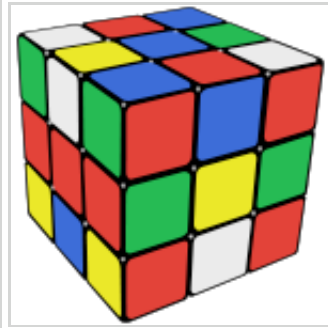
Optimal solutions for Rubik's Cube

From Wikipedia, the free encyclopedia


Rubik's cube scrambled

There are many algorithms to solve scrambled Rubik's Cubes. The minimum number of face turns needed to solve any instance of the Rubik's cube is 20.[1] This number is also known as the diameter of the Cayley graph of the Rubik's Cube group. An algorithm that solves a cube in the minimum number of moves is known as God's algorithm.

There are two common ways to measure the length of a solution. The first is to count the number of quarter turns. The second is to count the number of face turns. A move like F2 (a half turn of the front face) would be counted as 2 moves in the quarter turn metric and as only 1 turn in the face metric.

Contents

Lower bounds

It can be proven by counting arguments that there exist positions needing at least 18 moves to solve. To show this, first count the number of cube positions that exist in total, then count the number of positions achievable using at most 17 moves. It turns out that the latter number is smaller.

This argument was not improved upon for many years. Also, it is not a constructive proof: it does not exhibit a concrete position that needs this many moves. It was conjectured that the so-called *superflip* would be a position that is very difficult. The superflip is a position on the cube where all the middle squares and all corners are in their correct position, but each edge is of a different color. [2] In 1992, a solution for the superflip with 20 face turns was found by Dik T. Winter, of which the minimality was shown in in 1995 by Michael Reid , providing a new lower bound for the diameter of the cube group. Also in 1995, a solution for superflip in 24 quarter turns was found by Michael Reid, with its minimality proven by Jerry Bryan. [2] In

1998, a new position requiring more than 24 quarter turns to solve was found. The position, which was called a 'superflip composed with four spot' needs 26 quarter turns. [3]

Upper bounds

Thistlethwaite's Algorithm

The first upper bounds were based on the 'human' algorithms. By combining the worst-case scenarios for each part of these algorithms, the typical upper bound was found to be around 100. The breakthrough was found by Morwen Thistlethwaite; details of Thistlethwaite's Algorithm were published in *Scientific American* in 1981 by Douglas Hofstadter. The approaches to the cube that lead to algorithms with very few moves are based on group theory and on extensive computer searches. Thistlethwaite's idea was to divide the problem into subproblems. Where algorithms up to that point divided the problem by looking at the parts of the cube that should remain fixed, he divided it by restricting the type of moves you could execute. In particular he divided the cube group into the following chain of subgroups:

- $G_0 = \langle L,R,F,B,U,D \rangle$
- $G_1 = \langle L,R,F,B,U2,D2 \rangle$
- $G_2 = \langle L,R,F2,B2,U2,D2 \rangle$
- $G_3 = \langle L2,R2,F2,B2,U2,D2 \rangle$
- $G_4 = \{I\}$

Next he prepared tables for each of the right coset spaces $G_{[i+1]} \backslash G_i$. For each element he found a sequence of moves that took it to the next smaller group. After these preparations he worked as follows. A random cube is in the general cube group G_0. Next he found this element in the right coset space $G_1 \backslash G_0$. He applied the corresponding process to the cube. This took it to a cube in G_1. Next he looked up a process that takes the cube to G_2, next to G_3 and finally to G_4.

Although the whole cube group G_0 is very large (~4.3×10^{19}), the right coset spaces $G_1 \backslash G_0$, $G_2 \backslash G_1$, $G_3 \backslash G_2$ and G_3 are much smaller. The coset space $G_2 \backslash G_1$ is the largest and contains only 1082565 elements. The number of moves required by this algorithm is the sum of the largest process in each step. In the original version this was 52.

Kociemba's Algorithm

Thistlethwaites algorithm was improved by Herbert Kociemba in 1992. He reduced the number of intermediate groups to only two:

- $G_0 = \langle L,R,F,B,U,D \rangle$
- $G_1 = \langle L,R,F2,B2,U2,D2 \rangle$
- $G_2 = \{I\}$.

As with Thistlethwaite's Algorithm, he would search through the right coset space $G_1 \backslash G_0$ to take the cube to group G_1. Next he searched the optimal solution for group G_1. The searches in $G_1 \backslash G_0$ and G_1 were both done with a method equivalent to IDA*. The search in $G_1\backslash G_0$ needs at most 12 moves and the search in G_1 at most 18 moves, as Michael Reid showed in 1995. By generating also suboptimal solutions that take the cube to group G_1 and looking for short solutions in G_1, you usually get much shorter overall solutions. Using this algorithm solutions are typically found of fewer than 21 moves, though there is no proof that it will always do so.

In 1995 Michael Reid proved that using these two groups every position can be solved in at most 29 face turns, or in 42 quarter turns. This result was improved by Silviu Radu in 2005 to 40.

Korf's Algorithm

Using these group solutions combined with computer searches will generally quickly give very short solutions. But these solutions do not always come with a guarantee of their minimality. To search specifically for minimal solutions a new approach was needed.

