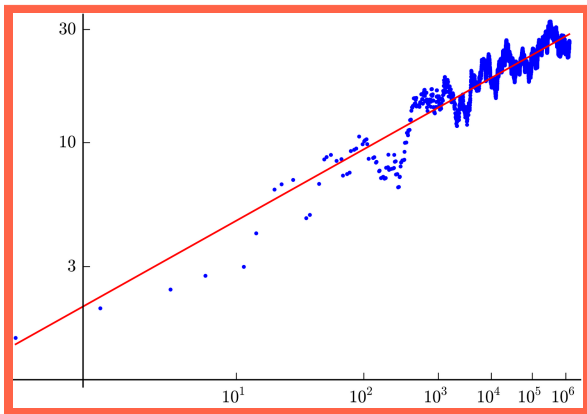


Why the excitement?



- ▶ **Tension** Mathematics wants exact statements; AI gives plausible attempts
- ▶ **Marriage** Plausible attempts are useful when bad ones are cheap to catch
- ▶ **Method** Let AI suggest many things; keep only what mathematics accepts

Why the excitement?



- ▶ **Myth** Mathematics = theorem, proof, next theorem
- ▶ **Reality** There is life beyond theorem proofs (interesting in its own right)
- ▶ **Daily work** Examples, pictures, computations, simplifications, failed ideas

Why the excitement?

What mathematicians actually do



Before anything becomes a theorem
there are examples, pictures, computations
and quite a lot of dead ends

▶ Myth Ma

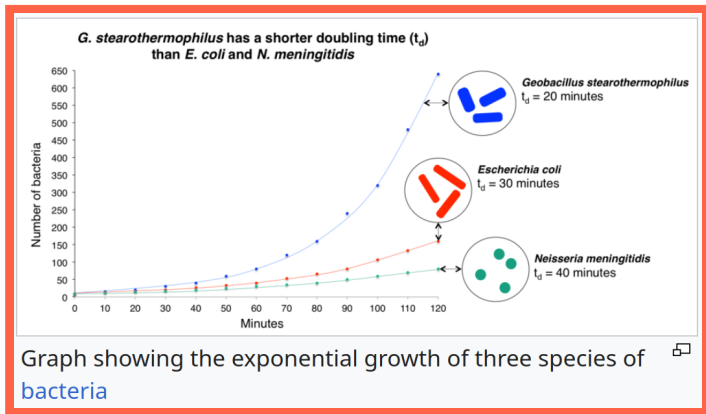
▶ Reality T

▶ Daily work

Examples, pictures, computations, simplifications, failed ideas

own right)

Why the excitement?



- ▶ **Worlds** Knots (more later), graphs, groups, polytopes, programs, proofs, ...
- ▶ **Problem** The rules are exact, but the number of choices explodes
- ▶ **Opening** A machine can search places humans would not patiently search

Why the excitement?

Structured chaos



▶ World

▶ Prob

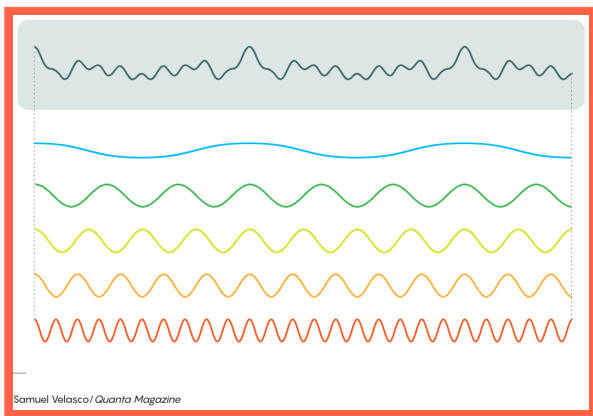
▶ Opening

The rules are precise enough to run
the choices are too many to inspect by hand
and success is often visible afterwards

proofs, ...

A machine can search places humans would not patiently search

Why the excitement?



