WHAT IN THE WORLD IS A RUBIK’S CUBE???

• Originally, a 3x3x3 Cube Shaped Puzzle.
  • There are now 2x2x2 all the way up to 10x10x10.

• Rotate the faces to make all of the colored stickers match up.

• When complete, there should be one color per side.
THE TECHNICAL DETAILS BEHIND THE CUBE

• Every 3x3x3 Cube is made up of 27 smaller cubes, or “Cubies”
• Each “Cubie” has a set of faces of different colors that make it unique
• Moves are the action of rotating the 9 “Cubies” that form a side and rotating it either 90 or 180 degrees around the central “Cubie”
• One a 3x3x3 cube, the center “Cubie” never changes, meaning the color of the center dictates the color for the entire face.
RUBIK’S CUBE NOTATIONS

• When solving, computers don’t associate with colors.

• Computers associate with orientation of the cube, notated with the first letter:
  • U(p), Front, R(ight), D(own), B(ack), L(eft).

• When rotating 90 degrees clockwise, only the face is printed. (ex: R)
  • Sometimes printed with a ‘+’ or ‘1’.

• If the rotation is counterclockwise, the face is followed by a prime symbol. (ex: R’)
  • Sometimes printed with a ‘3’.

• When rotating 180 degrees, the face is followed by a 2. (ex: R2)

• The set of moves is defined as S.
RUBIK'S CUBE PROGRAM REPRESENTATION

An example of scrambled cube is:

UL BD LB UF FL FD UR RF DR BU LD BR
BDL FRU BRD RFD BLU URB FUL FLD

A solved cube is represented as (12 edges, 8 corners)

UF UR UB UL DF DR DB DL FR FL BR BL
UFR URB UBL ULF DRF DFL DLB DBR
HISTORY BEHIND THE CUBE

- First created in 1974 by Erno Rubik; a Professor of architecture.
- It took Professor Rubik over a month to solve his invention.
- The original name was the ‘Magic Cube’, but was changed to what we all know as the ‘Rubik’s Cube’ in 1980.
  - Ideal Toy Company executives, who had agreed to sell it to the rest of the world, thought that the original name had overtones of witchcraft and it needed to change.
- Since the launch in 1980 an estimated 350 million cubes have been sold.
- Approximately 1 in 7 people have played with a cube.
QUESTIONS WORTH PONDERING

1.) What is God’s Number?
2.) How was God’s Number calculated?
3.) God’s Number on NxNxN cubes?
4.) What is Disk-Based Parallel Computing?
5.) What are the results of a parallel implementation for solving the cube?
Every cube solver uses an algorithm, or sequence of steps, to solve the cube. One algorithm might begin with a sequence to solve a single face, while others might start off by trying to achieve some other cube state.

God’s Number is the lowest possible number of turns, in which the cube can be solved, no matter the position.

A year after the cubes official release in 1980, Morwen Thistlewaite published the first algorithm for solving the 3x3x3 cube.

