Unsatisfiability Proofs for Weight 16 Codewords in Lam's Problem

Curtis Bright
University of Windsor

Joint work with Kevin Cheung, Brett Stevens, Ilias Kotsireas, Vijay Ganesh

International Joint Conference on Artificial Intelligence
January, 2021

Overview

We solve **Lam's problem** from projective geometry by generating unsatisfiability proofs with satisfiability (SAT) solvers.

Computer Science team solves centuries-old math problem

And they had to search through a thousand trillion combinations to do it

Simply put . . .

Whew! To complete a mathematical investigation as complicated as the one recently accomplished by a team from the faculty of Engineering and Computer Science, every human being on earth would have to do 50,000 complex calculations.

The team, made up of Computer Science's Clement Lam, John McKay, Larry Thiel and Stanley Swiercz, took three years to solve a problem which had stumped mathematicians since the 1700s.

The problem: To find out whether "a finite projective plane of the order of 10" can exist.

The answer: "No."

So far it seems simple. But the reality is far different. The Concordia team had to search through 1,000,000,000,000,000 (that's a thousand trillion) combinations using one of the fastest supercomputers on earth to arrive at a solution.

The problem is so complex that the only way another party could prove the team's answer would be to do the calculations all over again — something which only the most intrepid investigators would even contemplate. The problem has already taken up the lifetimes of many eminent mathematicians.

The particular skill required was in organizing and programming the computer rather than in formulating the equations themselves.

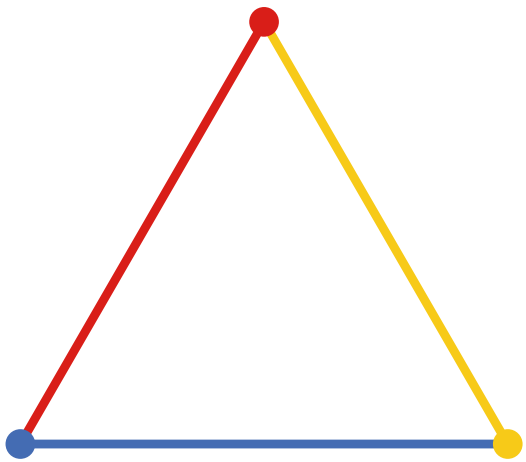
The skills which the team gained while solving the problem have future applications in developing communications networks and cryptography.

Charles Bélanger

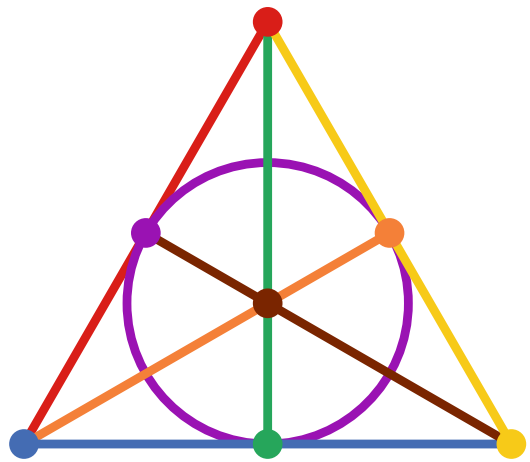
Let's see, if we added up the IQs of Concordia's crack Computer Science team, the number, though not in the trillions, would still be pretty high. Members of the team are (left to right) ***Larry Thiel, John McKay, Stanley Swiercz*** *and* ***Clement Lam.***

Projective planes

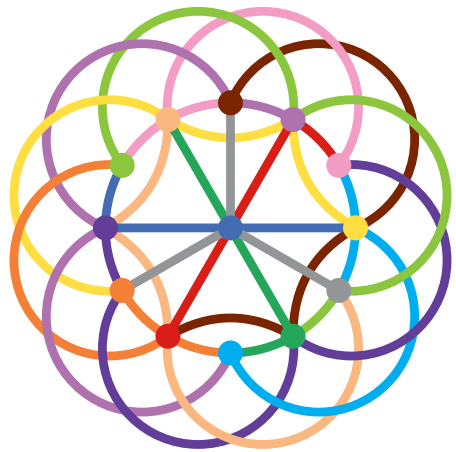
- Every pair of lines meet at a unique point.
- Every pair of points define a unique line.
- Every line contains $n + 1$ points for some *order* n.



