The Resolution of Keller's Conjecture

Joshua Brakensiek[1], Marijn Heule[2], John Mackey[2], and David Narváez[3]

[1] Stanford University, California
[2] Carnegie Mellon University, Pennsylvania
[3] Rochester Institute of Technology, New York

Abstract. We consider three graphs, $G_{7,3}$, $G_{7,4}$, and $G_{7,6}$, related to Keller's conjecture in dimension 7. The conjecture is false for this dimension if and only if at least one of the graphs contains a clique of size $2^7 = 128$. We present an automated method to solve this conjecture by encoding the existence of such a clique as a propositional formula. We apply satisfiability solving combined with symmetry-breaking techniques to determine that no such clique exists. This result implies that every unit cube tiling of $\mathbb{R}^7$ contains a facesharing pair of cubes. Since a faceshare-free unit cube tiling of $\mathbb{R}^8$ exists (which we also verify), this completely resolves Keller's conjecture.

1 Introduction

In 1930, Keller conjectured that any tiling of n-dimensional space by translates of the unit cube must contain a pair of cubes that share a complete $(n-1)$-dimensional face [13]. Figure 1 illustrates this for the plane and the 3-dimensional space. The conjecture generalized a 1907 conjecture of Minkowski [24] in which the centers of the cubes were assumed to form a lattice. Keller's conjecture was proven to be true for $n \leq 6$ by Perron in 1940 [25, 26], and in 1942 Hajós [6] showed Minkowski's conjecture to be true in all dimensions.

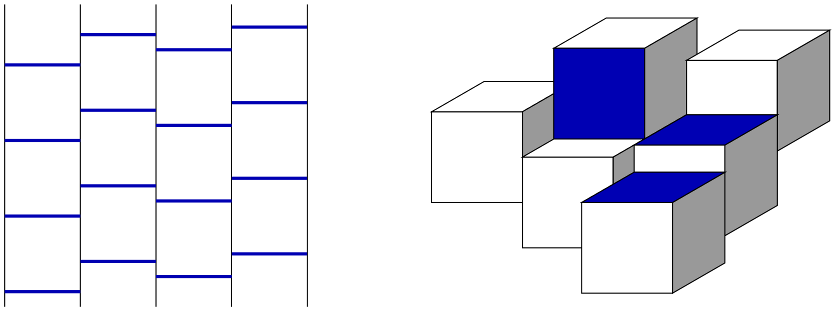

Fig. 1. Left, a tiling of the plane (2-dimensional space) with unit cubes (squares). The bold blue edges are fully face-sharing edges. Right, a partial tiling of the 3-dimensional space with unit cubes. The only way to tile the entire space would result in a fully face-sharing square at the position of the blue squares.

In 1986 Szabó [28] reduced Keller's conjecture to the study of periodic tilings. Using this reduction Corrádi and Szabó [3] introduced the Keller graphs: the graph $G_{n,s}$ has vertices $\{0, 1, \ldots, 2s-1\}^n$ such that a pair are adjacent if and only if they differ by exactly s in at least one coordinate and they differ in at least two coordinates. The size of cliques in $G_{n,s}$ is at most 2^n [5] and the size of the largest clique in $G_{n,s}$ is at most the size of the largest clique in $G_{n,s+1}$.

A clique in $G_{n,s}$ of size 2^n demonstrates that Keller's conjecture is false for dimensions greater than or equal to n. Lagarias and Shor [19] showed that Keller's conjecture is false for $n \geq 10$ in 1992 by exhibiting clique of size 2^{10} in $G_{10,2}$. In 2002, Mackey [22] found a clique of size 2^8 in $G_{8,2}$ to show that Keller's conjecture is false for $n \geq 8$. In 2011, Debroni, Eblen, Langston, Myrvold, Shor, and Weerapurage [5] showed that the largest clique in $G_{7,2}$ has size 124.

In 2015, Kisielewicz and Łysakowska [14,16] made substantial progress on reducing the conjecture in dimension 7. More recently, in 2017, Kisielewicz [15] reduced the conjecture in dimension 7 as follows: Keller's conjecture is true in dimension 7 if and only if there does not exist a clique in $G_{7,3}$ of size 2^7 [21].

The main result of this paper is the following theorem.

Theorem 1. *Neither $G_{7,3}$ nor $G_{7,4}$ nor $G_{7,6}$ contains a clique of size $2^7 = 128$.*

Although proving this property for $G_{7,3}$ suffices to prove Keller's conjecture true in dimension 7, we also show this for $G_{7,4}$ and $G_{7,6}$ to demonstrate that our methods need only depend on prior work of Kisielewicz and Łysakowska [14,16]. In particular, the argument for $G_{7,6}$ [14] predates and is much simpler than the one for $G_{7,4}$ [16] (although the publication dates indicate otherwise). It is not explicitly stated in either that it suffices to prove that $G_{7,4}$ or $G_{7,6}$ lacks a clique of size 128 to prove Keller's conjecture. We show this in the Appendix of the extended version, available at https://arxiv.org/abs/1910.03740.

We present an approach based on satisfiablity (SAT) solving to show the absence of a clique of size 128. SAT solving has become a powerful tool in computer-aided mathematics in recent years. For example, it was used to prove the Erdős discrepancy conjecture with discrepancy 2 [17], the Pythagorean triples problem [10], and Schur number five [7]. Modern SAT solvers can also emit proofs of unsatisfiability. There exist formally verified checkers for such proofs as developed in the ACL2, Coq, and Isabelle theorem-proving systems [4,20].

The outline of this paper is as follows. After describing some background concepts in Section 2, we present a compact encoding whether $G_{n,s}$ contains a clique of size 2^n as a propositional formula in Section 3. Without symmetry breaking, these formulas with $n > 5$ are challenging for state-of-the-art tools. However, the Keller graphs contain many symmetries. We perform some initial symmetry breaking that is hard to express on the propositional level in Section 4. This allows us to partially fix three vertices. On top of that we add symmetry-breaking clauses in Section 5. The soundness of their addition has been mechanically verified. We prove in Section 6 the absence of a clique of size 128 in $G_{7,3}$, $G_{7,4}$ and $G_{7,6}$. We optimize the proofs of unsatisfiability obtained by the SAT solver and certify them using a formally verified checker. Finally we draw some conclusions in Section 7 and present directions for future research.

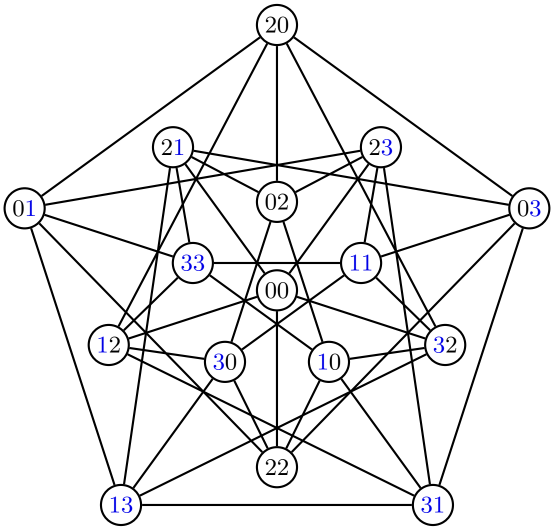

Fig. 2. Illustration of $G_{2,2}$. The coordinates of the vertices are compactly represented by a sequence of the digits.

