Searching for Kochen–Specker Systems With Orderly Generation and Satisfiability Solving

Curtis Bright

University of Windsor

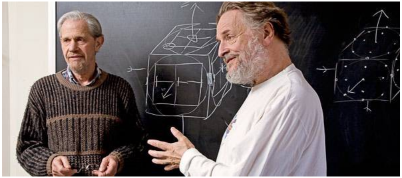
June 16, 2022

Simons Institute for the Theory of Computing

Satisfiability: Theory, Practice, and Beyond Reunion

The Free Will Theorem

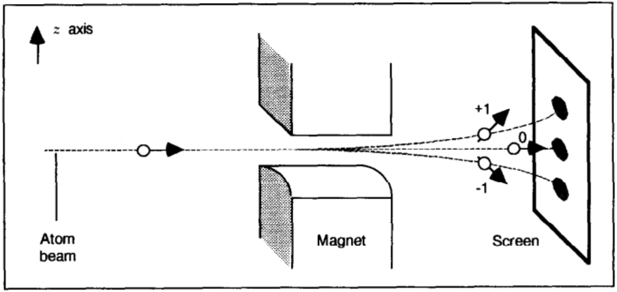
In 2006, John Conway and Simon Kochen proved the *Free Will Theorem*—if humans have free will then so do elementary particles.



Their proof crucially relies on a configuration of three dimensional vectors called a Kochen–Specker (KS) system.

The Stern–Gerlach Experiment (1922)

Shoot an atom of orthohelium through a magnetic field:



The atom will either deflect parallel or anti-parallel to the magnetic field or continue undisturbed. The *spin* of the atom (in the direction of the field) is $+1$, -1, or 0.

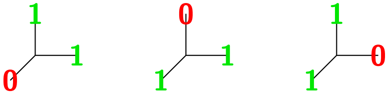
Image Source: S. Weinert

The SPIN Axiom

Suppose the $+1$ and -1 beams are combined together. This produces the *squared spin* which is either 1 or 0.

If you measure the squared spin in the x, y and z axes it will be zero **in exactly one of these directions**.

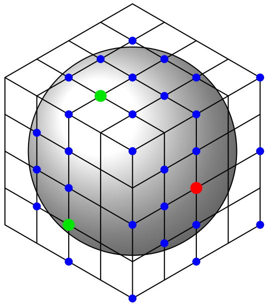
A consequence of quantum mechanics is that this holds in **any** three mutually orthogonal directions.



The 101 conspiracy

The KS Theorem

It is impossible to assign $\{\mathbf{0}, \mathbf{1}\}$ values to the following 31 vectors in a way that maintains the 101 conspiracy.



31 vector KS system of Conway and Kochen

The atom cannot have a predetermined spin in every direction.
Conway and Kochen use this to prove the Free Will Theorem.

Can We Do Better Than 31?

The best known result is that at least 22 vectors are required.[1]

This was shown by translating a hypothetical 21-vector KS system into a 21-vertex graph and performing an exhaustive search.

There are a huge number of such graphs and the computation took 75 CPU years using the best graph enumeration algorithms.

[1] S. Uijlen, B. Westerbaan. A Kochen-Specker System Has at Least 22 Vectors. *New Generation Computing*, 2016.

Reduction to SAT

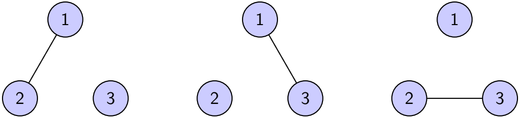
With some cleverness, many restrictive properties a "KS graph" must satisfy can be encoded in Boolean logic.

A SAT approach outperformed the previously used graph enumeration approach. However, a SAT solver generates many isomorphic copies of the same graph.

Instead, we use a **hybrid SAT and isomorph-free generation** approach (joint work with Zhengyu Li and Vijay Ganesh).

Isomorphisms

When generating combinatorial objects we only care about generating them *up to isomorphism*. Unfortunately, objects usually have many isomorphic representations.



$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \qquad \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \qquad \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

A graph with n vertices has up to $n!$ distinct isomorphic adjacency matrices. For efficiency, these should be detected and removed.

SAT Symmetry Breaking

A typical SAT approach is to add "symmetry breaking" constraints that remove as many isomorphic solutions as possible.

For example, you can order the rows of an adjacency matrix of a graph lexicographically.[2]

Typically many distinct isomorphic representations still exist, like

$$\begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix} \text{ and } \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}.$$

[2] M. Codish, A. Miller, P. Prosser, P. Stuckey. Constraints for symmetry breaking in graph representation. *Constraints*, 2019.

Isomorph-free Orderly Generation

Only "canonical" intermediate objects are recorded. The notion of canonicity is defined so that:

1. Every isomorphism class has exactly one canonical representative.
2. If an object is canonical then its parent in the search tree is also canonical.