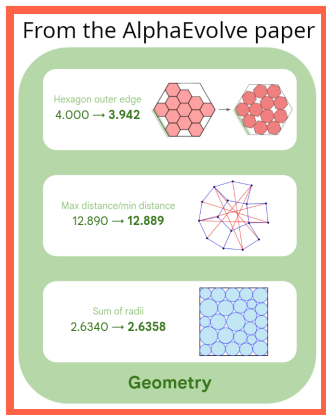
- ▶ **Rare** Good tools are hard to find, and people reuse them mercilessly
- ▶ **New** AI is not just a faster calculator; it changes the search
- ▶ **Analogy** Imagine you have 1000 people to throw at a problem

Why mathematicians get excited



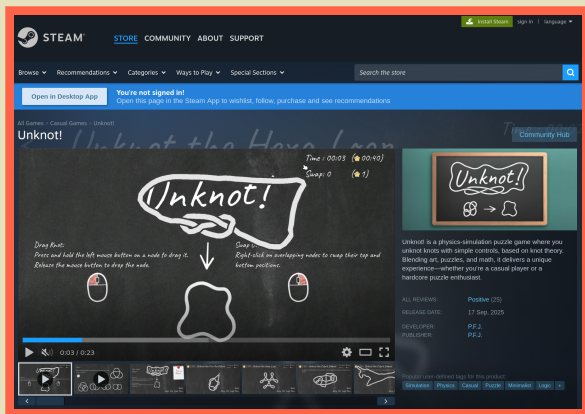
- ▶ **Rare** AI is useful in exact but huge mathematical worlds
new tools are rare, and this one is new (in math circles)
- ▶ **New** thus, excitement comes from exploration plus checkability
- ▶ **Analogy** Imagine you have 1000 people to throw at a problem

Why the excitement?



- ▶ AlphaEvolve Let an LLM evolve code, then score the mutants automatically
- ▶ Example Pack small hexagons into the smallest possible big hexagon
- ▶ Why does this work? Imagine you have 1000 people to throw at this problem

Honourable mentions



FunSearch: LLM + evolution, writes programs for hard searches

Ramanujan Machine: number search, conjectures formulas for constants

AlphaGeometry/Proof: LLM + symbolic/RL search, solves olympiad problems

Gamification: humans + scoring, turns search into a playable problem + AI harvest

Computer verification: LLM translates human math to computer math; Lean checks it
many more, but I will zoom into only one now

Knot land



- ▶ **Object** A knot is a closed loop of string in space
- ▶ **Rule** You may move it, but not cut it or pass it through itself
- ▶ **Question** Identify it! (E.g. is this tangle secretly just the plain circle?)

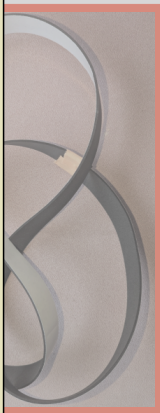
Knot land



The basic game

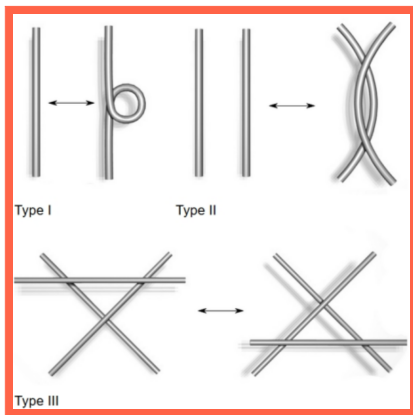
Binary (adaptive, cleaned) Skeleton (adaptive threshold) Overlay on Original

The string may wiggle wildly but the knot type is unchanged unless we cheat topologically through itself



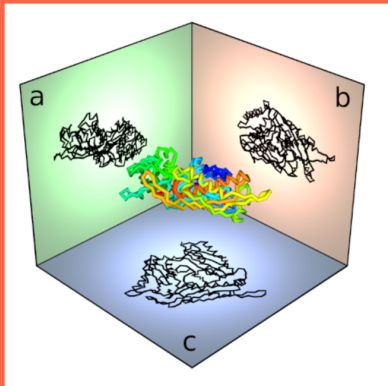
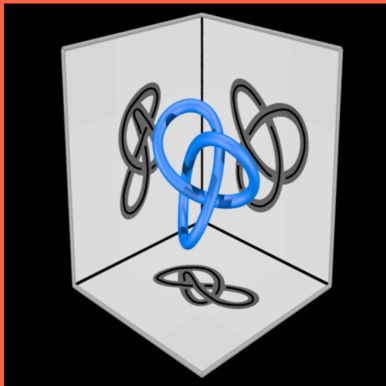
- ▶ **Object** A knot is a closed loop of string that cannot be pulled apart without cutting it.
- ▶ **Rule** You may move the string around in space, but you cannot pass it through itself.
- ▶ **Question** Identify it! (E.g. is this tangle secretly just the plain circle?)