- Lower Bound: 18 Moves
- Upper Bound: 52 Moves
<table>
<thead>
<tr>
<th>Date</th>
<th>Lower bound</th>
<th>Upper bound</th>
<th>Gap</th>
<th>Notes and Links</th>
</tr>
</thead>
<tbody>
<tr>
<td>July, 1981</td>
<td>18</td>
<td>52</td>
<td>34</td>
<td>Morwen Thistlethwaite proves 52 moves suffice.</td>
</tr>
<tr>
<td>December, 1990</td>
<td>18</td>
<td>42</td>
<td>24</td>
<td>Hans Kloosterman improves this to 42 moves.</td>
</tr>
<tr>
<td>May, 1992</td>
<td>18</td>
<td>39</td>
<td>21</td>
<td>Michael Reid shows 39 moves is always sufficient.</td>
</tr>
<tr>
<td>May, 1992</td>
<td>18</td>
<td>37</td>
<td>19</td>
<td>Dik Winter lowers this to 37 moves just one day later!</td>
</tr>
<tr>
<td>January, 1995</td>
<td>18</td>
<td>29</td>
<td>11</td>
<td>Michael Reid cuts the upper bound to 29 moves by analyzing Kociemba's two-phase algorithm.</td>
</tr>
<tr>
<td>January, 1995</td>
<td>20</td>
<td>29</td>
<td>9</td>
<td>Michael Reid proves that the &quot;superflip&quot; position (corners correct, edges placed but flipped) requires 20 moves.</td>
</tr>
<tr>
<td>December, 2005</td>
<td>20</td>
<td>28</td>
<td>8</td>
<td>Silviu Radu shows that 28 moves is always enough.</td>
</tr>
<tr>
<td>April, 2006</td>
<td>20</td>
<td>27</td>
<td>7</td>
<td>Silviu Radu improves his bound to 27 moves.</td>
</tr>
<tr>
<td>May, 2007</td>
<td>20</td>
<td>26</td>
<td>6</td>
<td>Dan Kunkle and Gene Cooperman prove 26 moves suffice.</td>
</tr>
<tr>
<td>March, 2008</td>
<td>20</td>
<td>25</td>
<td>5</td>
<td>Tomas Rokicki cuts the upper bound to 25 moves.</td>
</tr>
<tr>
<td>April, 2008</td>
<td>20</td>
<td>23</td>
<td>3</td>
<td>Tomas Rokicki and John Welborn reduce it to only 23 moves.</td>
</tr>
<tr>
<td>August, 2008</td>
<td>20</td>
<td>22</td>
<td>2</td>
<td>Tomas Rokicki and John Welborn continue down to 22 moves.</td>
</tr>
<tr>
<td>July, 2010</td>
<td>20</td>
<td>20</td>
<td>0</td>
<td>Tomas Rokicki, Herbert Kociemba, Morley Davidson, and John Dethridge prove that God’s Number for the Cube is exactly 20.</td>
</tr>
</tbody>
</table>
LATEST ALGORITHM: KOCIEMBA’S

• Identifies a subset of 20 billion positions as H.
  • Michael Reid showed that every position in the subset takes at most 18 moves to solve.
  • Every cube position is at most 12 moves from this subset.

• Every position in H is defined with 2 characteristics.
  • All corners and edges are properly oriented.
  • The edge cubes that belong in the middle layer are located in the middle layer.

• Remaining problem is how to transform a cube position into a position in H.
KOCIEMBA’S CONTINUED

• All positions in H can be “relabeled” so all position in H have the same physical appearance.

• With the conditions in the previous slide, the stickers on each cubie can be removed and relocated to make the same colored cube.

• Since any position is within 12 moves of the set, the cube can be solved within 30 moves
KOCIEMBA’S STILL HAS MORE TO SAY!

• $r(p)$ is the relabeling of a cube

• $d_2$ is a table lookup that takes a position and returns the distance to the identity element using moves

• phase 2 is the sequence to take the position into a fully solved state
KOCIEMBA’S ALGORITHM - PSUDOCODE

1: \( d \leftarrow 0 \)
2: \( b \leftarrow \infty \)
3: while \( d < b \) do
   4:   for \( s \in S^d, r(ps) = e \) do
      5:     if \( d + d_2(ps) < b \) then
         6:       Solve phase 2; report new better solution
         7:       \( b = d + d_2(ps) \)
      8:     end if
   9:   end for
10: \( d \leftarrow d + 1 \)
11: end while
• Primarily, we are interested in finding the minimum number of turns needed for all positions to be solved.
  • Total number of possible positions is $4.51 \times 10^{17}$.

• Even with finding the move sequences (of 20 moves or less) at a rate of 240 positions for second, finding the move sequences for all positions would take more than seven thousand computers more than seven thousand years.

• At the point computers can find nine hundred million positions a second, it will only take four years to finish the proof.
NEXT UP –
GOD’S NUMBER:
INFORMATIONAL VIDEO
Q3 - GOD’S NUMBER ON NxNxN CUBES?

• At this point in time there is no answer or algorithm that can solve NxNxN cubes once the value of N exceeds 3.
  • This is simply because of the number of possible positions is astronomical.
Q4 - WHAT IS DISK-BASED PARALLEL COMPUTING?

• Disk-Based computation is a way to increase working memory and achieve results that are not otherwise economical.