2 Preliminaries

We present the most important background concepts related to this paper and introduce some properties of $G_{n,s}$. First, for positive integers k, we define two sets: $[k] := \{1, 2, \ldots, k\}$ and $\langle k \rangle := \{0, 1, \ldots, k-1\}$.

Keller Graphs. The Keller graph $G_{n,s}$ consists of the vertices $\langle 2s \rangle^n$. Two vertices are adjacent if and only if they differ by exactly s in at least one coordinate and they differ in at least two coordinates. Figure 2 shows a visualization of $G_{2,2}$.

As noted in [5], $\{sw + \langle s \rangle^n : w \in \{0,1\}^n\}$ is a partition of the vertices of $G_{n,s}$ into 2^n independent sets. Consequently, any clique in $G_{n,s}$ has at most 2^n vertices. For example, $V(G_{2,2})$ is partitioned as follows:

$$
\begin{aligned}
&\{\{2(0,0) + \{0,1\}^2,\ 2(0,1) + \{0,1\}^2,\ 2(1,0) + \{0,1\}^2,\ 2(1,1) + \{0,1\}^2\} = \\
&\{\{(0,0),(0,1),(1,0),(1,1)\}, \{(0,2),(0,3),(1,2),(1,3)\}, \\
&\ \{(2,0),(2,1),(3,0),(3,1)\}, \{(2,2),(2,3),(3,2),(3,3)\}\}.
\end{aligned}
$$

We use the above observation for encoding whether $G_{n,s}$ has a clique of size 2^n. Instead of searching for such a clique on the graph representation of $G_{n,s}$, which consists of $(2s)^n$ vertices, we search for 2^n vertices, one from each $sw + \langle s \rangle^n$, such that every pair is adjacent.

For every $i \in \langle 2^n \rangle$, we let $w(i) = (w_1, w_2, \ldots, w_n) \in \{0,1\}^n$ be defined by $i = \sum_{k=1}^{n} 2^{k-1} \cdot w_k$. Given a clique of size 2^n, we let c_i be its unique element in $sw(i) + \langle s \rangle^n$ and we let $c_{i,j}$ be the jth coordinate of c_i.

Useful Automorphisms of Keller Graphs. Let S_n be the set of permutations of $[n]$ and let H_s be the set of permutations of $\langle 2s \rangle$ generated by the swaps $(i\ i+s)$ composed with any permutation of $\langle s \rangle$ which is identically applied to $s + \langle s \rangle$. The maps

$$(x_1, x_2, \ldots, x_n) \mapsto (\tau_1(x_{\sigma(1)}), \tau_2(x_{\sigma(2)}), \ldots, \tau_n(x_{\sigma(n)})),$$

where $\sigma \in S_n$ and $\tau_1, \tau_2, \ldots, \tau_n \in H_s$ are automorphisms of $G_{n,s}$. Note that applying an automorphism to every vertex of a clique yields another clique of the same size.

Propositional Formulas. We consider formulas in *conjunctive normal form* (CNF), which are defined as follows. A *literal* is either a variable x (a *positive literal*) or the negation $\overline{x}$ of a variable x (a *negative literal*). The *complement* $\overline{l}$ of a literal l is defined as $\overline{l} = \overline{x}$ if $l = x$ and $\overline{l} = x$ if $l = \overline{x}$. For a literal l, $var(l)$ denotes the variable of l. A *clause* is a disjunction of literals and a *formula* is a conjunction of clauses.

An *assignment* is a function from a set of variables to the truth values 1 (*true*) and 0 (*false*). A literal l is *satisfied* by an assignment α if l is positive and $\alpha(var(l)) = 1$ or if it is negative and $\alpha(var(l)) = 0$. A literal is *falsified* by an assignment if its complement is satisfied by the assignment. A clause is satisfied by an assignment α if it contains a literal that is satisfied by α. A formula is satisfied by an assignment α if all its clauses are satisfied by α. A formula is *satisfiable* if there exists an assignment that satisfies it and *unsatisfiable* otherwise.

Clausal Proofs. Our proof that Keller's conjecture is true for dimension 7 is predominantly a clausal proof, including a large part of the symmetry breaking. Informally, a clausal proof system allows us to show the unsatisfiability of a CNF formula by continuously deriving more and more clauses until we obtain the empty clause. Thereby, the addition of a derived clause to the formula and all previously derived clauses must preserve satisfiability. As the empty clause is trivially unsatisfiable, a clausal proof shows the unsatisfiability of the original formula. Moreover, it must be checkable in polynomial time that each derivation step does preserve satisfiability. This requirement ensures that the correctness of proofs can be efficiently verified. In practice, this is achieved by allowing only the derivation of specific clauses that fulfill some efficiently checkable criterion.

Formally, clausal proof systems are based on the notion of *clause redundancy*. A clause C is *redundant* with respect to a formula F if adding C to F preserves satisfiability. Given a formula $F = C_1 \land \cdots \land C_m$, a *clausal proof* of F is a sequence $(C_{m+1}, \omega_{m+1}), \ldots, (C_n, \omega_n)$ of pairs where each C_i is a clause, each ω_i (called the *witness*) is a string, and C_n is the empty clause [9]. Such a sequence gives rise to formulas $F_m, F_{m+1}, \ldots, F_n$, where $F_i = C_1 \land \cdots \land C_i$. A clausal proof is *correct* if every clause C_i $(i > m)$ is redundant with respect to F_{i-1}, and if this redundancy can be checked in polynomial time (with respect to the size of the proof) using the witness ω_i.

An example for a clausal proof system is the resolution proof system, which only allows the derivation of resolvents (with no or empty witnesses). However, the resolution proof system does not allow to compactly express symmetry breaking. Instead we will construct a proof in the resolution asymmetric tautology (RAT) proof system. This proof system is also used to validate the results of the SAT competitions [11]. For the details of RAT, we refer to the original paper [9]. Here, we just note that (1) for RAT clauses, it can be checked efficiently that their addition preserves satisfiability, and (2) every resolvent is a RAT clause but not vice versa.

3 Clique Existence Encoding

Recall that $G_{n,s}$ has a clique of size 2^n if and only if there exist vertices $c_i \in sw(i) + \langle s \rangle^n$ for all $i \in \langle 2^n \rangle$ such that for all $i \neq i'$ there exists at least two $j \in [n]$ such that $c_{i,j} \neq c_{i',j}$ and there exists at least one $j \in [n]$ such that $c_{i,j} = c_{i',j} \pm s$.

Our CNF will encode the coordinates of the c_i. For each $i \in \langle 2^n \rangle$, $j \in [n]$, $k \in \langle s \rangle$, we define Boolean variables $x_{i,j,k}$ which are true if and only if $c_{i,j} = sw(i)_j + k$. We therefore need to encode that exactly one of $x_{i,j,0}$, $x_{i,j,1}$, …, $x_{i,j,s-1}$ is true. We use the following clauses

$$\forall i \in \langle 2^n \rangle, \forall j \in [n],\; (x_{i,j,0} \vee x_{i,j,1} \vee \cdots \vee x_{i,j,s-1}) \wedge \bigwedge_{k<k' \in \langle s \rangle} (\overline{x}_{i,j,k} \vee \overline{x}_{i,j,k'}). \tag{1}$$

Next we enforce that every pair of vertices c_i and $c_{i'}$ in the clique differ in at least two coordinates. For most pairs of vertices, no clauses are required because $w(i)$ and $w(i')$ differ in at least two positions. Hence, a constraint is only required for two vertices if $w(i)$ and $w(i')$ differ in exactly one coordinate.