Knot land



- ▶ **Diagram** We draw the loop in the plane with over/under crossings
- ▶ **Annoyance** The same knot has many very different drawings
- ▶ **Moves** Small legal changes relate drawings of the same knot

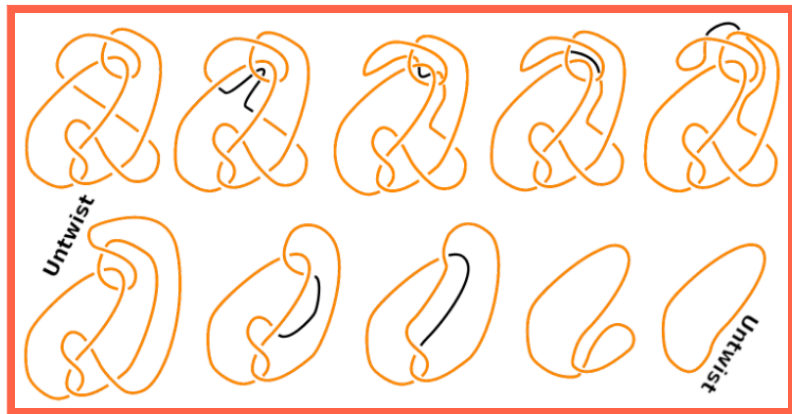
Every drawing lies



A diagram is only one photograph
of a three-dimensional object
and some photographs are very unhelpful

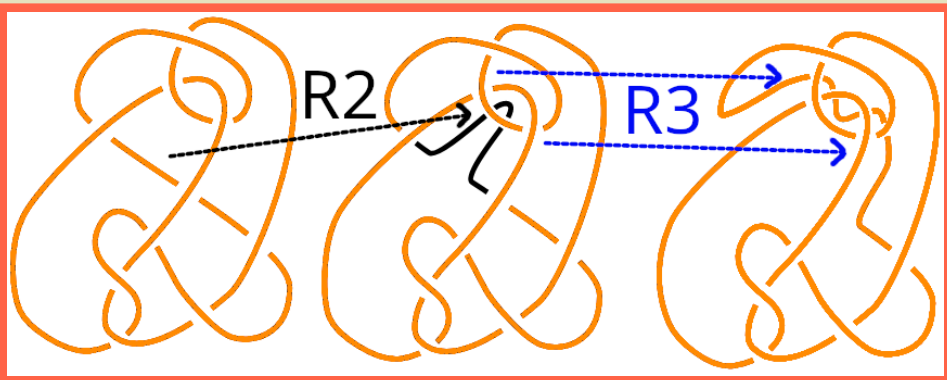
Moves Small legal changes relate drawings of the same knot

Knot land



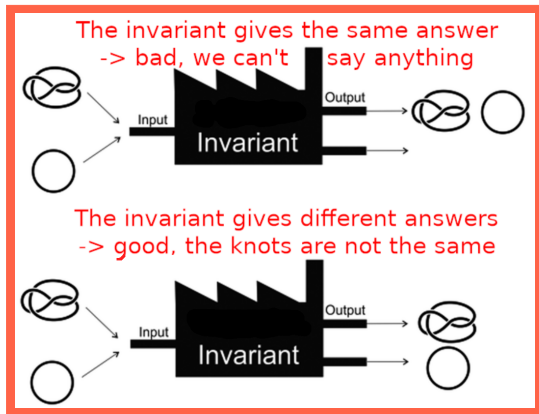
- ▶ **Unknot** The loop is not knotted, but the drawing may hide this
- ▶ **Difficulty** A simple knot can have a very complicated diagram
- ▶ **Search** Find the moves that reveal the simple picture

A puzzle with receipts



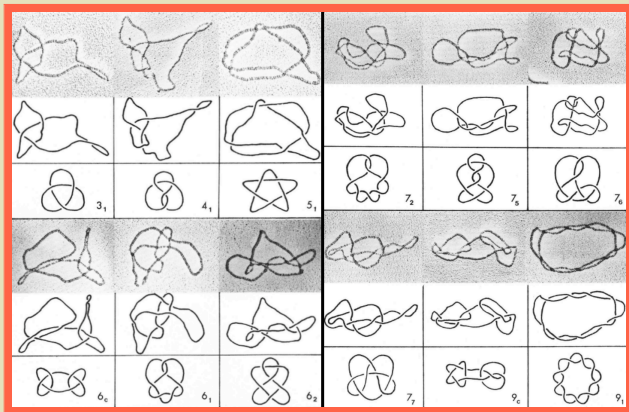
If the machine finds the moves
we can replay the moves ourselves
and check that nothing illegal happened

