 1, t - 1) \right\rceil, \quad L(v, k, 0) = 1.$$

2020 *Mathematics Subject Classification.* Primary 05B40; Secondary 05B05, 68V05, 68V15.

Key words and phrases. covering design, covering number, Schönheim bound, Turán number, SAT solving, DRAT proof, computer-assisted proof.

For $(v, k, t) = (16, 5, 3)$ it gives $L(16, 5, 3) = \lceil \frac{16}{5} \lceil \frac{15}{4} \lceil \frac{14}{3} \rceil \rceil \rceil = \lceil \frac{16}{5} \cdot 19 \rceil = 61$. A covering with 65 blocks was found by Rade Belić in 1997 [10], and Bluskov obtained the same bound by a construction from the Steiner system $S(3, 5, 17)$ [6, Corollary 2.3.18]. The bounds $61 \leq C(16, 5, 3) \leq 65$ are those recorded by the covering repositories [10, 1]. General improvements of the Schönheim bound give nothing more here: the hypotheses of the Mills–Mullin improvement [20] fail for these parameters, and the bounds of Horsley and Singh [14] do not exceed 61. We know of no earlier lower bound above 61 for $C(16, 5, 3)$ or, equivalently, for $T(16, 13, 11)$. Our main result improves the lower bound.

Theorem 1.1. *There is no $(16, 5, 3)$ covering with 61 blocks. Hence $62 \leq C(16, 5, 3) \leq 65$.*

The proof has a human part and a computer part. In Section 3, elementary counting shows that a 61-block covering is extremely constrained: one point lies in 20 blocks and every other point in 19; the pairs covered six times rather than five form a graph consisting of a five-edge star at the point of degree 20 and a perfect matching of the remaining ten points; every triple is covered once or twice, with a prescribed number of doubly covered triples through each pair; and the derived design at a point of degree 19 is a minimum $(15, 4, 2)$ covering. Minimum $(15, 4, 2)$ coverings have 19 blocks and fall into four isomorphism classes (Section 4). This classification is due to Allston, Buskens and Stanton [2]; since our argument depends on it, we confirm it by two independent exhaustive computations. In Section 5 we fix the derived design at a suitable point to one of the four representatives. Each of the four resulting problems becomes a propositional formula in conjunctive normal form (CNF), constructed so that every 61-block covering yields a satisfying assignment of one of them. In Section 6 we report that all four formulas are unsatisfiable. The refutations were produced by the SAT solver Lingeling [4] as DRAT proofs and verified by the proof checker drat-trim [26]; the SAT solver CaDiCaL [5] and a separately written CP-SAT model [21] give the same answer.

Computer-assisted proofs of this kind are common in combinatorics. The value $C(10, 5, 4) = 51$ was determined by Margot [17], with branch-and-cut and isomorphism pruning, and independently by Applegate, Rains and Sloane [3], who also determined several further covering numbers. SAT solvers with independently checkable proofs settled the Boolean Pythagorean triples problem [13] and Keller’s conjecture in dimension seven [7]. Closest to the present paper, Krug [16] recently proved $C(12, 6, 4) = 41$ by the same overall strategy: counting forces the structure of a putative smaller covering, every derived design is an optimal covering of smaller parameters, and the remaining cases are refuted by a SAT solver with certified proofs.

All code, formulas, logs and proofs are public (Section 8).

2. NOTATION AND BASIC COUNTING

Throughout, $\mathcal{B}$ is a family of 5-subsets of a 16-set V , a priori with repetitions. For a point x , a pair $\{x, y\}$ and a triple T we write r_x , λ_{xy} and μ_T for the numbers of blocks containing x , $\{x, y\}$ and T , counted with multiplicity, and call r_x the *degree* of x . The family $\mathcal{B}$ is a $(16, 5, 3)$ covering if and only if $\mu_T \geq 1$ for every triple T . For a point x , the *derived design* at x is the family (with multiplicities)

$$\mathcal{D}_x = \{B \setminus \{x\} : x \in B \in \mathcal{B}\}$$

of r_x quadruples on $V \setminus \{x\}$. A pair $\{u, w\} \subseteq V \setminus \{x\}$ lies in exactly μ_{xuw} members of $\mathcal{D}_x$, and a point $u \neq x$ lies in exactly λ_{xu} of them. In particular, if $\mathcal{B}$ is a $(16, 5, 3)$ covering then $\mathcal{D}_x$ is a $(15, 4, 2)$ covering. Two families are *isomorphic* if a bijection between their point sets carries one onto the other.

Lemma 2.1. *Let $\mathcal{B}$ be a $(16, 5, 3)$ covering. Then $\lambda_{xy} \geq 5$ for every pair and $r_x \geq 19$ for every point.*

Proof. Each of the 14 points $w \notin \{x, y\}$ needs a block containing $\{x, y, w\}$, and a block through $\{x, y\}$ contains only three further points, so $\lambda_{xy} \geq \lceil 14/3 \rceil = 5$. Each block through x contains four pairs through x , so $4r_x = \sum_{y \neq x} \lambda_{xy} \geq 15 \cdot 5 = 75$ and $r_x \geq 19$. $\square$

Since $\sum_x r_x = 5|\mathcal{B}|$, Lemma 2.1 gives $5|\mathcal{B}| \geq 16 \cdot 19$, that is $|\mathcal{B}| \geq 61$; this is the Schönheim bound. In particular a 61-block covering has no repeated block, since deleting one copy of a repeated block would leave a covering with 60 blocks.