Let $\oplus$ be the binary XOR operator and e_j be the indicator vector of the jth coordinate. If $w(i) \oplus w(i') = e_j$, then in order to ensure that c_i and $c_{i'}$ differ in at least two coordinates we need to make sure that there is some coordinate $j' \neq j$ for which $c_{i,j'} \neq c_{i',j'}$

$$\forall i \neq i' \in \langle 2^n \rangle \text{ s.t. } w(i) \oplus w(i') = e_j, \bigvee_{j' \in [n] \setminus \{j\}, k \in \langle s \rangle} (x_{i,j',k} \neq x_{i',j',k}). \tag{2}$$

We use the Plaisted-Greenbaum [27] encoding to convert the above constraint into CNF. We refer to the auxiliary variables introduced by the encoding as $y_{i,i',j',k}$, which if true imply $x_{i,j',k} \neq x_{i',j',k}$, or written as an implication

$$y_{i,i',j',k} \rightarrow (x_{i,j',k} \neq x_{i',j',k})$$

The following two clauses express this implication

$$(\overline{y}_{i,i',j',k} \vee x_{i,j',k} \vee x_{i',j',k}) \wedge (\overline{y}_{i,i',j',k} \vee \overline{x}_{i,j',k} \vee \overline{x}_{i',j',k}) \tag{3}$$

Notice that the implication is only in one direction as Plaisted-Greenbaum takes the polarity of constraints into account. The clauses that represent the other direction, i.e., $(y_{i,i',j',k} \vee x_{i,j',k} \vee \overline{x}_{i',j',k})$ and $(y_{i,i',j',k} \vee \overline{x}_{i,j',k} \vee x_{i',j',k})$ are redundant (and more specifically, they are blocked [18]).

Using the auxiliary variables, we can express the constraint (2) using clauses of length $s \cdot (n-1)$

$$\forall i \neq i' \in \langle 2^n \rangle \text{ s.t. } w(i) \oplus w(i') = e_j, \bigvee_{j' \in [n] \setminus \{j\}, k \in \langle s \rangle} y_{i,i',j',k}. \tag{4}$$

The last part of the encoding consists of clauses to ensure that each pair of vertices in the clique have at least one coordinate in which they differ by exactly s. Observe that $c_{i,j} = c_{i',j} \pm s$ implies that $c_{i,j} \neq c_{i',j}$ and $x_{i,j,k} = x_{i',j,k}$ for all $k \in \langle s \rangle$. We use auxiliary variables $z_{i,i',j}$, whose truth implies $c_{i,j} = c_{i',j} \pm s$, or written as an implication

$$\begin{aligned} &\forall i \neq i' \in \langle 2^n \rangle, \forall j \in [n] \text{ s.t. } c_{i,j} \neq c_{i',j}, \\ &\quad z_{i,i',j} \rightarrow \big((x_{i,j,0} = x_{i',j,0}) \wedge \cdots \wedge (x_{i,j,s-1} = x_{i',j,s-1})\big). \end{aligned}$$

Notice that the implication is again in one direction only. Below we enforce that some $z_{i,i',j}$ variables must be true, but there are no constraints that enforce $z_{i,i',j}$ variables to be false.

This can be written as a CNF using the following clauses:

$$\bigwedge_{k \in \langle s \rangle} \big((\overline{z}_{i,i',j} \vee x_{i,j,k} \vee \overline{x}_{i',j,k}) \wedge (\overline{z}_{i,i',j} \vee \overline{x}_{i,j,k} \vee x_{i',j,k})\big) \tag{5}$$

Finally, to make sure that $c_{i,j} = c_{i',j} \pm s$ for some $j \in [n]$, we specify

$$\forall i \neq i' \in \langle 2^n \rangle, \qquad \bigvee_{j : c_{i,j} \neq c_{i',j}} z_{i,i',j}. \tag{6}$$

The variables and clauses, including precise formulas for their counts, are summarized in Table 1. The sizes of the CNF encodings (before the addition of symmetry breaking clauses) of $G_{7,3}$, $G_{7,4}$, and $G_{7,6}$ are listed in Table 2. Notice that for fixed n, the dependence on s is quadratic, which is better than the s^{2n} dependence one would get in the naive encoding of $G_{n,s}$ as a graph. This compact encoding, when combined with symmetry breaking, is a core reason that we were able to prove Theorem 1.

The instances with $n = 7$ are too hard for state-of-the-art SAT solvers if no symmetry breaking is applied. We experimented with general-purpose symmetry-breaking techniques, similar to the symmetry-breaking predicates produced by shatter [1]. This allows solving the formula for $G_{7,3}$, but the computation takes a few CPU years. The formulas for $G_{7,4}$ and $G_{7,6}$ with these symmetry-breaking predicates are significantly harder.

Instead we employ problem-specific symmetry breaking by making use of the observations in Sections 4 and 5. This allows solving the clique of size 2^n existence problem for all three graphs in reasonable time.

Table 1. Summary of variable and clause counts in the CNF encoding.

Clauses	New Variable Count	Clause Count
(1)	$2^n \cdot n \cdot s$	$2^n \cdot n \cdot (1 + \binom{s}{2})$
(3)	$2^{n-1} \cdot n \cdot s \cdot (n-1)$	$2^n \cdot n \cdot s \cdot (n-1)$
(4)		$2^{n-1} \cdot n$
(5)	$2^{2n-2} \cdot n$	$2^{2n-1} \cdot n \cdot s$
(6)		$\binom{2^n}{2}$
Total	$2^{n-1} \cdot n \cdot (s(n+1) + 2^{n-1})$	$2^n \cdot n \cdot \left(\frac{3}{2} + \binom{s}{2} + n \cdot s - s\right) + 2^{2n-1}ns + \binom{2^n}{2}$

4 Initial Symmetry Breaking

Our goal is to prove that there exists no clique of size 128 in $G_{7,s}$ for $s \in \{3, 4, 6\}$. In this section, and the subsequent, we assume that such a clique exists and adapt some of the arguments of Perron [25,26] to show that it may be assumed to have a canonical form. We will use $\star_i$ to denote an element in $\langle i \rangle$.

Lemma 1. *If there is a clique of size* 128 *in* $G_{7,s}$*, then there is a clique of size* 128 *in* $G_{7,s}$ *containing the vertices* $(0, 0, 0, 0, 0, 0, 0)$ *and* $(s, 1, 0, 0, 0, 0, 0)$.

Proof. Let K be a clique of size 128 in $G_{7,s}$. Consider the following sets of vertices in $G_{6,s}$:

$$K_{<s} := \{(v_2, \ldots, v_7) \mid \exists v_1 \in \langle s \rangle \text{ s.t. } (v_1, \ldots, v_7) \in K\}$$

and

$$K_{\geq s} := \{(v_2, \ldots, v_7) \mid \exists v_1 \in s + \langle s \rangle \text{ s.t. } (v_1, \ldots, v_7) \in K\}.$$

Every pair of vertices in $K_{<s}$ differs by exactly s in at least one coordinate, because the corresponding pair of vertices in K can't differ by exactly s in the first coordinate. Similarly, every pair of vertices in $K_{\geq s}$ differs by exactly s in at least one coordinate. Although $K_{<s}$ and $K_{\geq s}$ are not necessarily cliques in $G_{6,s}$, they satisfy the first condition of the adjacency requirement. The partition of section 2 can thus be applied to deduce that $|K_{<s}| \leq 64$ and $|K_{\geq s}| \leq 64$. Since $|K_{<s}| + |K_{\geq s}| = 128$, we conclude that $|K_{<s}| = 64$ and $|K_{\geq s}| = 64$.