In 1997 Richard Korf[4] announced an algorithm with which he had optimally solved random instances of the cube. Of the ten random cubes he did, none required more than 18 face turns. The method he used is called IDA* and is described in his paper "Finding Optimal Solutions to Rubik's Cube Using Pattern Databases." Korf describes this method as follows

> IDA* is a depth-first search that looks for increasingly longer solutions in a series of iterations, using a lower-bound heuristic to prune branches once a lower bound on their length exceeds the current iterations bound.

It works roughly as follows. First he identified a number of subproblems that are small enough to be solved optimally. He used:

1. The cube restricted to only the corners, not looking at the edges
2. The cube restricted to only 6 edges, not looking at the corners nor at the other edges.
3. The cube restricted to the other 6 edges.

Clearly the number of moves required to solve any of these subproblems is a lower bound for the number of moves you will need to solve the entire cube.

Given a random cube C, it is solved as iterative deepening. First all cubes are generated that are the result of applying 1 move to them. That is C * F, C * U, … Next, from this list, all cubes are generated that are the result of applying two moves. Then three moves and so on. If at any point a cube is found that needs too many moves based on the upper bounds to still be optimal it can be eliminated from the list.

Although this algorithm will always find optimal solutions there is no worst case analysis. It is not known how many moves this algorithm might need. An implementation of this algorithm can be found here.[5]

Further improvements

In 2006, Silviu Radu further improved his methods to prove that every position can be solved in at most 27 face turns or 35 quarter turns.[6] Daniel Kunkle and Gene Cooperman in 2007 used a supercomputer to show that all unsolved cubes can be solved in no more than 26 moves (in face-turn metric). Instead of attempting to solve each of the billions of variations explicitly, the computer was programmed to bring the cube to one of 15,000 states, each of which could be solved within a few extra moves. All were proved solvable in 29 moves, with most solvable in 26. Those that could not initially be solved in 26 moves were then solved explicitly, and shown that they too could be solved in 26 moves. [7] [8]

Tomas Rokicki reported in 2008 computational proof that all unsolved cubes could be solved in 25 moves or fewer.[9] This was later reduced to 23 moves.[10] In August 2008 Rokicki announced that he had a proof for 22 moves.[11] In 2009, Tomas Rokicki proved that 29 moves in quarter turn metric is enough to solve any scrambled cube. [12] Finally, in 2010, an international Group around Morley Davidson gave the final proof that all cube positions could be solved with a maximum of 20 face turns.[1]

References

1. ^ ***a b*** God's Number is 20 (http://www.cube20.org/)
2. ^ ***a b*** Michael Reid's Rubik's cube page M-symmetric positions (http://www.math.ucf.edu/~reid/Rubik/m_symmetric.html)
3. **^** Posted to Cube lovers on 2 Aug 1998 (http://www.math.ucf.edu/~reid/Rubik/Cubelovers/cube-mail-25)
4. **^** Richard Korf's Finding Optimal Solutions to Rubik's Cube Using Pattern Databases (http://www-compsci.swan.ac.uk/~csphil/CS335/korfrubik.pdf)
5. **^** Michael Reid's Optimal Solver for Rubik's Cube (http://www.math.ucf.edu/~reid/Rubik/optimal_solver.html) (requires a compiler such as gcc)
6. **^** Rubik can be solved in 27f (http://cubezzz.homelinux.org/drupal/?q=node/view/53)
7. **^** Press Release on Proof that 26 Face Turns Suffice (http://www.neu.edu/nupr/news/0507/rubik.html)
8. **^** Kunkle, D.; Cooperman, C. (2007). "Twenty-Six Moves Suffice for Rubik's Cube" (http://www.ccs.neu.edu/home/gene/papers/rubik.pdf) (PDF). *Proceedings of the International Symposium on Symbolic and Algebraic Computation (ISSAC '07)* (http://www.ccs.neu.edu/home/gene/papers/rubik.pdf) . ACM Press. http://www.ccs.neu.edu/home/gene/papers/rubik.pdf.
9. **^** Tom Rokicki. "Twenty-Five Moves Suffice for Rubik's Cube" (http://arxiv.org/PS_cache/arxiv/pdf/0803/0803.3435v1.pdf) . http://arxiv.org/PS_cache/arxiv/pdf/0803/0803.3435v1.pdf. Retrieved 2008-03-24.
10. **^** Twenty-Three Moves Suffice (http://cubezzz.homelinux.org/drupal/?q=node/view/117) — Domain of the Cube Forum
11. **^** twenty-two moves suffice (http://cubezzz.homelinux.org/drupal/?q=node/view/121)
12. **^** Tom Rokicki. "Twenty-Nine QTM Moves Suffice" (http://cubezzz.homelinux.org/drupal/?q=node/view/143) . http://cubezzz.homelinux.org/drupal/?q=node/view/143. Retrieved 2010-02-19.

External links

- How to solve the Rubik's Cube, a Wikibooks article that describes an algorithm that has the advantage of being simple enough to be memorizable by humans, however it will usually not give an *optimal* solution which only uses the minimum possible number of moves.
- Herbert Kociemba's Two-Phase-Solver and Optimal Solver for Rubik's Cube (http://kociemba.org/cube.htm)
- Ryan Heise's Human version of the Thistlethwaite algorithm (http://www.ryanheise.com/cube/human_thistlethwaite_algorithm.html)

- A New Upper Bound on Rubik's Cube Group, Silviu Radu (http://arxiv.org/abs/math.CO/0512485)
- Online Solver using modified Kociemba's Algorithm to balance optimization vs. compute cycles (http://rubiksolve.com)

Categories: Rubik's Cube