3. THE STRUCTURE OF A 61-BLOCK COVERING

In this section $\mathcal{B}$ is a $(16, 5, 3)$ covering with exactly 61 blocks; by the remark above, its blocks are distinct.

Lemma 3.1. *There is a point z with $r_z = 20$, and $r_x = 19$ for every $x \neq z$.*

Proof. The degrees sum to $5 \cdot 61 = 305 = 16 \cdot 19 + 1$ and each is at least 19 by Lemma 2.1. $\square$

For a pair $\{x, y\}$ put $e_{xy} = \lambda_{xy} - 5 \geq 0$. Then

$$(1) \quad \sum_{y \neq x} e_{xy} = 4r_x - 75 = \begin{cases} 1 & \text{if } r_x = 19, \\ 5 & \text{if } x = z. \end{cases}$$

Lemma 3.2. *Every pair has $\lambda_{xy} \in \{5, 6\}$. Let E be the graph on V whose edges are the pairs with $\lambda_{xy} = 6$. Then there are a 5-set $A \subseteq V \setminus \{z\}$ and a perfect matching P of $M = V \setminus (A \cup \{z\})$ such that*

$$E = \{\{z, a\} : a \in A\} \cup P.$$

In particular every point other than z has exactly one neighbour in E , and E has ten edges and no triangle.

Proof. By (1), a point $x \neq z$ has $e_{xy} = 1$ for exactly one y and $e_{xy} = 0$ for all other y . In particular $e_{zy} \leq 1$ for every $y \neq z$, so the five units of excess at z lie on five pairs $\{z, a\}$ with a in a 5-set A . Each $a \in A$ has used its single unit on $\{z, a\}$, so it has no other neighbour in E . Every $x \in M$ therefore has its unique neighbour in M , and these pairs form a perfect matching of M . $\square$

For a pair $\{x, y\}$, each block through $\{x, y\}$ contains three triples through $\{x, y\}$, so $\sum_w \mu_{xyw} = 3\lambda_{xy}$, where w runs over the 14 other points. Hence

$$(2) \quad \sum_{w \notin \{x, y\}} (\mu_{xyw} - 1) = 3\lambda_{xy} - 14 = \begin{cases} 1 & \text{if } \{x, y\} \notin E, \\ 4 & \text{if } \{x, y\} \in E. \end{cases}$$

Call a triple *doubled* if $\mu_T = 2$.

Lemma 3.3. *Every triple T has $\mu_T \in \{1, 2\}$. Each pair outside E lies in exactly one doubled triple, each pair in E lies in exactly four doubled triples, and there are exactly 50 doubled triples.*

Proof. If $\mu_T \geq 3$, then T contributes at least 2 to the left side of (2) for each of its three pairs. A pair outside E has total 1 there, so all three pairs of T lie in E ; but E has no triangle. So $\mu_T \leq 2$ for all T , and (2) counts the doubled triples through each pair. Let D be the number of doubled triples. Counting incidences between pairs and doubled triples gives $3D = 110 \cdot 1 + 10 \cdot 4$, so $D = 50$. (As a check, $10 \cdot 61 = \binom{16}{3} + D$.) $\square$

Lemma 3.4. *Let $x \neq z$ and let p be the neighbour of x in E . Then $\mathcal{D}_x$ is a $(15, 4, 2)$ covering with 19 distinct blocks, in which p lies in six blocks and every other point in five. The pairs that lie in two blocks of $\mathcal{D}_x$ are the pairs $\{u, w\}$ for which $\{x, u, w\}$ is doubled; they form a star with centre p and four leaves together with a perfect matching of the remaining ten points, and every other pair lies in exactly one block of $\mathcal{D}_x$.*

Proof. $\mathcal{D}_x$ has $r_x = 19$ blocks, and they are distinct because the blocks of $\mathcal{B}$ are. The degree of u in $\mathcal{D}_x$ is λ_{xu} , which is 6 for $u = p$ and 5 otherwise. The multiplicity of $\{u, w\}$ in $\mathcal{D}_x$ is $\mu_{xuw} \in \{1, 2\}$. By Lemma 3.3, $\{x, p\}$ lies in four doubled triples and every other pair $\{x, u\}$ in exactly one, so in the graph of pairs covered twice by $\mathcal{D}_x$ the point p has degree four and every other point degree one. Such a graph is a star at p with four leaves together with a perfect matching of the other ten points. $\square$

We record what the reduction in Section 5 uses.

Proposition 3.5. *If $\mathcal{B}$ is a 61-block $(16, 5, 3)$ covering, then its blocks are distinct and, with z , A , M and E as above:*

- (N1) *every triple T satisfies $1 \leq \mu_T \leq 2$;*
- (N2) *every pair outside E lies in exactly one doubled triple, and every pair in E in exactly four;*

- (N3) E consists of the five pairs $\{z, a\}$, $a \in A$, and a perfect matching of M ;
- (N4) for every $a \in A$, the derived design $\mathcal{D}_a$ is a $(15, 4, 2)$ covering with 19 blocks in which z lies in six blocks.

