A SAT and Orderly Generation Approach in the Quest for the Minimum Kochen–Specker System

Curtis Bright
University of Windsor

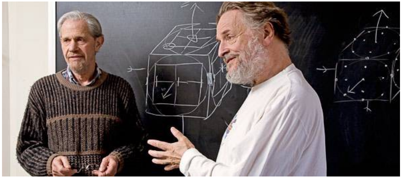
June 6, 2023

CanaDAM 2023

Discrete Mathematics in Quantum Information Processing

The Free Will Theorem

Conway and Kochen proved the *Free Will Theorem* in 2006—if humans have have free will then so do quantum particles.[1] Last year the assumption that humans have free will was removed.[2]

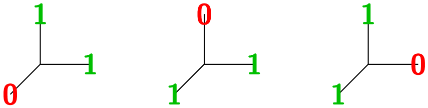


Their proof relies on a finite set of vectors called a Kochen–Specker (KS) system.

[1] J. Conway, S. Kochen. The Free Will Theorem. *Foundations of Physics*, 2006.
[2] S. Kochen. On the Free Will Theorem. Preprint, 2022.

The SPIN Axiom

The "squared spin" of a spin-1 particle in any three mutually orthogonal directions will be **0** **in exactly one of these directions**.

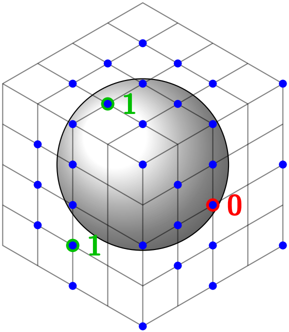


The 101 "conspiracy"

In particular, two orthogonal directions cannot both have a squared spin of **0**.

The KS Theorem (1967)

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

A *KS system* is a set of vectors on which the particle **cannot** be assigned a spin value for each vector simultaneously.

KS Systems in Quantum Information Processing

In 2014, a 6-dimensional 7-experiment KS system was shown to be the simplest possible KS system with certain symmetry.[3]

Later in 2014, this KS system was used to experimentally certify measurements were accessing a quantum system rather than a classical system.[4]

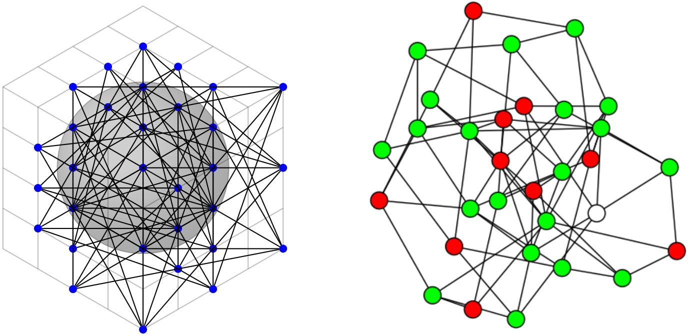
KS systems have other applications, like providing security against classical attacks to quantum cryptographic protocols.[3]

[3] P. Lisoněk, P. Badziąg, J. Portillo, A. Cabello. Kochen-Specker Set with Seven Contexts. *Physical Review A*, 2014.

[4] G. Cañas et al. Applying the Simplest Kochen-Specker Set for Quantum Information Processing. *Physical Review Letters*, 2014.

KS Graphs and 101-colourability

Consider the graph formed by a KS system by connecting all pairs of orthogonal vectors:



The property required for the KS theorem is that the graph cannot be *101-coloured* (triangles have exactly one colour-**0** vertex and edges have at most one colour-**0** vertex).

Are 31 Vectors Minimal in 3D?

Previously, it was known that at least 22 vectors are required.[5]

This was shown by performing an exhaustive enumeration for all **non**-101-colourable graphs with up to 21 vertices (up to properties that a KS graph must satisfy like being squarefree).

The computation took 75 CPU years using the state-of-the-art graph enumeration tool `nauty`.[6]

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

[6] B. McKay, A. Piperno. Practical Graph Isomorphism, II. *Journal of Symbolic Computation*, 2014.

SAT to the Rescue

Satisfiability (SAT) solvers take a formula in Boolean logic and try to solve it, i.e., find an assignment that makes it true.

Example: Is $(x \lor y) \land (\neg x \lor \neg y)$ satisfiable?

SAT to the Rescue

Satisfiability (SAT) solvers take a formula in Boolean logic and try to solve it, i.e., find an assignment that makes it true.

Example: Is $(x \lor y) \land (\neg x \lor \neg y)$ satisfiable?
Yes; take x to be true and y to be false.

SAT solvers are used *declaratively*—you state the constraints of your problem, and they search for a solution. They can be amazingly effective, even for problems not arising from logic.[7]

Tomorrow at 5:30 in 1L06, Cayden Codel will be talking about SAT encodings (rescheduled from yesterday).

[7] C. Bright, J. Gerhard, I. Kotsireas, V. Ganesh. Effective Problem Solving Using SAT Solvers. *Maple Conference 2019*.

Reduction to SAT

With some cleverness, many restrictive properties a KS graph must satisfy can be encoded in Boolean logic, so a SAT solver can search for KS graphs.

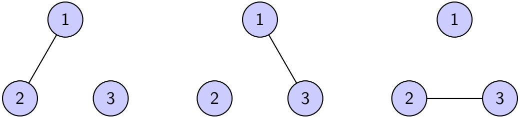
The SAT solver outperforms the previously used graph enumeration approach. However, the solver generates many isomorphic copies of the same graph.