► Search Find the moves that reveal the simple picture



- ▶ **Fingerprints** A knot invariant is a computable shadow of the knot
- ▶ **Use** Different fingerprints prove that two knots are different
- ▶ **Caveat** Same fingerprint does not always mean same knot

Fingerprints, not souls



Invariants are not the knot itself
but they are cheap evidence
and cheap evidence changes the search

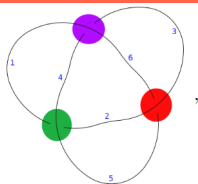
► Fingerprint

► Use D

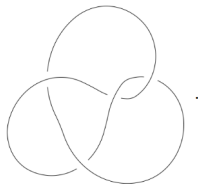
► Caveat Same fingerprint does not always mean same knot

Knot land

$PD[X[1, 5, 2, 4], X[3, 1, 4, 6], X[5, 3, 6, 2]] \iff$

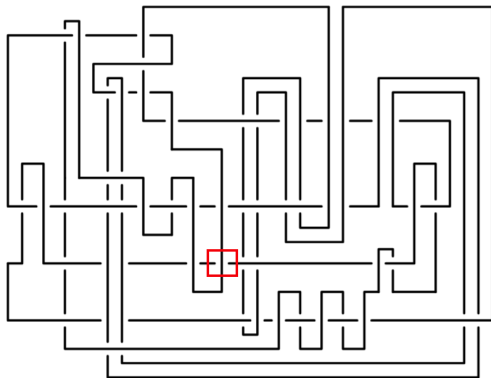


$PD[X[4, 2, 5, 1], X[8, 6, 1, 5], X[6, 3, 7, 4], X[2, 7, 3, 8]] \iff$



- ▶ **Exact** Diagrams can be encoded as finite combinatorial data
- ▶ **Huge** The number of possible moves quickly becomes silly
- ▶ **Perfect** This is exactly where AI search can be useful

Why knots fit the story



Exact rules, huge search spaces
visual objects, checkable output
and plenty of room for bad intuition

▶ Exact Diag

▶ Huge The

▶ Perfect This is exactly where AI search can be useful

Two case studies

From Applebaum et al.:

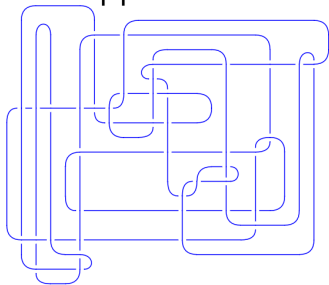
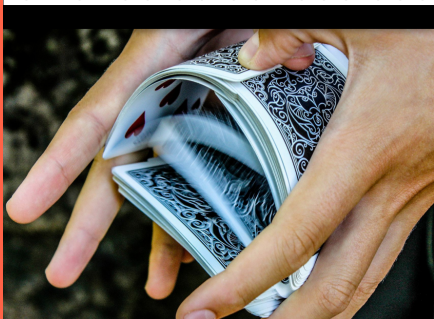
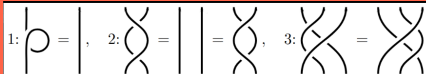


FIGURE 17. A 42-crossing hard unknot diagram with 6225 R3-equivalent diagrams that we have not been able to simplify by calling SnapPy's 'global' heuristic 10000 times.

-
- ▶ **Case 1** Can a machine untie a knot diagram?
 - ▶ **State** The current drawing, encoded as combinatorial data
 - ▶ **Action** Try one legal local move, then try again

Two case studies

Reinforcement learning translation



State = diagram

action = legal move (1+2 add/remove cards, 3 shuffles)

reward = progress toward simplicity

► Case 1 C

► State T

► Action Try one legal local move, then try again

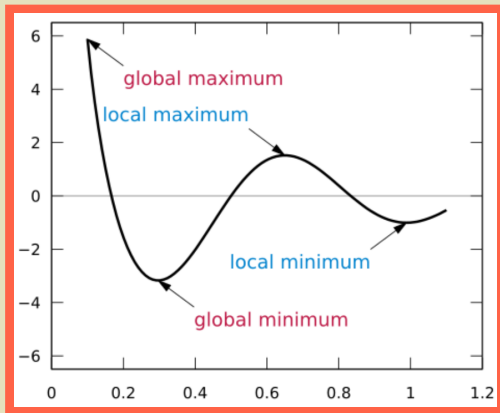
Two case studies



-
- ▶ **Trap** The best route may first make the diagram worse
 - ▶ **Reward 2** Give reward for successful “add-shuffle-remove” move
 - ▶ **AI part** Learn which bad looking moves are worth trying