Remark 3.6. The counting gives more, although we do not use it. The doubled triples through z form a graph X_z on $V \setminus \{z\}$ in which every point of A has degree 4 and every point of M has degree 1. If X_z has n_2 edges inside A , it has $20 - 2n_2$ edges between A and M and $n_2 - 5$ edges inside M , so $5 \leq n_2 \leq 10$.

4. MINIMUM $(15, 4, 2)$ COVERINGS

It is classical that $C(15, 4, 2) = 19$ [18, 19]; the Schönheim bound is $\lceil \frac{15}{4} \lceil \frac{14}{3} \rceil \rceil = 19$. The argument of Lemma 3.4 applies to every minimum $(15, 4, 2)$ covering.

Lemma 4.1. *Let $\mathcal{D}$ be a $(15, 4, 2)$ covering with 19 blocks, a priori with repetitions. Then one point h , the hub, lies in six blocks and every other point in five. No pair lies in three blocks. The pairs lying in two blocks form a star with centre h and four leaves together with a perfect matching of the other ten points, and every other pair lies in exactly one block. In particular the blocks of $\mathcal{D}$ are distinct.*

Proof. Each point lies in at least $\lceil 14/3 \rceil = 5$ blocks, and the degrees sum to $76 = 15 \cdot 5 + 1$, so there is exactly one point of degree 6. For a point u of degree d_u we have $\sum_{w \neq u} (m_{uw} - 1) = 3d_u - 14$, where m_{uw} is the number of blocks containing $\{u, w\}$; this is 4 at the hub and 1 elsewhere. A pair in three blocks would contribute 2 at each of its ends, which is impossible because at most one end is the hub. So all $m_{uw} \leq 2$, and in the graph of pairs with $m_{uw} = 2$ the hub has degree four and every other point degree one; such a graph is a star at the hub with four leaves together with a perfect matching of the other ten points. A repeated block would put all six of its pairs in two blocks, but the graph of such pairs has no triangle. $\square$

We call the graph of pairs lying in two blocks the *excess graph* of $\mathcal{D}$; it is isomorphic to $K_{1,4} \cup 5K_2$. A 19-block $(15, 4, 2)$ covering on $\{2, \dots, 16\}$ is in *standard form* if its hub is 2, the leaves of its star are 3, 4, 5, 6, and its matching is $\{7, 8\}, \{9, 10\}, \{11, 12\}, \{13, 14\}, \{15, 16\}$. Let G be the group of permutations of $\{2, \dots, 16\}$ that preserve this standard excess graph; it has order $4! \cdot 5! \cdot 2^5 = 92,160$. Every 19-block $(15, 4, 2)$ covering is isomorphic to one in standard form. An isomorphism between two coverings preserves pair multiplicities and hence excess graphs, so an isomorphism between two coverings in standard form is an element of G . Isomorphism classes of 19-block $(15, 4, 2)$ coverings therefore correspond to G -orbits of coverings in standard form.

Theorem 4.2. *There are exactly four isomorphism classes of $(15, 4, 2)$ coverings with 19 blocks. Representatives $\mathcal{R}_1, \dots, \mathcal{R}_4$ in standard form are listed in Table 1; their automorphism groups have orders 4, 2, 2 and 4.*

That there are exactly four classes is due to Allston, Buskens and Stanton [2, Theorem 12], who call these designs D2, D1, B4 and B3 and report the same automorphism group orders. Explicit isomorphisms from their published block lists onto $\mathcal{R}_1, \dots, \mathcal{R}_4$ are in the repository [9]. Their classification used computer isomorphism testing. Only the first assertion of Theorem 4.2, that every 19-block $(15, 4, 2)$ covering is isomorphic to some $\mathcal{R}_i$, is used in the proof of Theorem 1.1. Since the proof rests on it, we proved Theorem 4.2 again by two exhaustive computations, written separately and sharing no code; each of Computations 1 and 2 below is a complete proof.

Computation 1: hub configurations. Let $\mathcal{D}$ be in standard form. The hub lies in six *hub blocks*, which have 18 further slots. Each leaf fills two of them, in different hub blocks, and each of the ten matched points fills one. Regard the six hub blocks as vertices; each leaf gives an edge of a graph Γ_L joining the two hub blocks containing it, and each matching pair $\{m, m'\}$ gives an edge of a multigraph Γ_M joining the hub blocks containing m and m' (a loop if they are equal). Every vertex has total degree three, a loop counting twice, and Γ_L is simple, since two leaves in the same two hub blocks would cover their pair twice. Two hub-block sets lie in the same G -orbit exactly when their coloured graphs (Γ_L, Γ_M) are isomorphic. There are 1335 labelled graphs Γ_L in eight isomorphism classes, giving 206 classes of coloured graphs (Γ_L, Γ_M) . For a representative of each class, the remaining 13 blocks avoid the hub and must cover a prescribed multigraph of residual pair demands exactly. A depth-first search that branches on the pair with fewest remaining candidate blocks enumerated every such decomposition; all 206 searches completed, with 31,989 nodes in total. Exactly four classes extend, with 12, 4, 2 and 96 completions; the other 202 have none. A separate program replayed every search tree and verified each branching, forced move and dead end, and a CP-SAT enumeration of the four feasible cases returned the same 114 completions. Within each feasible class, with hub-block set H , any isomorphism between completions lies in the stabilizer G_H of H in G ; these stabilizers have orders 48, 8, 4 and 384, and in each class the completions form a single G_H -orbit. Hence there are exactly four isomorphism classes, with automorphism groups of orders $48/12 = 4$, $8/4 = 2$, $4/2 = 2$ and $384/96 = 4$.