By the truth of Keller's conjecture in dimension 6, $K_{<s}$ is not a clique in $G_{6,s}$. Thus, some pair of vertices in $K_{<s}$ are identical in five of the six coordinates.

Table 2. Summary of variable and clause counts of the CNF encoding for $G_{7,3}$, $G_{7,4}$, and $G_{7,6}$. These counts do not include the clauses introduced by the symmetry breaking.

Keller Graph	Variable Count	Clause Count
$G_{7,3}$	39 424	200 320
$G_{7,4}$	43 008	265 728
$G_{7,6}$	50 176	399 232

After application of an automorphism, we may without loss of generality assume that this pair is $(s, 0, 0, 0, 0, 0)$ and $(0, 0, 0, 0, 0, 0)$. Since the pair comes from $K_{<s}$, there exist $v_1 \neq v_1' \in \langle s \rangle$ such that $(v_1, s, 0, 0, 0, 0, 0)$ and $(v_1', 0, 0, 0, 0, 0, 0)$ are in the clique.

After application of an automorphism that moves v_1 to 1 and v_1' to 0, we deduce that without loss of generality $(1, s, 0, 0, 0, 0, 0)$ and $(0, 0, 0, 0, 0, 0, 0)$ are in the clique. Application of the automorphism that interchanges the first two coordinates yields a clique of size 128 containing the vertices $c_0 = (0, 0, 0, 0, 0, 0, 0)$ and $c_1 = (s, 1, 0, 0, 0, 0, 0)$. □

Theorem 2. *If there is a clique of size* 128 *in* $G_{7,s}$*, then there is a clique of size* 128 *in* $G_{7,s}$ *containing the vertices* $(0, 0, 0, 0, 0, 0, 0)$*,* $(s, 1, 0, 0, 0, 0, 0)$*, and* $(s, s + 1, \star_2, \star_2, 1, 1, 1)$*.*

Proof. Using the preceding lemma, we can choose from among all cliques of size 128 that contain $c_0 = (0, 0, 0, 0, 0, 0, 0)$ and $c_1 = (s, 1, 0, 0, 0, 0, 0)$, one in which c_3 has the fewest number of coordinates equal to 0. Let λ be this least number.

Observe that the first two coordinates of c_3 must be $(s, s + 1)$ in order for it to be adjacent with both c_0 and c_1. Thus, we have

$$\begin{aligned} c_0 &= (0\,,\quad 0\quad ,\,0\,,\,0\,,\,0\,,\,0\,,\,0\,) \\ c_1 &= (s\,,\quad 1\quad ,\,0\,,\,0\,,\,0\,,\,0\,,\,0\,) \\ c_3 &= (s\,,\,s+1\,,\star_s\,,\star_s\,,\star_s\,,\star_s\,,\star_s) \end{aligned}$$

In the above, we can apply automorphisms that fix 0 in the last five coordinates to replace $\star_s$ by $\star_2$. We can apply an automorphism that permutes the last five coordinates to assume that the 0's and 1's in c_3 are sorted in increasing order. Notice that not all of the $\star_2$ coordinates in c_3 can be 0, because c_1 and c_3 are adjacent and must therefore differ in at least two coordinates. Hence at least the last coordinate of c_3 is 1.

Case 1) $\lambda = 4$. In this case $c_3 = (s, s + 1, 0, 0, 0, 0, 1)$. In order for c_{67} to be adjacent with c_0, c_1, and c_3, it must start with $(s, s + 1)$ and end with $s + 1$:

$$\begin{aligned} c_0 &= (0\,,\quad 0\quad ,\,0\,,\,0\,,\,0\,,\,0\,,\quad 0\quad) \\ c_1 &= (s\,,\quad 1\quad ,\,0\,,\,0\,,\,0\,,\,0\,,\quad 0\quad) \\ c_3 &= (s\,,\,s+1\,,\,0\,,\,0\,,\,0\,,\,0\,,\quad 1\quad) \\ c_{67} &= (s\,,\,s+1\,,\star_s\,,\star_s\,,\star_s\,,\star_s\,,\,s+1) \end{aligned}$$

Not all $\star_s$ elements in c_{67} can be 0, because c_3 and c_{67} differ in at least two coordinates. However, if one of the $\star_s$ elements in c_{67} is nonzero, then we can swap 1 and $s + 1$ in the last coordinate to obtain a clique in which c_3 has three or fewer coordinates equal to 0, contradicting $\lambda = 4$. Thus, $\lambda \leq 3$.

Case 2) $\lambda = 3$, in which case $c_3 = (s, s + 1, 0, 0, 0, 1, 1)$:

$$\begin{aligned} c_0 &= (0\,,\quad 0\quad ,\,0\,,\,0\,,\,0\,,\quad 0\quad ,\quad 0\quad) \\ c_1 &= (s\,,\quad 1\quad ,\,0\,,\,0\,,\,0\,,\quad 0\quad ,\quad 0\quad) \\ c_3 &= (s\,,\,s+1\,,\,0\,,\,0\,,\,0\,,\quad 1\quad ,\quad 1\quad) \\ c_{35} &= (s\,,\,s+1\,,\star_s\,,\star_s\,,\star_s\,,\,s+1\,,\quad \star_s\quad) \\ c_{67} &= (s\,,\,s+1\,,\star_s\,,\star_s\,,\star_s\,,\quad \star_s\quad ,\,s+1) \end{aligned}$$

Since c_{67} is adjacent with c_0, c_1, and c_3, it must start with $(s, s+1)$ and end with $s+1$. Similarly, since c_{35} is adjacent with c_0, c_1, and c_3, it must start with $(s, s+1)$ and have $s+1$ as its penultimate coordinate. Since c_{35} and c_{67} are adjacent, either the last coordinate of c_{35} must be 1, or the penultimate coordinate of c_{67} must be 1. Without loss of generality we can assume that the penultimate coordinate of c_{67} is 1 as we can permute the last two coordinates which would swap c_{35} and c_{67} without involving the other cubes. The remaining three $\star_s$ elements in c_{67} cannot all be 0, since c_3 and c_{67} differ in at least two coordinates. However, if one of the $\star_s$ elements is non-zero, then we can swap 1 and $s+1$ in the last coordinate to obtain a clique in which c_3 has two or fewer coordinates equal to 0, contradicting $\lambda = 3$. Thus, we have $\lambda \leq 2$ and $c_3 = (s, s+1, \star_2, \star_2, 1, 1, 1)$, as desired. □

Notice that most of the symmetry breaking discussed in this section is challenging, if not impossible, to break on the propositional level: The proof of Lemma 1 uses the argument that Keller's conjecture holds for dimension 6, while the proof of Theorem 2 uses the interchangeability of 1 and $s+1$, which is not a symmetry on the propositional level. We will break these symmetries by adding some unit clauses to the encoding. All additional symmetry breaking will be presented in the next section and will be checked mechanically.

5 Clausal Symmetry Breaking

Our symmetry-breaking approach starts with enforcing the initial symmetry breaking: We assume that vertices $c_0 = (0,0,0,0,0,0,0)$, $c_1 = (s,1,0,0,0,0,0)$ and $c_3 = (s, s+1, \star_s, \star_s, 1, 1, 1)$ are in our clique K, which follows from Theorem 2. We will not use the observation that $\star_s$ occurrences in c_3 can be reduced to $\star_2$ and instead add and validate clauses that realize this reduction.