order 1



order 2

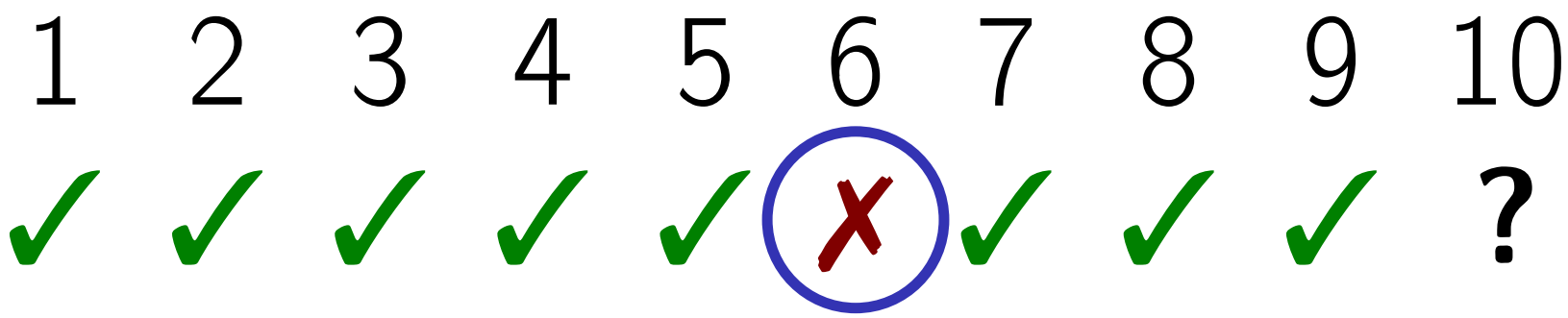


order 3

Projective planes of small orders

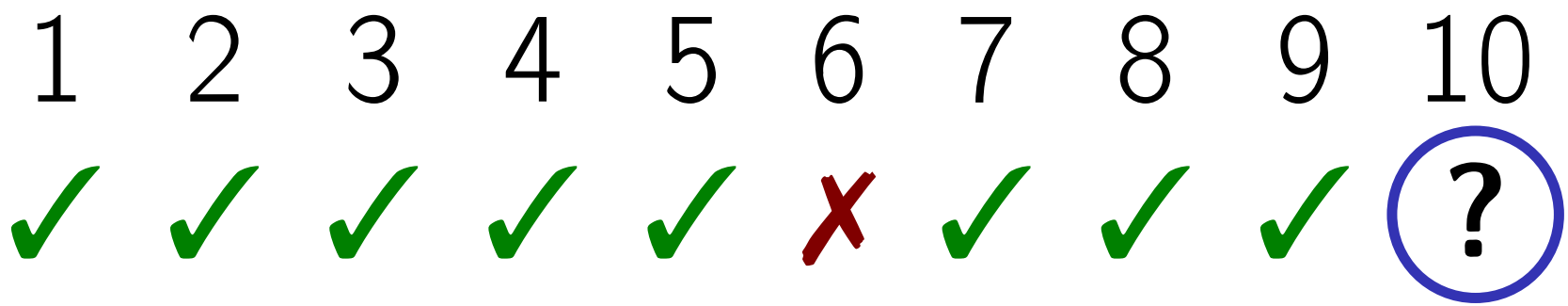
1	2	3	4	5	6	7	8	9	10
✓	✓	✓	✓	✓	✗	✓	✓	✓	?

Projective planes of small orders



Theoretical obstruction

Projective planes of small orders



No such plane known

No theoretical obstruction known

Projective planes of small orders

1	2	3	4	5	6	7	8	9	10
✓	✓	✓	✓	✓	✗	✓	✓	✓	?

Somehow, this problem has a beauty that fascinates me as well as many other mathematicians.