Developed independently by Faradžev and Read in 1978.

Canonicity Example

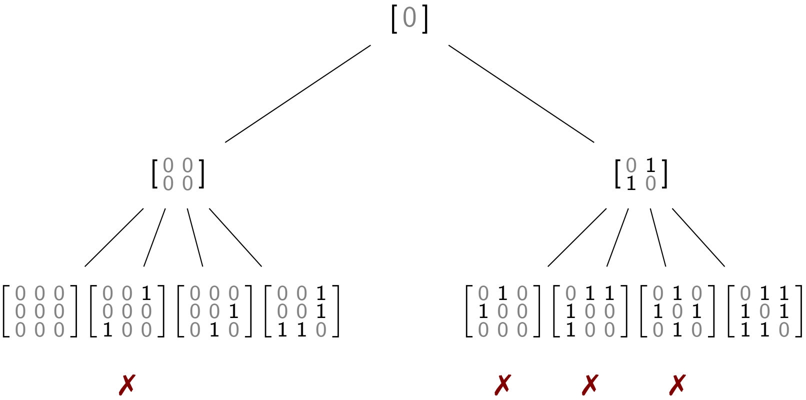
An adjacency matrix of a graph is *canonical* if the vector of its entries below the diagonal is lexicographically smallest (among all matrices in the same isomorphism class).

For example,

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \text{ and } \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

are isomorphic adjacency matrices but only the last is canonical.

Orderly Generation of Graphs



Orderly Generation: Key Points

Each canonical test is independent, making the method easy to parallelize.

Canonical testing introduces overhead but every negative test prunes a large part of the search space.

Verifying that a matrix is *noncanonical* is often fast, as it requires finding a single permutation of the vertices giving a lexicographically smaller adjacency matrix.

SAT and Isomorph-free Generation

Recently, isomorph-free generation and SAT solving were combined to solve Lam's problem from projective geometry.[3]

There have been surprisingly few attempts at combining isomorph-free generation and SAT solving.[4,5]

I'll now discuss applying orderly generation and SAT to the minimum Kochen–Specker problem.

---

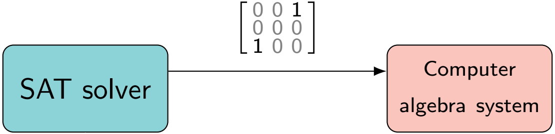
[3] C. Bright, K. Cheung, B. Stevens, I. Kotsireas, V. Ganesh. A SAT-based Resolution of Lam's Problem. *AAAI 2021*.

[4] T. Junttila, M. Karppa, P. Kaski, J. Kohonen. An adaptive prefix-assignment technique for symmetry reduction. *Journal of Symbolic Computation*, 2020.

[5] J. Savela, E. Oikarinen, M. Järvisalo. Finding periodic apartments via Boolean satisfiability and orderly generation. *LPAR 2020*.

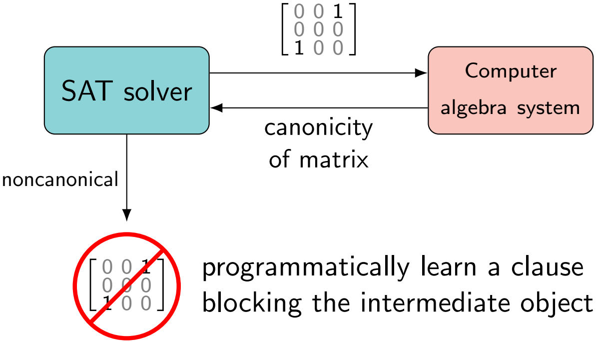
Orderly Generation in SAT

During the search the SAT solver will find partial solutions (complete definitions for the edges in some subgraphs)...



Orderly Generation in SAT

During the search the SAT solver will find partial solutions (complete definitions for the edges in some subgraphs)...



programmatically learn a clause blocking the intermediate object

KS Search Results

The speedup factor that we found when using SAT-based orderly generation in the search for KS systems of a given order:

order	speedup factor
16	6.5
17	13.6
18	37.8
19	104.5

The order 21 case was resolved in 25.7 CPU days (over 1000 times faster than the previous search).

KS Search Results

The speedup factor that we found when using SAT-based orderly generation in the search for KS systems of a given order:

order	speedup factor
16	6.5
17	13.6
18	37.8
19	104.5

The order 21 case was resolved in 25.7 CPU days (over 1000 times faster than the previous search).

The order 22 case was resolved in 5.3 CPU years. No KS system was found, so a *KS system must have at least 23 directions*.

A Promising Future

Isomorph-free generation and SAT can be combined to produce a hybrid solver capable of outperforming either a pure SAT or isomorph-free generation approach.

The approach is very general, can be applied to many combinatorial generation problems, and I believe it has yet to be used to its full potential.

Thank You!

`curtisbright.com`