We fix the above initial vertices by adding unit clauses to the CNF encoding. This is the only part of the symmetry breaking that is not checked mechanically. Let $\Phi_{7,s}$ be the formula obtained from our encoding in Section 3 together with the unit clauses corresponding to the 19 coordinates fixed among c_0, c_1 and c_3. In this section we will identify several symmetries in $\Phi_{7,s}$ that can be further broken at the CNF level by adding symmetry breaking clauses. The formula ultimately used in Section 6 for the experiments is the result of adding these symmetry breaking clauses to $\Phi_{7,s}$. Symmetry breaking clauses are added in an incremental fashion. For each addition, a clausal proof of its validity with respect to $\Phi_{7,s}$ and the clauses added so far is generated, as well. Each of these clausal proofs has been validated using the `drat-trim` proof checker.

Our approach can be described in general terms as identifying groups of coordinates whose assignments exhibit interesting symmetries and calculating the equivalence classes of these assignments. Given a class of symmetric assignments, it holds that one of these assignments can be extended to a clique of size 128 if and only if every assignment in that class can be extended as well. It is then enough to pick a canonical representative for each class, add clauses forbidding every assignment that is not canonical, and finally determine the satisfiability of the formula under the canonical representative of every class of assignments: if no canonical assignment can be extended to a satisfying assignment for the formula, then the formula is unsatisfiable. In order to forbid assignments that are not canonical, we use an approach similar to the one described in [8].

5.1 The Last Three Coordinates of c_{19}, c_{35} and c_{67}

The reasoning in the proof of Theorem 2 leads to the following forced settings, once we assign $c_3 = (s, s+1, \star_s, \star_s, 1, 1, 1)$ and apply unit propagation:

$$\begin{aligned}(c_{19,1}, c_{19,2}, c_{19,5}) &= (s, s+1, s+1),\\ (c_{35,1}, c_{35,2}, c_{35,6}) &= (s, s+1, s+1),\\ (c_{67,1}, c_{67,2}, c_{67,7}) &= (s, s+1, s+1).\end{aligned}$$

Let's now focus on the 3×3 matrix of the coordinates below and do a case split on all of the s^6 possible assignments of coordinates labeled with $\star_s$.

	5	6	7
c_{19}	$s+1$	$\star_s$	$\star_s$
c_{35}	$\star_s$	$s+1$	$\star_s$
c_{67}	$\star_s$	$\star_s$	$s+1$

Notice, however, that since the only positions in which c_{19} and c_{35} can differ by exactly s are positions 5 and 6, and since $c_{19,5}$ and $c_{35,6}$ are already set to $s+1$, at least one of $c_{19,6}$ and $c_{35,5}$ has to be set to 1. Similarly, it is not possible for both $c_{35,7}$ and $c_{67,6}$ to not be 1 and for both $c_{67,5}$ and $c_{19,7}$ to not be 1. By the inclusion-exclusion principle, this reasoning alone discards $3(s-1)^2s^4 - 3(s-1)^4s^2 + (s-1)^6$ cases. All of these cases can be blocked by adding the binary clauses: $(x_{19,6,1} \vee x_{35,5,1}) \wedge (x_{35,7,1} \vee x_{67,6,1}) \wedge (x_{67,5,1} \vee x_{19,7,1})$. These three clauses are RAT clauses [12] with respect to the formula $\Phi_{7,s}$.

Furthermore, among the remaining $(2s-1)^3$ cases, several assignment pairs are symmetric. For example, the following two assignments are symmetric because one can be obtained from the other by swapping columns and rows:

	5	6	7
c_{19}	$s+1$	1	2
c_{35}	2	$s+1$	2
c_{67}	1	1	$s+1$

	5	6	7
c_{19}	$s+1$	1	1
c_{35}	2	$s+1$	1
c_{67}	2	2	$s+1$

As with many problems related to symmetries, we can encode each assignment as a vertex-colored graph and use canonical labeling algorithms to determine a canonical assignment representing all the symmetric assignments of each equivalence class, and which assignments are symmetric to each canonical form. Our approach is similar to the one by McKay and Piperno for isotopy of matrices [23].

This additional symmetry breaking reduces the number of cases for the last three coordinates of the vertices c_{19}, c_{37}, and c_{67} from the trivial s^6 to 25 cases for $s = 3$ and 28 cases for $s \geq 4$. Figure 3 shows the 25 canonical cases for $s = 3$.

$(0,0,1,0,1,1)$	$(0,0,1,1,1,1)$	$(0,0,1,1,1,2)$	$(0,1,1,0,0,1)$	$(0,1,1,0,1,1)$
$(0,1,1,0,2,1)$	$(0,1,1,1,0,2)$	$(0,1,1,1,1,0)$	$(0,1,1,1,1,1)$	$(0,1,1,1,1,2)$
$(0,1,1,1,2,0)$	$(0,1,1,1,2,1)$	$(0,1,1,1,2,2)$	$(0,1,1,2,1,1)$	$(0,1,1,2,2,1)$
$(0,2,1,1,1,1)$	$(0,2,1,1,1,2)$	$(0,2,1,2,1,1)$	$(1,1,1,1,1,1)$	$(1,1,1,1,1,2)$
$(1,1,1,1,2,2)$	$(1,1,1,2,2,1)$	$(1,1,2,1,2,1)$	$(1,1,2,1,2,2)$	$(1,2,2,1,1,2)$

Fig. 3. The 25 canonical cases for $s = 3$. Each vector corresponds to the values of the coordinates $(c_{19,6}, c_{19,7}, c_{35,5}, c_{35,7}, c_{67,5}, c_{67,6})$.

5.2 Coordinates Three and Four of Vertices c_3, c_{19}, c_{35} and c_{67}

The symmetry breaking in the previous subsection allows us to fix, without loss of generality, the last coordinate of c_{19} to 1. It also constrains the third and fourth coordinates of c_3 to take values in $\langle 2\rangle$ instead of $\langle s\rangle$.

We break the computation into further cases by enumerating over choices for the third and fourth coordinates of vertices c_3, c_{19}, c_{37}, and c_{67}. Up to this point, our description of the partial clique is invariant under the permutations of $\langle s-1\rangle$ in the third and fourth coordinates as well as swapping the third and fourth coordinates. With respect to these automorphisms, for $s = 3$ there are only 861 equivalence classes for how to fill in the $\star_s$ cases for these four vertices. For $s = 4$ there are 1326 such equivalence classes, and for $s = 6$ there are 1378 such equivalence classes. This gives a total of $25 \times 861 = 21\,525$ cases to check for $s = 3$, $28 \times 1326 = 37\,128$ cases to check for $s = 4$, and $28 \times 1378 = 38\,584$ cases to check for $s = 6$.