Clement Lam

Projective planes of small orders

1	2	3	4	5	6	7	8	9	10
✓	✓	✓	✓	✓	✗	✓	✓	✓	✗

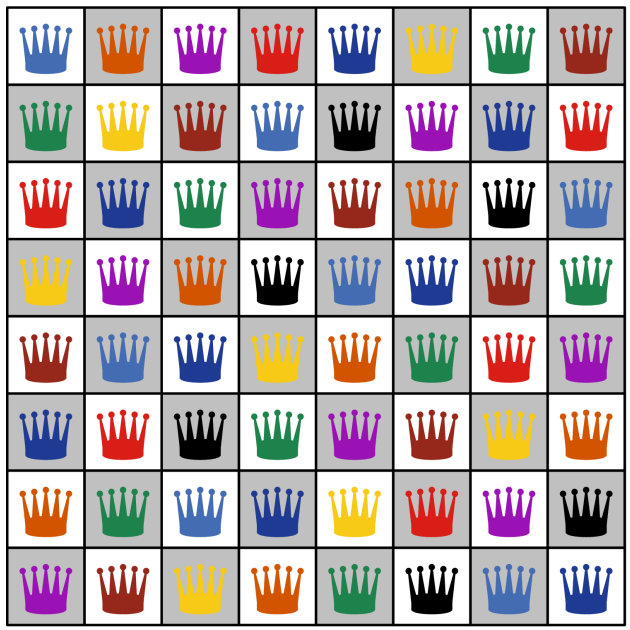
Supercomputer search (1980s)

However, the search was not verifiable

SAT solvers

SAT solvers can generate unsatisfiability proofs when problems have no solution.

For example, colouring the squares on a chessboard with no repetitions in any row, column, or diagonal. There is no solution with 8 colours.

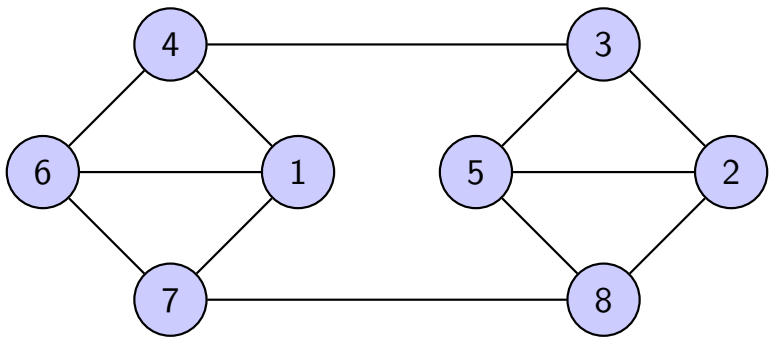


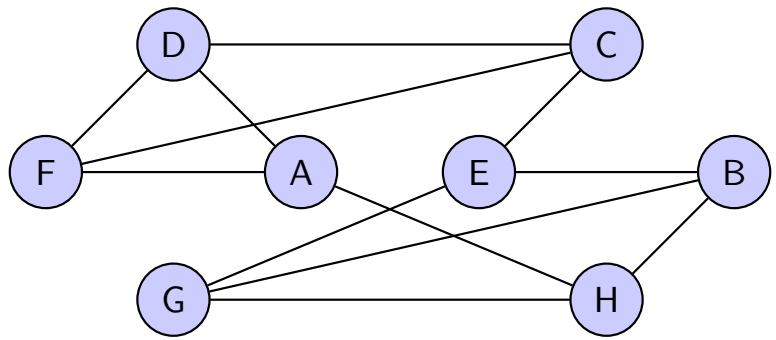
Optimal solution (9 colours)

Computer algebra systems (CASs)

CASs allow performing expressive mathematical calculations.

For example, checking if two graphs are isomorphic.