Two case studies

The useful move may look stupid



▶ Trap T

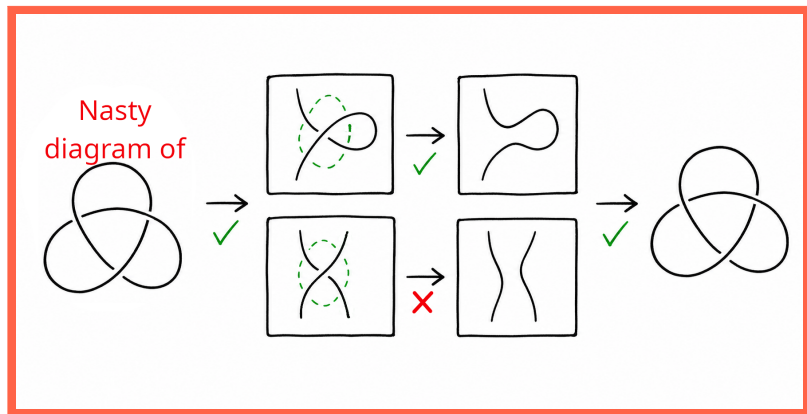
▶ Reward

▶ AI part

Humans like monotone simplification
knots often refuse to cooperate (stuck in a local minimum)
so the machine gets paid to be patient

Learn which bad looking moves are worth trying

Two case studies



- ▶ **Output** A concrete sequence of moves, not a philosophical essay
- ▶ **Check** Replay the sequence and verify every local step
- ▶ **Moral** The search is learned, but the certificate is mathematical

Two case studies

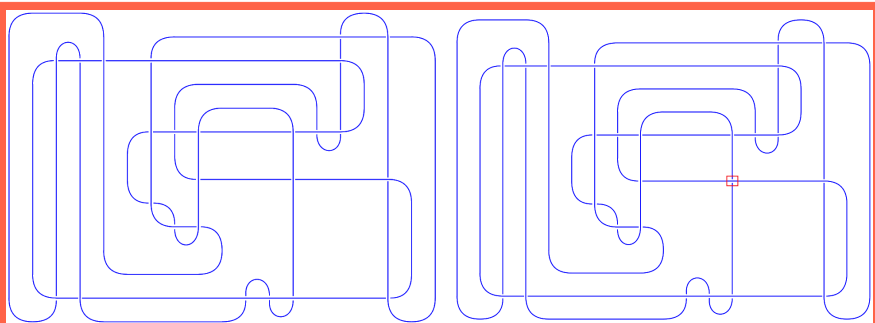
Results

run	resolved_count	resolved_pct	median_min_crossings	max_min_crossings	
0	1	367	95.324675	0.0	46
1	2	367	95.324675	0.0	38
2	3	365	94.805195	0.0	39
3	4	357	92.727273	0.0	46
4	5	357	92.727273	0.0	41
5	6	360	93.506494	0.0	46
6	7	367	95.324675	0.0	39
7	8	368	95.584416	0.0	46
8	9	365	94.805195	0.0	46
9	10	368	95.584416	0.0	40

The unknotter can undo all known
hard unknot diagrams
with roughly 95% probability

Moral The search is learned, but the certificate is mathematical

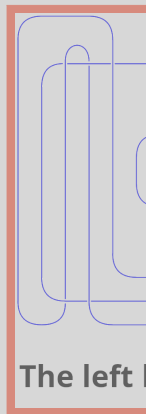
Two case studies



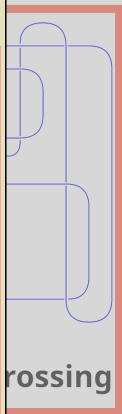
The left knot gets easier by flopping the marked crossing

- ▶ **Problem** How many crossing flops are needed to untie a knot?
- ▶ **Catch** The useful crossing may not be visible in the first picture
- ▶ **Trick** First change the picture, then try the crossing flop