Computation 2: direct enumeration. A second program, written from scratch without reference to the first, enumerates all 19-block coverings in standard form directly, as an exact cover problem in which the nine excess pairs must be covered twice and the other 96 pairs once. Two depth-first searches with different branching rules both found the same $N = 138,240$

coverings. As a third check, the program listed all 5,745,600 labelled hub-block sets, which fall into 206 G -orbits, and for one representative H of each orbit CP-SAT enumerated all completions of H ; they were exactly the coverings with hub-block set H found by the depth-first searches. Since G maps the completions of H onto those of gH , this gives again $N = \sum_H |G \cdot H| c(H) = 1920 \cdot 12 + 11520 \cdot 4 + 23040 \cdot 2 + 240 \cdot 96 = 138,240$, where $c(H)$ is the number of completions of H . Applying all 92,160 elements of G partitions these coverings into four orbits, of sizes 23,040, 46,080, 46,080 and 23,040, so the automorphism groups have orders 4, 2, 2 and 4, and $N = \sum_i |G|/|\text{Aut}(\mathcal{R}_i)|$. The four representatives in Table 1 lie in the four different orbits.

TABLE 1. Representatives of the four classes of 19-block $(15, 4, 2)$ coverings, in standard form (hub 2, leaves 3, 4, 5, 6, matching $\{7, 8\}, \dots, \{15, 16\}$); each entry is a block. In the notation of [2] these are D2, D1, B4 and B3, and in the accompanying repository `shape-1`, `shape-4`, `shape-44` and `shape-47`.

$\mathcal{R}_1$	$\mathcal{R}_2$	$\mathcal{R}_3$	$\mathcal{R}_4$
2 3 4 5	2 3 4 5	2 3 4 5	2 3 4 5
2 3 6 7	2 3 6 7	2 3 6 7	2 3 7 8
2 4 6 8	2 4 6 8	2 4 9 11	2 4 9 10
2 5 9 10	2 5 9 11	2 5 10 13	2 5 11 12
2 11 13 15	2 10 13 15	2 6 15 16	2 6 13 14
2 12 14 16	2 12 14 16	2 8 12 14	2 6 15 16
3 8 11 12	3 8 13 14	3 8 9 10	3 6 9 10
3 9 13 16	3 9 10 16	3 11 12 15	3 11 13 14
3 10 14 15	3 11 12 15	3 13 14 16	3 12 15 16
4 7 13 14	4 7 15 16	4 6 10 12	4 6 11 12
4 9 12 15	4 9 12 13	4 7 8 16	4 7 13 15
4 10 11 16	4 10 11 14	4 13 14 15	4 8 14 16
5 6 15 16	5 6 10 12	5 6 9 14	5 6 7 8
5 7 11 12	5 7 13 14	5 7 8 15	5 9 13 16
5 8 13 14	5 8 15 16	5 11 12 16	5 10 14 15
6 9 11 14	6 9 14 15	6 8 11 13	7 9 12 14
6 10 12 13	6 11 13 16	7 9 12 13	7 10 11 16
7 8 9 10	7 8 9 10	7 10 11 14	8 9 11 15
7 8 15 16	7 8 11 12	9 10 15 16	8 10 12 13
$ \text{Aut} = 4$	$ \text{Aut} = 2$	$ \text{Aut} = 2$	$ \text{Aut} = 4$

5. REDUCTION TO FOUR SATISFIABILITY PROBLEMS

Suppose that $\mathcal{B}$ is a 61-block $(16, 5, 3)$ covering, and let z , A , M , E be as in Section 3. Choose $a \in A$. By Proposition 3.5(N4), $\mathcal{D}_a$ is a 19-block $(15, 4, 2)$

covering with hub z . By Theorem 4.2 there are an index $i \in \{1, 2, 3, 4\}$ and a bijection $\sigma: V \setminus \{a\} \rightarrow \{2, \dots, 16\}$ that carries $\mathcal{D}_a$ onto $\mathcal{R}_i$. It carries the hub z to the hub 2 of $\mathcal{R}_i$. Extend σ by $\sigma(a) = 1$ and replace $\mathcal{B}$ by its image. After this relabeling:

- $a = 1$ and $z = 2$, and the blocks through 1 are exactly the 19 blocks $\{1\} \cup R$ with $R \in \mathcal{R}_i$;
- $A = \{1\} \cup A'$ for a 4-subset $A' \subseteq \{3, \dots, 16\}$, and E consists of $\{1, 2\}$, the pairs $\{2, v\}$ with $v \in A'$, and a perfect matching of $\{3, \dots, 16\} \setminus A'$;
- conditions (N1) and (N2) hold.