5.3 Identifying Hardest Cases

In initial experiments we observed for each $s \in \{3, 4, 6\}$ that out of the many thousands of subformulas (cases), one subformula was significantly harder to solve compared to the other subformulas. Figure 5 shows the coordinates of the key vertices of this subformula for $s \in \{3, 4, 6\}$. Notice that the third and fourth coordinates are all 0 for all the key vertices. We therefore applied additional symmetry breaking in case all of these coordinates are 0. Under this case, the third and the fourth coordinates of vertex c_2 can be restricted to $(0, 0)$, $(0, 1)$,

$$\begin{array}{rl}
c_0 &= (\mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}) \\
c_1 &= (\boldsymbol{s}, \mathbf{1}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}) \\
c_3 &= (\boldsymbol{s}, \boldsymbol{s+1}, \star_2, \star_2, \mathbf{1}, \mathbf{1}, \mathbf{1}) \\
c_{19} &= (s, s+1, \star_3, \star_3, s+1, \star_3, 1) \\
c_{35} &= (s, s+1, \star_4, \star_4, \star_3, s+1, \star_3) \\
c_{67} &= (s, s+1, \star_5, \star_5, \star_4, \star_4, s+1)
\end{array}$$

Fig. 4. Part of the symmetry breaking on the key vertices. The bold coordinates show the (unverified) initial symmetry breaking. The bold s and $s+1$ coordinates in c_1 and c_3 are also implied by unit propagation. The additional symmetry breaking is validated by checking a DRAT proof expressing the symmetry breaking clauses.

c_0	$(0,0,0,0,0,0,0)$	$(0,0,0,0,0,0,0)$	$(0,0,0,0,0,0,0)$
c_1	$(3,1,0,0,0,0,0)$	$(4,1,0,0,0,0,0)$	$(6,1,0,0,0,0,0)$
c_3	$(3,4,0,0,1,1,1)$	$(4,5,0,0,1,1,1)$	$(6,7,0,0,1,1,1)$
c_{19}	$(3,4,0,0,4,0,1)$	$(4,5,0,0,5,0,1)$	$(6,7,0,0,7,0,1)$
c_{35}	$(3,4,0,0,1,4,0)$	$(4,5,0,0,1,5,0)$	$(6,7,0,0,1,7,0)$
c_{67}	$(3,4,0,0,0,1,4)$	$(4,5,0,0,0,1,5)$	$(6,7,0,0,0,1,7)$

Fig. 5. The hardest instance for $s = 3$ (left), $s = 4$ (middle), and $s = 6$ (right).

and $(1,1)$, and the last three coordinates of c_2 can only take values in $\langle 3 \rangle$. Furthermore, any assignment (a, b, c) to the last three coordinates of c_2 is symmetric to the same assignment "shifted right", i.e. (c, a, b), by swapping columns and rows appropriately. These symmetries define equivalence classes of assignments that can also be broken at the CNF level. Under the case shown in Figure 5, there are only 33 non-isomorphic assignments remaining for vertex c_2 for $s \geq 3$. We replace the hard case for each $s \in \{3, 4, 6\}$ by the corresponding 33 cases, thereby increasing the total number of cases mentioned above by 32.

5.4 SAT Solving

Each of the cases was solved using a SAT solver, which produced a proof of unsatisfiability that was validated using a formally verified checker (details are described in the following section). To ensure that the combined cases cover the entire search space, we constructed for each $s \in \{3, 4, 6\}$ a tautological formula in disjunctive normal form (DNF). The building blocks of a DNF are conjuctions of literals known as cube. We will use α as a symbol for cubes as they are can also be considered variable assignments. For each cube α in the DNF, we prove that the formula after symmetry breaking under α is unsatisfiable. Additionally, we mechanically check that the three DNFs are indeed tautologies.

6 Experiments

We used the CaDiCaL[4] SAT solver developed by Biere [2] and ran the simulations on a cluster of Xeon E5-2690 processors with 24 cores per node. CaDiCaL supports proof logging in the DRAT format. We used DRAT-trim [29] to optimize the emitted proof of unsatisfiability. Afterwards we certified the optimized proofs with ACL2check, a formally verified checker [4]. All of the code that we used is publicly available on GitHub.[5] We have also made the logs of the computation publicly available on Zenodo.[6]

[4] Commit 92d72896c49b30ad2d50c8e1061ca0681cd23e60 of https://github.com/arminbiere/cadical

[5] https://github.com/marijnheule/Keller-encode

[6] https://doi.org/10.5281/zenodo.3755116

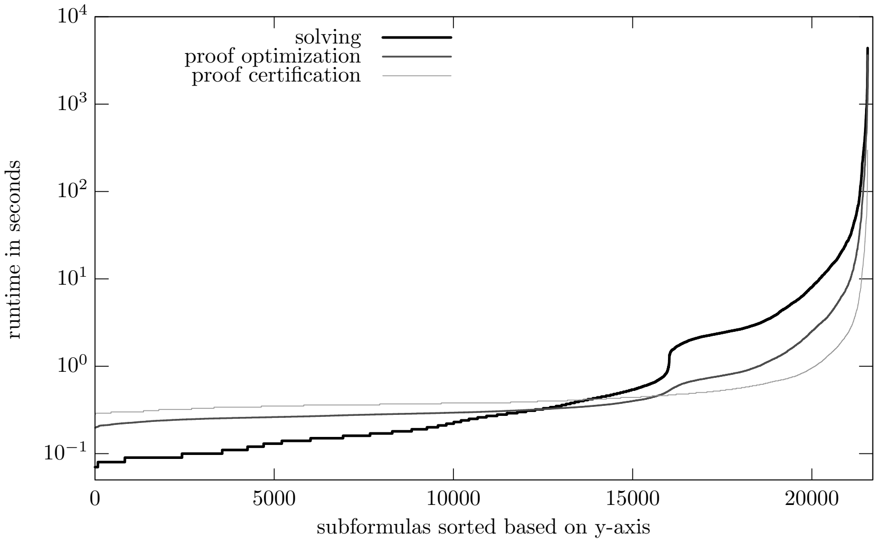

Fig. 6. Cactus plot of the runtime in seconds (logscale) to solve the 21 557 subformulas of $G_{7,3}$ as well as the times to optimize and certify the proofs of unsatisfiability.

6.1 Results for Dimension 7

Table 3 summarizes the running times are for experiment. The subformula-solving runtimes for $s = 3, 4$ and 6 are summarized in cactus plots in Figures 6, 7 and 8. The combined size of all unsatisfiability proofs of the subformulas of $s = 6$ is 224 gigabyte in the binary DRAT proof format. These proofs contained together $6.18 \cdot 10^9$ proof steps (i.e., additions of redundant clauses). The DRAT-trim proof checker only used $6.39 \cdot 10^8$ proof steps to validate the unsatisfiability of all subformulas. In other words, almost 90% of the clauses generated by CaDiCaL are not required to show unsatisfiability. It is therefore likely that a single DRAT proof for the formula after symmetry breaking can be constructed that is about 20 gigabytes in size. That is significantly smaller compared to other recently solved problems in mathematics that used SAT solvers [7, 10].