Thus, we combine SAT with isomorph-free exhaustive generation (also previously used to solve *Lam's problem*).[8]

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

Isomorphisms

Objects typically have many isomorphic representations, e.g., an n-vertex graph has up to $n!$ representations.



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

When enumerating combinatorial objects we would like to generate them in a *isomorph-free* way.

Orderly Generation

Only "canonical" intermediate objects are recorded. The notion of canonicity is defined so that every isomorphism class has exactly one canonical representative.

Developed independently by Faradžev and Read in 1978.[9,10]

[9] I. Faradžev. Constructive enumeration of combinatorial objects. *Problèmes combinatoires et théorie des graphes*, 1978.

[10] R. Read. Every one a winner or how to avoid isomorphism search when cataloguing combinatorial configurations. *Annals of Discrete Mathematics*, 1978.

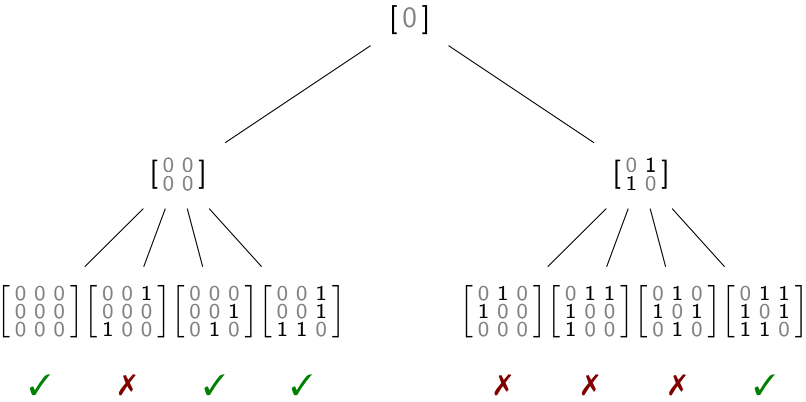
Definition of Canonicity

An adjacency matrix is *canonical* if its "vector representation" is lex-minimal among all matrices in the same isomorphism class.

Adj. matrix	$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$
Vector rep.	$\begin{bmatrix} 1 & 0 & 0 \end{bmatrix} >_{\text{lex}}$	$\begin{bmatrix} 0 & 1 & 0 \end{bmatrix} >_{\text{lex}}$	$\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$
Canonical?	✗	✗	✓

A consequence of this definition is that a noncanonical matrix *never becomes canonical* after appending a row and column.

Orderly Generation of Graphs



Canonical testing introduces overhead, but every negative test prunes a large part of the search space (and tests that are negative are usually fast).

Isomorph-free Exhaustive Generation and SAT

I believe there should be more research combining the well-established methods of isomorph-free exhaustive generation with the well-established methods of SAT solving.

Recently there have been some visionary research along these lines,[11,12,13] but it has yet to be used to its full potential.

---

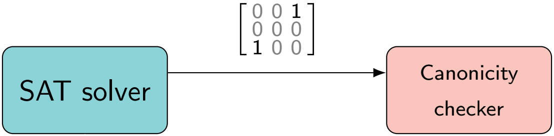
[11] T. Junttila, M. Karppa, P. Kaski, J. Kohonen. An adaptive prefix-assignment technique for symmetry reduction. *Journal of Symbolic Computation*, 2020.

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

[13] M. Kirchweger, M. Scheucher, S. Szeider. A SAT Attack on Rota's Basis Conjecture. *SAT 2022*.

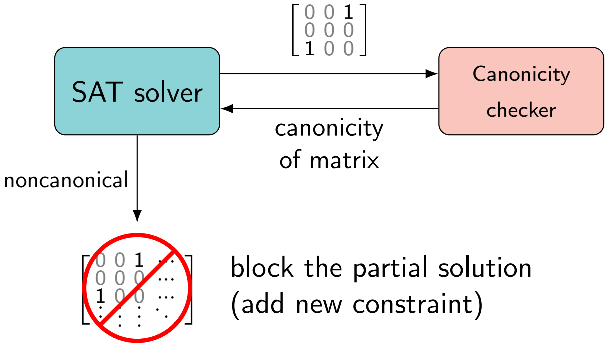
Orderly Generation in SAT

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). . .



KS Search Results

The time it takes to run an exhaustive search for KS graphs of order n using `nauty`, pure SAT, and SAT + orderly generation:

n	`nauty`	Pure SAT	SAT+O.G.	Speedup
17	25.3 m	8.8 m	0.3 m	$\sim$66x
18	455.6 m	266.6 m	1.7 m	$\sim$209x
19	9,506.4 m	11,705.1 m	8.9 m	$\sim$1,193x
20			83.8 m	
21	75 years		20 hours	$\sim$32,649x
22			19 days	
23			6 years	

No KS system was found,[14] so a three-dimensional KS system **must have at least 24 directions**.

[14] Z. Li, C. Bright, V. Ganesh. An SC-Square Approach to the Minimum Kochen–Specker Problem. *SC-Square Workshop 2022*.

A Promising Future

I have repeatedly seen SAT + isomorph-free generation, and more generally SAT + computer algebra, lead to exponential speedups over SAT or computer algebra alone.[15]

review articles

The science of less-than-brute force.

BY CURTIS BRIGHT, ILIAS KOTSIREAS, AND VIJAY GANESH

When Satisfiability Solving Meets Symbolic Computation

There are countless problems to which the method can be applied.

Thank You! `curtisbright.com`

[15] C. Bright, I. Kotsireas, V. Ganesh. When Satisfiability Checking Meets Symbolic Computation. *Communications of the ACM 2022.*