What is unknown is A' , the matching, and the 42 blocks that avoid the point 1.

The formula F_i . For each i we define a CNF formula F_i with the following variables: x_B for each 5-subset B of $\{1, \dots, 16\}$ (4368 variables, meaning that B is a block); d_T for each 3-subset T (560 variables, meaning that T is doubled); a_v for $v \in \{3, \dots, 16\}$ (14 variables, meaning that $v \in A'$); and m_{uv} for each pair $\{u, v\} \subseteq \{3, \dots, 16\}$ (91 variables, meaning that $\{u, v\}$ is a matching edge). For a pair Q let ε_Q be the constant or literal expressing $Q \in E$:

$$\varepsilon_{\{1,2\}} = \top, \quad \varepsilon_{\{1,v\}} = \perp, \quad \varepsilon_{\{2,v\}} = a_v, \quad \varepsilon_{\{u,v\}} = m_{uv} \quad (3 \leq u, v \leq 16).$$

The constraints are:

- (C1) x_B is true for the 19 sets $B = \{1\} \cup R$ with $R \in \mathcal{R}_i$, and false for every other 5-set containing 1;
- (C2) for every triple T : $1 \leq \sum_{B \supseteq T} x_B \leq 2$, and d_T is true if and only if $\sum_{B \supseteq T} x_B \geq 2$;
- (C3) exactly four of the variables a_v are true;
- (C4) $m_{uv} \rightarrow \neg a_u$ and $m_{uv} \rightarrow \neg a_v$; every $v \in \{3, \dots, 16\}$ has at most one u with m_{uv} true; and a_v or some m_{uv} is true;
- (C5) for every pair Q : $\sum_{T \supseteq Q} d_T \geq 1$; if ε_Q holds then $\sum_{T \supseteq Q} d_T = 4$, and otherwise $\sum_{T \supseteq Q} d_T = 1$.

The cardinality conditions in (C2) and (C5) are encoded with a two-sided variant of the sequential counter of Sinz [24], in which each counter variable is equivalent to the statement that at least j of the first ℓ inputs are true: for each triple T a counter with three registers runs over the 78 variables x_B , $B \supseteq T$, and for each pair Q a counter with five registers runs over the 14 variables d_T , $T \supseteq Q$. Condition (C3) uses the sequential-counter encoding of PySAT [15], whose auxiliary variables we do not interpret, and the at-most-one conditions in (C4) use the pairwise encoding. In (C5), when ε_Q is a literal, the two cases are implications guarded by ε_Q and by $\neg \varepsilon_Q$. Each F_i has 144,553 variables and 512,424 clauses (Table 2). The four formulas differ only in the unit clauses of (C1).

TABLE 2. Variables and clauses of each formula F_i by constraint family. All counter variables are auxiliary.

Family	Variables	Clauses
Block variables x_B ; (C1) as unit clauses	4,368	1,365
Doubled-triple variables d_T	560	0
(C2) triple counters and their outputs	131,040	478,800
Variables a_v ; (C3) with 80 auxiliary variables	94	160
(C4) matching variables and clauses	91	1,288
(C5) pair counters and their outputs	8,400	30,811
Total	144,553	512,424

Proposition 5.1. *If a $(16, 5, 3)$ covering with 61 blocks exists, then at least one of the formulas $F_1, \dots, F_4$ is satisfiable.*

Proof. Relabel $\mathcal{B}$ as above, with derived design $\mathcal{R}_i$ at the point 1. Set x_B true exactly for the blocks of $\mathcal{B}$, d_T true exactly for the doubled triples, a_v true exactly for $v \in A'$, m_{uv} true exactly for the matching edges, every counter variable of (C2) and (C5) to the truth value of the statement it represents, and the auxiliary variables of (C3) to values satisfying its clauses, which exist because exactly four a_v are true (checked exhaustively; Section 6). Condition (C1) holds because the blocks of $\mathcal{B}$ are distinct and the blocks through 1 are the sets $\{1\} \cup R$, $R \in \mathcal{R}_i$; (C2) is (N1) together with the definition of a doubled triple; (C3) holds because $|A'| = 4$; (C4) holds because the points of A' have no neighbour in E other than 2, while every point of $\{3, \dots, 16\} \setminus A'$ has exactly one neighbour in E , and it lies in $\{3, \dots, 16\} \setminus A'$. For (C5), the description of E after relabeling shows that ε_Q is true exactly when $Q \in E$: the only neighbour of 1 in E is 2, the neighbours of 2 are 1 and the points of A' , and the edges of E inside $\{3, \dots, 16\}$ are the matching edges. Then (C5) is (N2). $\square$