$1 \leftrightarrow D$ $3 \leftrightarrow E$ $5 \leftrightarrow G$ $7 \leftrightarrow A$

$2 \leftrightarrow B$ $4 \leftrightarrow C$ $6 \leftrightarrow F$ $8 \leftrightarrow H$

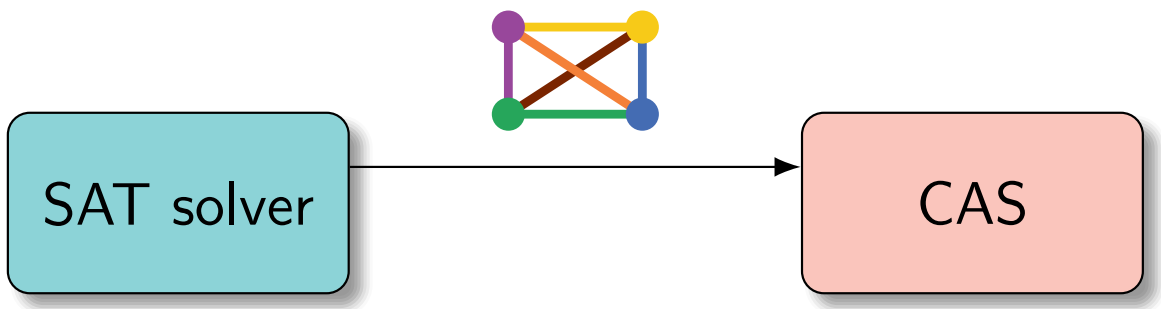
Explicit isomorphism

SAT + CAS

Search + Math

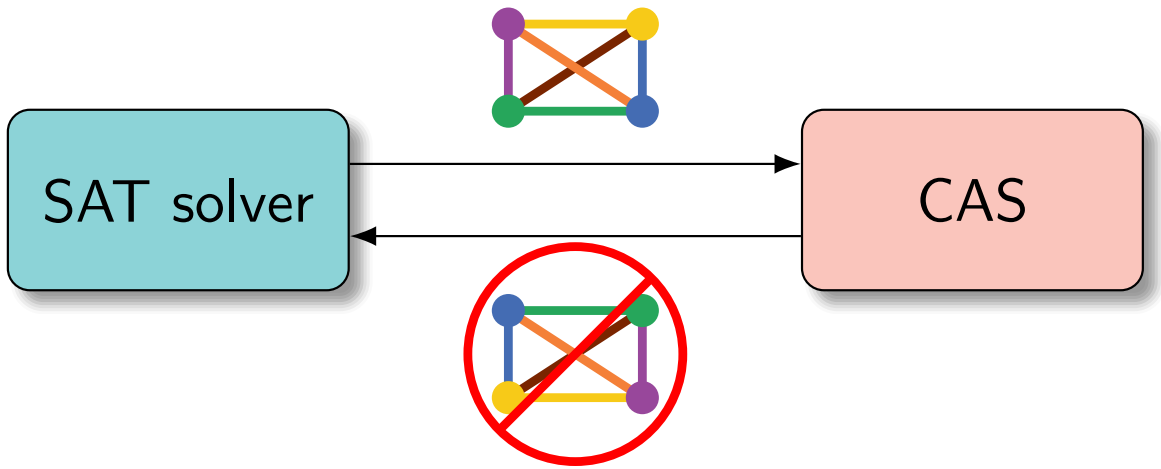
SAT+CAS isomorphism blocking

The SAT solver finds partial solutions and sends them to a CAS...



SAT+CAS isomorphism blocking

The SAT solver finds partial solutions and sends them to a CAS...



...and the CAS finds a nontrival isomorphism and blocks it.

Nonexistence proofs

We use a SAT+CAS method to generate certificates that a third party can use to verify our resolution of Lam's problem.

A subcase of Lam's problem that was previously solved in 16,000 computing hours was resolved by our system in 30 hours.

See `uwaterloo.ca/mathcheck` for the certificates and scripts.

Conclusion

Many mathematical problems stand to benefit from fast, verifiable, and expressive search tools.

Requires some knowledge of SAT and CAS—but avoids using special-purpose search code that is

- hard to write,
- even harder to make efficient,
- and extremely difficult to verify.

Future work

I'm actively looking for students and collaborators to extend and apply this paradigm to new applications.

Please get in touch if interested (and pass on the word to those who may be)!

Thank you!

`curtisbright.com`