Two case studies

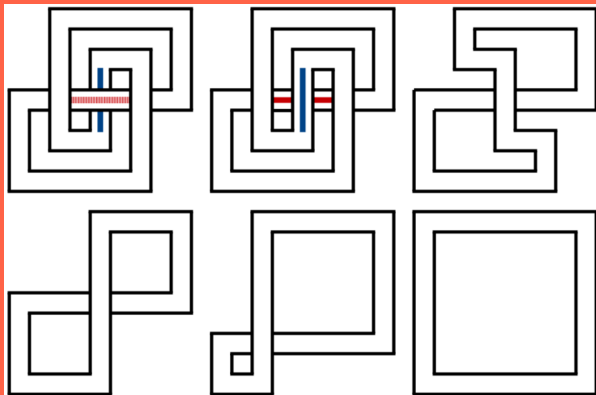


The left



crossing

Crossing flop



► Problem

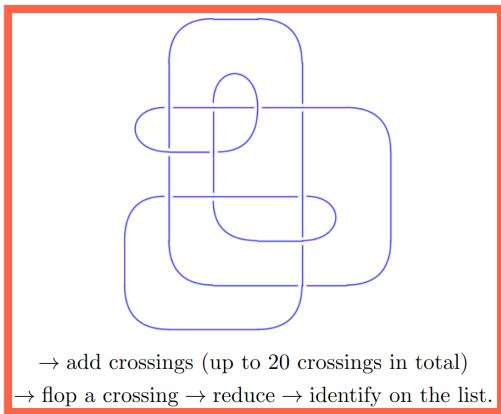
► Catch

► Trick

At one crossing, swap over and under
do this enough times and every knot unties
the question is how few times suffice

First change the picture, then try the crossing flop

Two case studies



- ▶ **Search** Let an RL agent create many alternative diagrams
- ▶ **Guess** Look for diagrams where one crossing flop seems promising
- ▶ **Test** Flop the crossing, simplify, and see what knot remains

Why this is hard

From Kauffman & Lambropoulou

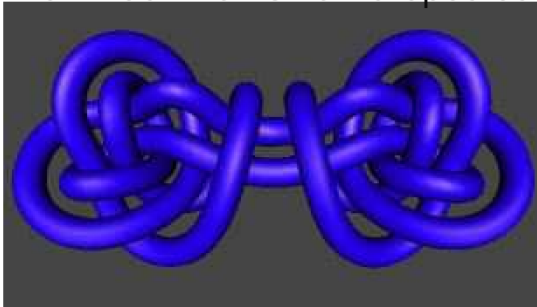


Figure 8 - Another Hard Unknot

▶ Search

▶ Guess

▶ Test

The same knot has infinitely many drawings
a useless flop in one drawing may become useful in another
so the search is over pictures, not just crossings

Flop the crossing, simplify, and see what knot remains

Two case studies



- ▶ **Point** The agent is not proving the unknotting number directly
- ▶ **Role** It guesses good places to perform a hard local edit
- ▶ **Moral** AI explores diagrams; mathematics checks the aftermath

Takeaway from Case 2

Knot	KnotInfo	current workbook
11a14	[2, 3]	[2, 2]
11a18	[2, 3]	[2, 2]
11a83	[2, 3]	[2, 2]
11n23	[2, 3]	[2, 2]
12a41	[2, 3]	[2, 2]
12a49	[2, 3]	[2, 2]
12a240	[2, 3]	[2, 2]
12a244	[2, 3]	[2, 2]
13a568	[2, 3]	[2, 2]
13a579	[1, 3]	[1, 2]
13a616	[3, 4]	[3, 3]
13a650	[2, 4]	[2, 2]
13a656	[1, 3]	[1, 2]
13a660	[3, 4]	[3, 3]
13a1698	[1, 4]	[1, 3]
13a1712	[1, 3]	[1, 2]
13a3147	[1, 3]	[1, 2]
13a3149	[1, 3]	[1, 2]
13a3177	[1, 3]	[1, 2]
13a3184	[1, 3]	[1, 2]
13a3212	[3, 4]	[3, 3]
13a3236	[1, 3]	[1, 2]
13a3268	[1, 3]	[1, 2]
13n616	[3, 4]	[3, 3]
13n669	[3, 4]	[3, 3]
13n675	[2, 4]	[2, 3]
13n709	[3, 4]	[3, 3]
13n735	[2, 4]	[2, 3]

TABLE 1. Current exact improvements visible in the self-improving workbook. In some cases the interval collapses to the exact value $u(K) = a$ if $[a, a]$ is displayed.