Table 3. Summary of solve times for $s = 3, 4, 6$. Times without a unit are in CPU hours. "No. Hard" is the number of subformulas which required more than 900 seconds to solve. "Hardest" is the solve time of the hardest subformula in CPU hours,

s	Tot. Solve	Avg. Solve	Proof Opt.	Proof Cert.	No. Hard	Hardest
3	43.27	7.23 s	22.46	4.98	28 form.	≈ 1.2
4	77.00	7.46 s	44.00	9.70	62 form.	≈ 2.7
6	81.85	7.63 s	34.84	14.53	63 form.	≈ 1.25

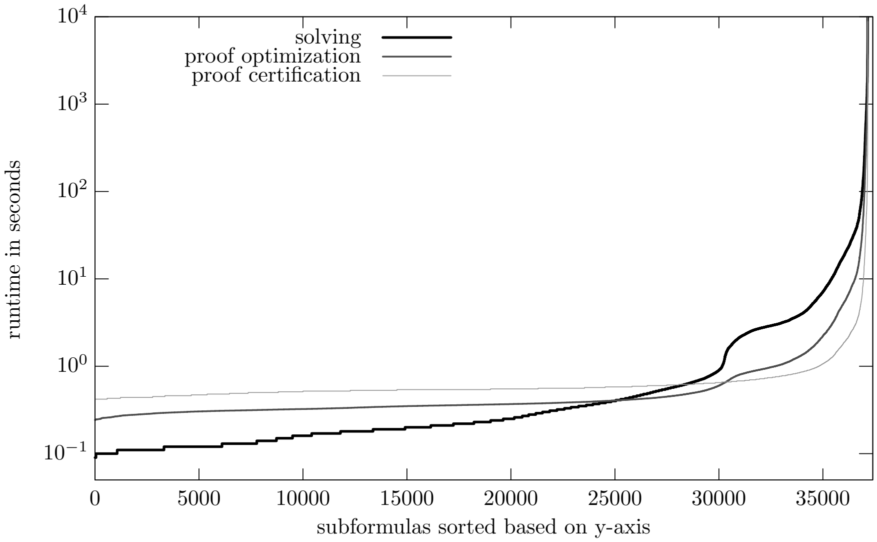

Fig. 7. Cactus plot of the runtime in seconds (logscale) to solve the 37 160 subformulas of $G_{7,4}$ as well as the times to optimize and certify the proofs of unsatisfiability.

We ran all three experiments simultaneously on 20 nodes on the Lonestar 5 cluster and computing on 24 CPUs per node in parallel. All instances were reported unsatisfiable and all proofs of unsatisfiability were certified by the formally verified checker. This proves Theorem 1.

6.2 Refuting Keller's Conjecture in Dimension 8

To check the accuracy of the CNF encoding, we verified that the generated formulas for $G_{8,2}$, $G_{8,3}$, $G_{8,4}$ and $G_{8,6}$ are satisfiable — thereby confirming that Keller's conjecture is false for dimension 8. These instances, by themselves, have too many degrees of freedom for the solver to finish. Instead, we added to the CNF the unit clauses consistent with the original clique found in the paper of Mackey [22] (as suitably embedded for the larger graphs). Specification of the vertices was per the method in Section 3 and 4. These experiments were run on Stanford's Sherlock cluster and took less than a second to confirm satisfiability.

Figure 9 shows an illustration of the clique of size 256 in $G_{8,2}$. This is the smallest counterexample for Keller's conjecture, both in the dimension ($n = 8$) as in the number of coordinates ($s = 2$). The illustration uses a black, dark blue, white, or light blue dot to represent a coordinate set to 0, 1, 2, or 3, respectively. Notice that for each pair of vertices holds that they have a complementary (black vs white or dark blue vs light blue) dot and at least one other different coordinate (a different color).

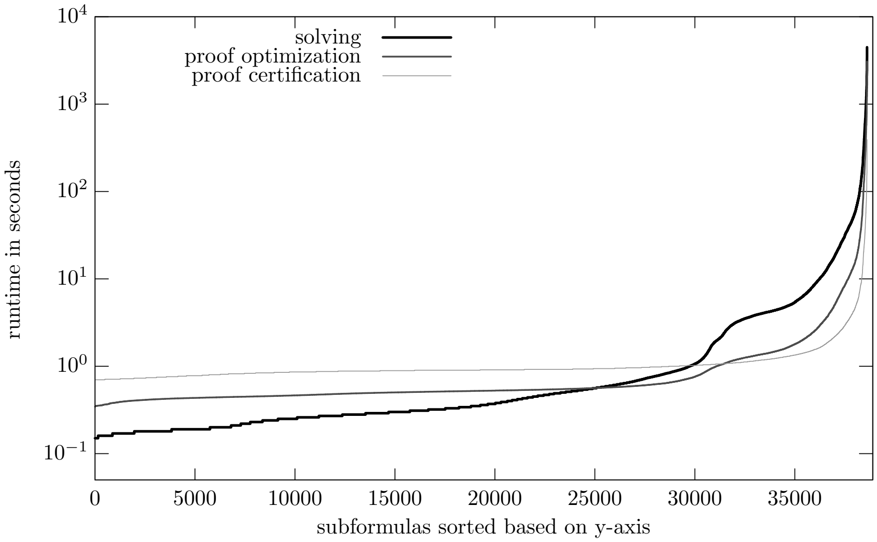

Fig. 8. Cactus plot of the runtime in seconds (logscale) to solve the 38 616 subformulas of $G_{7,6}$ as well as the times to optimize and certify the proofs of unsatisfiability.

7 Conclusions and Future Work

In this paper, we analyzed maximal cliques in the graphs $G_{7,3}$, $G_{7,4}$, and $G_{7,6}$ by combining symmetry-breaking and SAT-solving techniques. For the initial symmetry breaking we adapt some of the arguments of Perron. Additional symmetry breaking is performed on the propositional level and this part is mechanically verified. We partitioned the resulting formulas into thousands of subformulas and used a SAT solver to check that each subformula cannot be extended to a clique of size 128. Additionally, we optimized and certified the resulting proofs of unsatisfiability. As a result, we proved Theorem 1, which resolves Keller's conjecture in dimension 7.

In the future, we hope to construct a formally verified argument for Keller's conjecture, starting with a formalization of Keller's conjecture down to the relation of the existence of cliques of size 2^n in Keller graphs and finally the correctness of the presented encoding. This effort would likely involve formally verifying most of the theory discussed in the Appendix of the extended version of the paper. On top of that, we would like to construct a single proof of unsatisfiability that incorporates all the clausal symmetry breaking and the proof of unsatisfiability of all the subformulas and validate this proof using a formally verified checker.

Furthermore, we would like to extend the analysis to $G_{7,s}$, including computing the size of the largest cliques for various values of s. Another direction to consider is to study the maximal cliques in $G_{8,s}$ in order to have some sort of classification of all maximal cliques.

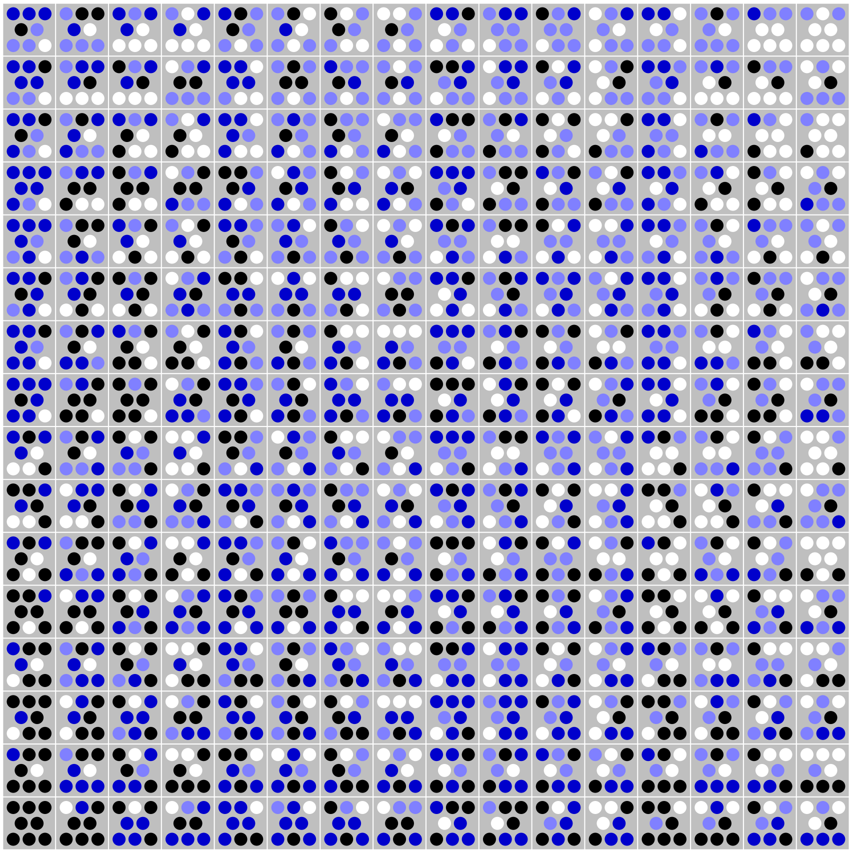

Fig. 9. Illustration of a clique of 256 vertices in $G_{8,2}$. Each “dice” with eight dots represents a vertex, and each dot represents a coordinate. A black, dark blue, white, and light blue dot represent a coordinate set to 0, 1, 2, and 3, respectively.