• Access disks from cluster in parallel in place of ram.

• Discs have a latency and the network needs to keep up.
WHY DISC-BASED?

• Not enough ram on most motherboards
• Nearing the end of “Moore’s Law”
• A large amount of space is needed (6 TB)
DISK-BASED: SORTING DUPLICATES

• Breadth First Search to store the list on the disks.
• Then, bucket sort to eliminate duplicate states.
  • Use the RAM as the cache
• The list requires 6 TB of disk space, but can use more.
DISK-BASED: STORING STATES

• Representation of state could only use 2 bits per state

• Mathematical group theory was used to design a perfect hash function for all states

• Hash slots store the level in the search tree modulo 3
  • Able to be done since each turn only changes the number of moves by -1, 0, or 1
ANOTHER PARALLEL IMPLEMENTATION

• Every processor searches separately for a solution
• Avoids communication by having every processor initialize it’s own database
• Communicates with processor 0
ANOTHER PARALLEL COMPUTATION CONT.

• Processing stops when the goal is found by one with an optimal solution

• Sends information to processor 0, then to the others
  • Done asynchronously

• Upon receiving the message that the goal was found, processor exits

• With more processors, this reduces the search time more than serial
PSEUDO CODE

---

1. Initialize the Cube
2. Initialize the tables
3. $n$ processors
4. Search the tree
5. If goal found?
   - No: Go back to search tree.
   - Yes: Communicate and stop.
6. Search the tree

---

search limit
Q5 - WHAT ARE THE RESULTS OF A PARALLEL IMPLEMENTATION FOR SOLVING THE CUBE?

• All results we discovered were positive.
  • Speedup and efficiency all varied dependent upon the algorithm used.
SOME RESULTS

Table 5.1: Data for average time Vs depth (12-18) for Reid’s sequential implementation

<table>
<thead>
<tr>
<th>Depth</th>
<th>Conf 1</th>
<th>Conf 2</th>
<th>Conf 3</th>
<th>Conf 4</th>
<th>Conf 5</th>
<th>Average time in seconds</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>103</td>
<td>101</td>
<td>101</td>
<td>102</td>
<td>100</td>
<td>101.3</td>
</tr>
<tr>
<td>14</td>
<td>100</td>
<td>103</td>
<td>104</td>
<td>100</td>
<td>100</td>
<td>101.4</td>
</tr>
<tr>
<td>15</td>
<td>103</td>
<td>104</td>
<td>100</td>
<td>100</td>
<td>101</td>
<td>101.6</td>
</tr>
<tr>
<td>16</td>
<td>99</td>
<td>101</td>
<td>100</td>
<td>99</td>
<td>100</td>
<td>99.8</td>
</tr>
<tr>
<td>17</td>
<td>315</td>
<td>104</td>
<td>104</td>
<td>118</td>
<td>114</td>
<td>151</td>
</tr>
<tr>
<td>18</td>
<td>379</td>
<td>166</td>
<td>157</td>
<td>155</td>
<td>142</td>
<td>199.8</td>
</tr>
</tbody>
</table>

Fig 5.2: Average time Vs Depth (19-22) for Reid’s sequential implementation
MORE RESULTS

Fig 5.5: Comparisons of time in minutes required to solve the cube for 8 random cubes of depth 21
SUMMATION OF Q & A’S

1.) What is God’s Number?
   The max number it takes to solve the cube no matter the orientation of the cube.

2.) How was God’s Number concluded?
   Algorithms over time got better and better and computing power increased to allow for mass computation for all subsets of positions to be solved.

3.) God’s Number on NxNxN cubes?
   At this point in time it is virtually impossible for cubes larger than a 3x3x3.

4.) What is Disk-Based Parallel Computing?
   Disk-Based computation is a way to increase working memory and achieve results that are not otherwise economical.

5.) What are the results of a parallel implementation for solving the cube?
   All results were positive. Some more than others depending on the algorithm used to solve the cube.
WORKS CITED

[1] https://rubiks.com/about/the-history-of-the-rubiks-cube
IMAGE WORKS CITED

AND THAT’S THE END…
ANY FURTHER QUESTIONS?