Solving Skewb with Monte Carlo

Law Chun Man

PHYS 4061 Project B

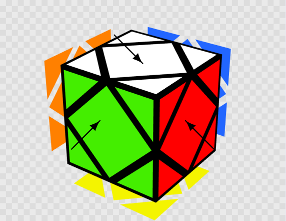
November 21, 2024

Aim of This Project

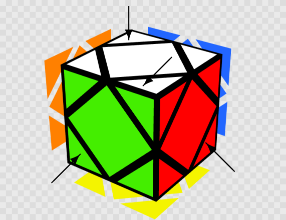
- Allow user to solve skewb **easily**!
- Even without prior knowledge to skewb.

Understanding the Mechanism of a Skewb

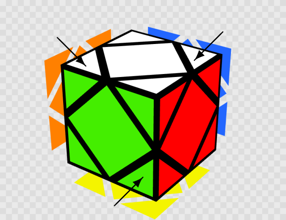
Centre



Attached corners



Floating corners



God’s Number

- Maximum number of moves needed to solve a Rubik’s cube.

God’s Number

- Maximum number of moves needed to solve a Rubik’s cube.
- God’s number for skewb is 11 moves.

Define Turns

- There are only 4 different sides to turn.
- 8 different turns in total.
- Map R to 0, R' to 1, and so on...

Monte Carlo

Theory

If you try hard enough, you will eventually find the solution.

Code

Generate a random move for each step, and check if the cube was solved.

Weighted Monte Carlo

Global array

```
int ratio[4] = {1, 1, 1, 1};
```

Weighted Monte Carlo function

```
int side = rand()%(ratio[0]+ratio[1]+ratio[2]+ratio[3]); //choose which side to turn
int clockwise = rand()%2; //0: counterclockwise, 1: clockwise

if(side<ratio[0]){ //if right side is chosen
    if(clockwise==1){current_move = 0;} //set current move to R
    else{current_move = 1;} //set current move to R'
}
```

Check Duplicated Moves

Find next move function

```
int current_move;
if(last_move==0 || last_move==1){//if the last move was R or R',
    the next move should not contain R
     do{
          current_move = weighted_monte_carlo();
          }while(current_move==0 || current_move==1);
}
//...
```

Actual Solve

Solve skewb function

```
int solved = 0; //0: not yet solved, 1: solved
while(!solved&&iteration<MAX_ITERATION){
    for(int i=0; i<GODS_NUMBER; i++){
        current_move = find_next_move(last_move); //find next
           move that is different from previous move
        last_move = current_move; //store current move
        turn[current_move](); //turn the cube
        solved = check(); //check if the cube is solved
        if(solved==1){
            break; //then stop the iteration
        }
    }
}
```

Scan Cube Using Webcam

- Demo video: Skewb solver programme demo

Reviewing Carter Kucala Solves

Solve	Result	Scramble (11 moves)	Fewest number of moves
1	2.49	R B L’ U’ L’ R B’ R’ B R B	9
2	1.77	U L U’ B’ U’ B’ U’ L B U’ R’	9
3	3.32	U L B U’ B’ L’ B R’ B’ L U	7
4	4.65	R B’ U’ B U B L’ U B’ U R	8
5	0.75 (WR!)	B L R’ B’ L U B’ R U’ R U	8

- The third fastest solve (0.85) was done by Simon Kellum in this competition, solving the same scramble!

The End

Thank

you

More details

If you are really interested, you are welcome to ask for the code.

Download this PDF