Remark 5.2. Neither the number of blocks nor the pair multiplicities are written into F_i . They are implied, so the reduction loses nothing: in any satisfying assignment, (C3) and (C4) give a graph E with $1 + 4 + 5 = 10$ edges, so (C5) gives $3D = 110 + 40$ for the number D of doubled triples, and (C2) gives $10|\mathcal{B}| = \sum_T \mu_T = 560 + D = 610$, where $\mathcal{B}$ is the set of B with x_B true. For a pair Q , (C2) and (C5) give $3\lambda_Q = \sum_{w \notin Q} \mu_{Q \cup \{w\}} = 14 + 1 + 3[\varepsilon_Q]$, so $\lambda_Q = 5 + [\varepsilon_Q]$; hence $4r_2 = 15 \cdot 5 + 5$ and $r_2 = 20$. Conversely, if a 61-block covering has the sets $\{1\} \cup R$, $R \in \mathcal{R}_i$, as its blocks through 1 and its point of degree 20 is 2, then $\lambda_{12} = 6$ because 2 is the hub of $\mathcal{R}_i$, so $1 \in A$, and the proof of Proposition 5.1 applies with $a = 1$ and σ the identity. Thus F_i is satisfiable if and only if there is a 61-block covering whose blocks through 1 are the sets $\{1\} \cup R$, $R \in \mathcal{R}_i$, and whose point of degree 20 is 2.

6. COMPUTATION AND CERTIFICATION

Theorem 6.1. *The formulas F_1, F_2, F_3, F_4 are unsatisfiable.*

We recall the certification method. A DRAT proof of unsatisfiability of a CNF formula is a list of clause additions and deletions ending with the empty clause. Each added clause, or *lemma*, must be a reverse unit propagation (RUP) consequence of the clauses present at that point (assigning all its literals false and applying unit propagation yields a conflict), or more generally a resolution asymmetric tautology (RAT). Both kinds of addition preserve satisfiability and deletions cannot make an unsatisfiable set satisfiable, so a valid proof that reaches the empty clause shows that the formula is unsatisfiable. In backward mode, drat-trim starts from the empty clause and checks only the lemmas needed to derive it; these lemmas and the original clauses they use form the *core*. DIMACS is the standard text format for CNF formulas.

The formulas were generated by a short Python script using PySAT (version 1.9.dev15) [15] and written in DIMACS format. Each was solved by Lingeling (version bbc-9230380-160707, the SAT Competition 2016 version [4], as bundled with PySAT), which recorded a proof of unsatisfiability in the DRAT format. Each proof was then checked by drat-trim [26] (commit 2e3b2dc) in backward mode with the time limit raised (option `-t 1000000`), which reported `s VERIFIED` in all four cases. Table 3 gives the figures. drat-trim reports no RAT lemmas in any core, so every lemma in the verified cores is a RUP lemma. The runs were made on 5 October 2026 on one Apple M5 Max laptop with 18 cores and 128 GB of memory; the four cases ran concurrently with other jobs, so the times are wall-clock times on a shared machine and are given only for orientation.

TABLE 3. Refutation of $F_1, \dots, F_4$ by Lingeling and verification of the DRAT proofs by drat-trim. Times in seconds; sizes in decimal gigabytes. Solve is the wall-clock time of the Lingeling call through PySAT, including loading the clauses and retrieving the proof; check times and core figures are as reported by drat-trim.

	Solve	Proof lines	Size	Check	Core clauses	Core lemmas
F_1	14,714	13,065,165	7.04	5,269	223,884	1,914,196
F_2	19,495	15,609,452	8.82	3,512	228,423	2,143,298
F_3	19,640	15,471,588	8.48	3,439	235,146	2,179,394
F_4	14,774	13,044,849	7.30	4,829	230,704	1,765,503

Cross-checks. Two further computations agree with Theorem 6.1; neither is part of the proof. First, CaDiCaL 1.9.5 [5], also called through PySAT on the same formulas, reported unsatisfiability in 1,045 to 1,255 seconds

after 3.5 to 4.2 million conflicts; these runs produced no proofs. Second, a separately written CP-SAT model in Google OR-Tools 9.15 [21] states the same conditions directly as linear constraints ($\sum_{B \supseteq T} x_B = 1 + d_T$ for every triple, $\sum_{T \supseteq Q} d_T = 1 + 3\varepsilon_Q$ and, redundantly, $\sum_{B \supseteq Q} x_B = 5 + \varepsilon_Q$ for every pair, and the conditions on A' and the matching). It reported all four problems infeasible in 1,623 to 2,089 seconds. This model shares the reduction and the representatives with the CNF formulas, but none of the encoding: no counters, no cardinality encodings and no DIMACS output.

Checks on the formulas. A proof checker certifies that a formula is unsatisfiable, not that the formula says what it is meant to say. We therefore checked the encoding separately. The DIMACS files are regenerated byte for byte from the representatives alone by the script `gen_cnf.py` in the ancillary files, which consists of the encoding lines of the solve script without the solver call; their SHA256 checksums are listed with the data. That each counter variable is equivalent to its intended statement follows by induction on the number of inputs; we also tested the counter code exhaustively for up to 12 inputs with bounds up to 6, and for 14 inputs with the bounds 3, 5 and 6 (the bound 5 on 14 inputs is the case used in (C5)): in every case each assignment of the inputs extends uniquely, with every counter variable equal to its intended value. The clauses of (C3) and (C4) were checked exhaustively over all assignments of the a_v : the clauses of (C3), with their auxiliary variables, are satisfiable exactly for the 1001 assignments with four a_v true, and the solutions of (C4) on the m_{uv} are exactly the 945 perfect matchings of the complement of A' for each of the 1001 choices of A' . Finally, for 1200 random assignments of the main variables, extended to the auxiliary variables (the counters canonically, the auxiliary variables of (C3) by a satisfying extension), the violated constraint instances of each family were compared with the violated instances of (C1)–(C5); there were no mismatches. We also confirmed that every representative $\mathcal{R}_i$ is a 19-block $(15, 4, 2)$ covering in standard form with hub 2, as the definition of ε_Q requires. The scripts for these checks, with their output, are in the directory `experiments/2026-10-06/1b61-encoder-audit` of the repository [9].