Acknowledgments

The authors acknowledge the Texas Advanced Computing Center (TACC) at The University of Texas at Austin, RIT Research Computing, and the Stanford Research Computing Center for providing HPC resources that have contributed to the research results reported within this paper. Joshua is supported by an NSF graduate research fellowship. Marijn and David are supported by NSF grant CCF-1813993. We thank Andrzej Kisielewicz and Jasmin Blanchette for valuable comments on an earlier version of the manuscript. We thank William Cooperman for helpful discussions on a previous attempt at programming simulations to study the half-integral case. We thank Alex Ozdemir for helpful feedback on both the paper and the codebase. We thank Xinyu Wu for making this collaboration possible.

References

1. Aloul, F.A., Markov, I.L., Sakallah, K.A.: Shatter: Efficient symmetry-breaking for boolean satisfiability. In: Proceedings of the 40th Annual Design Automation Conference. pp. 836–839. DAC '03, ACM, Anaheim, CA, USA (2003)
2. Biere, A.: CaDiCaL, Lingeling, Plingeling, Treengeling and YalSAT Entering the SAT Competition 2018. In: Proc. of SAT Competition 2018 – Solver and Benchmark Descriptions. Department of Computer Science Series of Publications B, vol. B-2018-1, pp. 13–14. University of Helsinki (2018)
3. Corrádi, K., Szabó, S.: A combinatorial approach for keller's conjecture. Period. Math. Hungar. **21**, 91–100 (1990)
4. Cruz-Filipe, L., Heule, M.J.H., Hunt Jr., W.A., Kaufmann, M., Schneider-Kamp, P.: Efficient certified RAT verification. In: Automated Deduction – CADE 26. pp. 220–236. Springer (2017)
5. Debroni, J., Eblen, J., Langston, M., Myrvold, W., Shor, P., Weerapurage, D.: A complete resolution of the Keller maximum clique problem. In: Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms. pp. 129–135. SIAM, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA (2011)
6. Hajós, G.: Uber einfache und mehrfache Bedeckung des n-dimensionalen Raumes mit einen Wurfelgitter. Math. Z. **47**, 427–467 (1942)
7. Heule, M.J.H.: Schur number five. In: Proc. of the 32nd AAAI Conference on Artificial Intelligence (AAAI 2018). pp. 6598–6606. AAAI Press (2018)
8. Heule, M.J.H., Hunt Jr., W.A., Wetzler, N.D.: Expressing symmetry breaking in DRAT proofs. In: Proc. of the 25th Int. Conference on Automated Deduction (CADE 2015). LNCS, vol. 9195, pp. 591–606. Springer, Cham (2015)
9. Heule, M.J.H., Kiesl, B., Biere, A.: Short proofs without new variables. In: Proc. of the 26th Int. Conference on Automated Deduction (CADE-26). LNCS, vol. 10395, pp. 130–147. Springer, Cham (2017)
10. Heule, M.J.H., Kullmann, O., Marek, V.W.: Solving and verifying the Boolean Pythagorean Triples problem via Cube-and-Conquer. In: Proc. of the 19th Int. Conference on Theory and Applications of Satisfiability Testing (SAT 2016). LNCS, vol. 9710, pp. 228–245. Springer, Cham (2016)
11. Heule, M.J., Schaub, T.: What's hot in the sat and asp competition. In: Twenty-Ninth AAAI Conference on Artificial Intelligence 2015. pp. 4322–4323. AAAI Press (2015)
12. Järvisalo, M., Heule, M.J.H., Biere, A.: Inprocessing rules. In: Proc. of the 6th Int. Joint Conference on Automated Reasoning (IJCAR 2012). LNCS, vol. 7364, pp. 355–370. Springer, Heidelberg (2012)
13. Keller, O.H.: Über die lückenlose Erfüllung des Raumes mit Würfeln. Journal für die reine und angewandte Mathematik **163**, 231–248 (1930)
14. Kisielewicz, A.P.: Rigid polyboxes and Keller's conjecture. Advances in Geometry **17**(2), 203–230 (2017)
15. Kisielewicz, A.P.: Towards resolving Keller's cube tiling conjecture in dimension seven. arXiv preprint arXiv:1701.07155 (2017)
16. Kisielewicz, A.P., Łysakowska, M.: On Keller's conjecture in dimension seven. The Electronic Journal of Combinatorics **22**(1), P1–16 (2015)
17. Konev, B., Lisitsa, A.: Computer-aided proof of Erdős discrepancy properties. Artificial Intelligence **224**(C), 103–118 (Jul 2015)

18. Kullmann, O.: On a generalization of extended resolution. Discrete Applied Mathematics **96-97**, 149 – 176 (1999)
19. Lagarias, J.C., Shor, P.W.: Keller's cube-tiling conjecture is false in high dimensions. Bulletin of the American Mathematical Society **27**(2), 279–283 (1992)
20. Lammich, P.: Efficient verified (UN)SAT certificate checking. In: Automated Deduction – CADE 26. pp. 237–254. Springer (2017)
21. Łysakowska, M.: Extended Keller graph and its properties. Quaestiones Mathematicae **42**(4), 551–560 (2019)
22. Mackey, J.: A cube tiling of dimension eight with no facesharing. Discrete and Computational Geometry **28**(2), 275–279 (2002)
23. McKay, B.D., Piperno, A.: nauty and traces user's guide (version 2.6), available at http://users.cecs.anu.edu.au/~bdm/nauty/nug26.pdf
24. Minkowski, H.: Diophantische Approximationen. Leipzig, B.G. Teubner (1907)
25. Perron, O.: Über lückenlose ausfüllung desn-dimensionalen raumes durch kongruente würfel. Mathematische Zeitschrift **46**(1), 1–26 (1940)
26. Perron, O.: Über lückenlose ausfüllung desn-dimensionalen raumes durch kongruente würfel. ii. Mathematische Zeitschrift **46**(1), 161–180 (1940)
27. Plaisted, D.A., Greenbaum, S.: A structure-preserving clause form translation. Journal of Symbolic Computation **2**(3), 293 – 304 (1986)
28. Szabó, S.: A reduction of Keller's conjecture. Periodica Mathematica Hungarica **17**(4), 265–277 (1986)
29. Wetzler, N., Heule, M.J., Hunt, W.A.: DRAT-trim: Efficient checking and trimming using expressive clausal proofs. In: International Conference on Theory and Applications of Satisfiability Testing. pp. 422–429. Springer (2014)