- **Point** The a ectly
- **Role** It guesses and one is able to improve human-known-results
- **Moral** AI explores diagrams; mathematics checks the aftermath

Why the excitement?



- **Tenacity** Mathematics wants exact statements. AI gives plausible attempts
- **Message** Plausible attempts are useful when bad ones are cheap to catch
- **Method** Let AI suggest many things; keep only what mathematics accepts

Knot land



- **Object** A knot is a closed loop of string in space
- **Rule** You may move it, but not cut it or pass it through itself
- **Question** Identify it! (E.g. is this tangle secretly just the plain circle?)

Two case studies



- **Case 1** State = diagram
- **State 2** action = legal move (1-2 add/remove cards, 3 shuffle)
reward = progress toward simplicity
- **Action** Try one legal local move, then try again

Why the excitement?



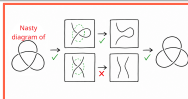
- **Text** All is treated as "1000 unfeasible but productive candidates". It is used as a source of attempts (it does a lot of them) and attempts are cheap to reject (as it doesn't matter if the often wrong)
- **Message** Let AI suggest many things; keep only what mathematics accepts

Knot land



- **Diagram** We draw the loop in the plane with over/under crossings
- **Appearance** The same knot has many very different drawings
- **Moves** Small legal changes relate drawings of the same knot

Two case studies



- **Output** A concrete sequence of moves, not a philosophical essay
- **Check** Replay the sequence and verify every local step
- **Moral** The search is learned, but the certificate is mathematical

Why the excitement?



- **Text** All is useful in exact but huge mathematical worlds new tools are rare, and this one is new (in much circles) thus, excitement comes from exploration plus checkability
- **Message** Imagine you have 1000 people to throw at a problem

Knot land



- **Links** The loop is not knotted, but the drawing may hide this
- **Difficulty** A simple knot can have a very complicated diagram
- **Search** Find the moves that reveal the simple picture

Two case studies



- **Problem** How many crossing flips are needed to untie a knot?
- **Catch** The careful crossing may not be visible in the first picture
- **Trick** First change the picture, then try the crossing flip

There is still much to do...

Why the excitement?



- **Tenacity** Mathematics wants exact statements. AI gives plausible attempts
- **Message** Plausible attempts are useful when bad ones are cheap to catch
- **Method** Let AI suggest many things; keep only what mathematics accepts

Knot land



- **Object** A knot is a closed loop of string in space
- **Rule** You may move it, but not cut it or pass it through itself
- **Question** Identify it! (E.g. is this tangle secretly just the plain circle?)

Two case studies



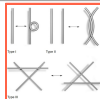
- **Case 1** State = diagram
- **State 2** action = legal move (1-2 add/remove cards, 3 shuffle)
reward = progress toward simplicity
- **Action** Try one legal local move, then try again

Why the excitement?



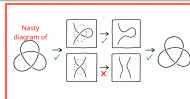
- **Text** AI is treated as "1000 unreliable but productive coworkers". It is used as a source of attempts (it does a lot of them) and attempts are cheap to reject (as it doesn't matter if the often wrong)
- **Message** Let AI suggest many things; keep only what mathematics accepts

Knot land



- **Diagram** We draw the loop in the plane with over/under crossings
- **Appearance** The same knot has many very different drawings
- **Moves** Small legal changes relate drawings of the same knot

Two case studies



- **Output** A concrete sequence of moves, not a philosophical essay
- **Check** Replay the sequence and verify every local step
- **Moral** The search is learned, but the certificate is mathematical

Why the excitement?



- **Text** AI is useful in exact but huge mathematical worlds new tools are rare, and this one is new (in math circles) thus, excitement comes from exploration plus checkability
- **Message** Imagine you have 1000 people to throw at a problem

Knot land



- **Links** The loop is not knotted, but the drawing may hide this
- **Difficulty** A simple knot can have a very complicated diagram
- **Search** Find the moves that reveal the simple picture

Two case studies



- **Problem** How many crossing flips are needed to untie a knot?
- **Catch** The careful crossing may not be visible in the first picture
- **Trick** First change the picture, then try the crossing flip

Thanks for your attention!