What the proof relies on. Theorem 1.1 rests on four things. First, the arguments proved in the text: Lemma 2.1, the counting of Section 3, Lemma 4.1 and the reduction of Proposition 5.1. Second, the completeness part of Theorem 4.2. It has been obtained three times, by Allston, Buskens and Stanton and by the two computations of Section 4, which agree; the search trees of Computation 1 were replayed by a separate program, but unlike the DRAT proofs they are not checked by an established proof checker. Third, the correctness of the short encoder that produces $F_1, \dots, F_4$, including the PySAT cardinality encoding it calls for (C3), checked as described above. Fourth, the correctness of drat-trim. The proof does not rely on the correctness of Lingeling, CaDiCaL or CP-SAT. The DRAT proofs could also

be converted to the LRAT format, in which each lemma lists the clauses that justify it, and checked with a formally verified checker such as `cake_lpr` [25], as was done in [16]; we have not yet done this.

Proof of Theorem 1.1. By Proposition 5.1, a 61-block covering would make one of $F_1, \dots, F_4$ satisfiable, contradicting Theorem 6.1. Hence $C(16, 5, 3) \geq 62$. The covering of Belić [10] gives $C(16, 5, 3) \leq 65$; see also [6, Corollary 2.3.18]. $\square$

7. CONCLUDING REMARKS

The value of $C(16, 5, 3)$ is now known to be 62, 63, 64 or 65. Part of the method carries over to more blocks. For any $(16, 5, 3)$ covering with b blocks, the identities of Section 3 give $\sum_{\{x,y\}}(\lambda_{xy} - 5) = 10b - 600$ and $\sum_x(r_x - 19) = 5b - 304$. At $b = 62$ at most six points have degree greater than 19, so at least ten points have degree 19. The derived design at each of them is a $(15, 4, 2)$ covering with 19 blocks, so by Theorem 4.2 it is isomorphic to one of $\mathcal{R}_1, \dots, \mathcal{R}_4$. The difference is that the graph of pairs with $\lambda_{xy} > 5$ is no longer forced: at 62 blocks the counting allows many degree sequences and excess graphs, so an analogous case split would be considerably larger.

8. DATA AVAILABILITY

The repository [9], licensed CC BY 4.0, contains the structural argument; Computation 1 of Section 4 (`experiments/2026-10-03/link-hub-independent`) and Computation 2 (`experiments/2026-10-06/reclassify-15-4-2`) with their evidence; the isomorphisms from the designs of [2] onto $\mathcal{R}_1, \dots, \mathcal{R}_4$ (`experiments/2026-10-03/literature-and-novelty`); the encoder (`experiments/2026-10-05/lb61-certified`) and its audit (`experiments/2026-10-06/lb61-encoder-audit`); the CaDiCaL and CP-SAT cross-checks; and the solver records and drat-trim logs. The SAT computations of Section 6 are those of commit `fd7adc2`, and the classification and encoder checks added for this paper are in commit `d490808`. The four CNF formulas and DRAT proofs (about 32 GB uncompressed), with SHA256 checksums, the search trees of Computation 1 and instructions for checking the proofs, are archived on Zenodo [8]. The ancillary files of this arXiv submission contain the representatives $\mathcal{R}_1, \dots, \mathcal{R}_4$ (as the 19 blocks through the point 1), the script `gen_cnf.py` that writes the four formulas without solving them, the CP-SAT model and the SHA256 checksums of all formulas and proofs.

USE OF AI ASSISTANCE

AI coding assistants (OpenAI Codex and Anthropic Claude Code), working under the author’s direction, wrote the programs in the accompanying repository, including both classification programs of Section 4, the encoder and the CP-SAT model. They also helped develop the counting arguments

of Section 3, search the literature, draft this manuscript and check its references and numbers. In this paper, computations described as independent or separately written are separate programs that share no code; they were not written by different people. The author has checked the arguments and the references and takes full responsibility for the content.

REFERENCES

1. Giovanni Acerbi, *Covering Repository*, <https://coveringrepository.com>, 2026, Accessed 2026-10-06.
2. J. L. Allston, R. W. Buskens, and R. G. Stanton, *An examination of the non-isomorphic solutions to a problem in covering designs on fifteen points*, J. Combin. Math. Combin. Comput. **4** (1988), 189–206.
3. David Applegate, E. M. Rains, and N. J. A. Sloane, *On asymmetric coverings and covering numbers*, J. Combin. Des. **11** (2003), no. 3, 218–228.
4. Armin Biere, *Splatz, Lingeling, Plingeling, Treengeling, YalSAT entering the SAT Competition 2016*, Proc. of SAT Competition 2016 – Solver and Benchmark Descriptions, Department of Computer Science Series of Publications B, vol. B-2016-1, University of Helsinki, 2016, pp. 44–45.
5. Armin Biere, Tobias Faller, Katalin Fazekas, Mathias Fleury, Nils Froleyks, and Florian Pollitt, *CaDiCaL 2.0*, Computer Aided Verification – CAV 2024, Part I, Lecture Notes in Comput. Sci., vol. 14681, Springer, 2024, pp. 133–152.
6. Iliya Bluskov, *New designs and coverings*, Ph.D. thesis, Simon Fraser University, 1997.
7. Joshua Brakensiek, Marijn Heule, John Mackey, and David Narváez, *The resolution of Keller’s conjecture*, J. Automat. Reason. **66** (2022), no. 3, 277–300.
8. Christopher B. Celaya, *Certified DRAT proofs that no 61-block $(16, 5, 3)$ covering exists, so $C(16, 5, 3) \geq 62$* , Zenodo dataset, <https://doi.org/10.5281/zenodo.23191655>, 2026.
9. ———, *covering64: research on the covering number $C(16, 5, 3)$* , <https://github.com/celaya-solutions/covering64>, 2026.
10. Daniel M. Gordon, *La Jolla Coverings Repository*, Zenodo dataset, version 1.2, <https://doi.org/10.5281/zenodo.19735294>, 2026.
11. Daniel M. Gordon, Greg Kuperberg, and Oren Patashnik, *New constructions for covering designs*, J. Combin. Des. **3** (1995), no. 4, 269–284.
12. Daniel M. Gordon and Douglas R. Stinson, *Coverings*, Handbook of Combinatorial Designs (Charles J. Colbourn and Jeffrey H. Dinitz, eds.), Discrete Mathematics and Its Applications, Chapman & Hall/CRC, Boca Raton, FL, second ed., 2007, pp. 365–373.
13. Marijn J. H. Heule, Oliver Kullmann, and Victor W. Marek, *Solving and verifying the Boolean Pythagorean triples problem via cube-and-conquer*, Theory and Applications of Satisfiability Testing – SAT 2016, Lecture Notes in Comput. Sci., vol. 9710, Springer, 2016, pp. 228–245.
14. Daniel Horsley and Rakhi Singh, *New lower bounds for t -coverings*, J. Combin. Des. **26** (2018), no. 8, 369–386.
15. Alexey Ignatiev, Antonio Morgado, and Joao Marques-Silva, *PySAT: A Python toolkit for prototyping with SAT oracles*, Theory and Applications of Satisfiability Testing – SAT 2018, Lecture Notes in Comput. Sci., vol. 10929, Springer, 2018, pp. 428–437.
16. Charlie Krug, *The covering number $C(12, 6, 4)$ is 41*, arXiv:2607.23766, 2026.
17. François Margot, *Small covering designs by branch-and-cut*, Math. Program. **94** (2003), no. 2–3, 207–220.
18. W. H. Mills, *On the covering of pairs by quadruples I*, J. Combin. Theory Ser. A **13** (1972), no. 1, 55–78.

19. ———, *On the covering of pairs by quadruples. II*, J. Combin. Theory Ser. A **15** (1973), no. 2, 138–166.
20. W. H. Mills and R. C. Mullin, *Coverings and packings*, Contemporary Design Theory: A Collection of Surveys (Jeffrey H. Dinitz and Douglas R. Stinson, eds.), Wiley, New York, 1992, pp. 371–399.
21. Laurent Perron and Frédéric Didier, *CP-SAT, Google OR-Tools, version 9.15*, https://developers.google.com/optimization/cp/cp_solver/, 2026.
22. J. Schönheim, *On coverings*, Pacific J. Math. **14** (1964), no. 4, 1405–1411.
23. Alexander Sidorenko, *What we know and what we do not know about Turán numbers*, Graphs Combin. **11** (1995), no. 2, 179–199.
24. Carsten Sinz, *Towards an optimal CNF encoding of Boolean cardinality constraints*, Principles and Practice of Constraint Programming – CP 2005, Lecture Notes in Comput. Sci., vol. 3709, Springer, 2005, pp. 827–831.
25. Yong Kiam Tan, Marijn J. H. Heule, and Magnus O. Myreen, *cake_lpr: Verified propagation redundancy checking in CakeML*, Tools and Algorithms for the Construction and Analysis of Systems – TACAS 2021, Lecture Notes in Comput. Sci., vol. 12652, Springer, 2021, pp. 223–241.
26. Nathan Wetzler, Marijn J. H. Heule, and Warren A. Hunt, Jr., *DRAT-trim: Efficient checking and trimming using expressive clausal proofs*, Theory and Applications of Satisfiability Testing – SAT 2014, Lecture Notes in Comput. Sci., vol. 8561, Springer, 2014, pp. 422–429.

CELAYA SOLUTIONS RESEARCH, EL PASO, TEXAS, USA

Email address: hello@celayasolutions.com