Martin Cochran

Proof 2.2

Feb. 12, 2002

Problem: Use the processes we have noted to write down an algorithm for solving the Rubiks Cube. Specifically, list your algorithm steps and explain which of our processes to use in order to attain the goals of that step (example is excluded - however, the process in the example is used in the algorithm).

Proof: The general outline of the algorithm is as follows:

a. Get all upper corner cubelets on the upper layer and in the correct position.

b. Correctly orient the corner cubelets in the upper layer.

c. Correctly position the bottom corner cubelets.

d. Correctly orient the bottom corner cubelets.

e. Position all of the edge cubelets correctly.

f. Correctly orient the edge cubelets.

This algorithm was chosen because there were more processes at the author's disposal that fixed the corner cubelets and permuted the edge cubelets than there were processes that fixed the edge cubelets and permuted the corner cubelets. This is why all the corner cubelets are positioned and oriented before edge cubelets are even considered.

a. Get all upper corner cubelets on the upper layer and in the correct position.

Here we will refer to an algorithm developed in a previous proof to correctly positition the upper layer: For each of the four corner cubelets to be positioned, either the desired cubelet is on the bottom level or the top level of the cube. If the cubelet is on the bottom level, then we will separate the algorithm into two cases: (i) the cubelet is directly 'below' the desired position (e.g. we want a cubelet that is currently in the frd position to move to the fru postition); (ii) the cubelet is not directly below the desired position. In both cases, we are looking for a conjugation of D or D^{-1} by one of F, B, R, L or their inverses, but in the first case we will want to move the cubelet out of the way first. So, if we use the example in the first case, we will move the cubelet from the frd position to the fld postition by D^{-1} then we will perform the sequence $R^{-1}DR$. The R^{-1} does not move the cubelet, then the D moves the cubelet back to the frd postition, then the R moves the cubelet to the desired postition. Similar processes can be derived for slightly different circumstances - but the idea is that we move the desired cubelet onto the bottom layer of the cube and then conjugate D or D^{-1} by one of F, B, R, L or their inverses.

If the cubelet is already on the top but in a wrong position, we will want to move the cubelet to the bottom position where we know what to do with it. This will also just be the conjugation of D or D^{-1} by one of F, R, L, B or their inverses. For example, if the cubelet is in the blu position and we want to move it to the bru position, we perform the sequence BDB^{-1} to move the cubelet to the fld postion, and now we use the previous algorithm to move the cube to the bru postition.

There are a couple of shortcuts that might come in handy, however. If there are two cubelets already on the top layer that need their positions switched - we will use the following example: the cubelet currently in the fur postion needs to be moved to the ful position and vice versa - then $[F, R] = FRF^{-1}R^{-1} = (ful, fru)_+(fdr, dbr)_-(uf, rf, rd)$ will switch the two.
Another nice process is when all four top layer cubelets are already in the top layer and one of them is in postition but the other three aren't. Then either one or two applications of $(flu, rub, rfu)(fu, ru, dr) = [F, R]U[R, F]U^{-1}$ will do the trick. This move was taken from the Rubiks Cube Handout.

b. Correctly orient the corner cubelets in the upper layer.

There are a couple of options here because we don't have to worry about stabilizing many cubelets.

Case 1: Two adjacent corner cubelets both need a +1 or -1 rotation. The move

$$[F, R]^2 = (ful)_+(fur)_+(fdr)_-(bdr)_-(uf, rd, rf)$$

or the inverse of this move, if more convenient, will rotate the cubelets accordingly with the faces defined appropriately.

Case 2: Three of the cubelets all need a +1 or -1 rotation. The following move from the Rubiks Cube Handout from class or its inverse will do the trick:

$$(flu)_-(fur)_-(rub)_-(fu, fl, ru, df, fr) = [F, U]^2([F, R]^2 R^{-1}[R, D]^2 R)$$

Case 3: One cubelet needs a +1 rotation and one adjacent to it needs a -1 rotation. This move, also taken from the handout, will do just that:

$$(flu)_+(fdl)_-(fu, df, fr) = [F, R]^2 R^{-1}[R, D]^2 R$$

Default Case: There is a corner that needs rotating. We will refer to the algorithm given in a solution to problem 2.1. The following is a list of moves for the +1 or -1 rotations of any corner cubelet:

+1 rotation of fur: $[D, F]^2$

-1 rotation of fur: $[F, D]^2$

+1 rotation of ful: $[D, L]^2$

-1 rotation of ful: $[L, D]^2$

+1 rotation of bul: $[D, B]^2$

-1 rotation of bul: $[B, D]^2$

+1 rotation of bur: $[D, R]^2$

-1 rotation of bur: $[R, D]^2$

Note that these moves also permute a bunch of other things (see case 1), but these other cubelets are not on the upper layer.

c. Correctly position the bottom corner cubelets.

If the bottom cubelets are not in order, then we have some work to do. We can assume that at least one cubelet can be positioned correctly by applying D until one of the cubelets is in the right place. The same shortcut for switching three cubelets that are out of place mentioned previously ($(flu, rub, rfu)(fu, ru, dr) = [F, R]U[R, F]U^{-1}$) in this algorithm (at the end of step a) works here because it stabilzes the already correctly positioned top cubelets. If only two cubelets are out of place there are two cases: the two cubelets are adjacent or they are diagonally across from each other. In the latter case just apply the three-corner switch to get two cubelets out of place adjacent to each other. Now, apply the following algorithm taken from the little booklet that came with the Rubik's cube (note: this is the only move I looked at in the booklet):

$$R^{-1}D^{-1}RFDF^{-1}R^{-1}DRD^2 = (fdl, dfr)(dbl)_-(df, fd)(dl, db, ld, bd)$$

This will swap two corner cubelets, so now all of the corner cubelets should be in the correct position, and the top cubelets should have the correct orientation as well.

d. Correctly orient the bottom corner cubelets.

Just use the cases 2-3 covered previously in step b. We don't need to worry about case 1 because if two cubelets are out of orientation, then by the symmetry of the Rubik's Cube, one cubelet must need a +1 rotation and the other must need a -1 rotation. Similarly, if all four bottom cubelets are not in correct orientation, then two must need a +1 rotation and the other two must need a -1 rotation. There is one case which this does not cover - there are two cubelets that are not correctly oriented and they are diagonally across from one another. In this case, we will want to conjugate the move in case 3 with a set of moves to make the two corners adjacent to each other. For example, let's say that the *bul* cubelet needs a -1 rotation and the fur cubelet needs a +1 rotation. Then we would want to compose the process from case 3 with L, so that the final move would look like this:

$$L([F,D]^2D^{-1}[D,L]^2D)L^{-1}$$

Now all the corner cubelets should be in the correct position with correct orientation.

e. Position all of the edge cubelets correctly.

The solution to problem 1.6 was an algorithm for switching any two pairs of edge cubelets. Here is that algorithm:

¿From the solution to problem 1.5, we know how to swap two pairs of edge cubelets if two of the cubelets lie next to each other (that is, not directly across from one another - e.g. uf and ur on face U) on the same face A, and if the two other cubelets to be swapped are on the face opposite of face A, and directly 'under' the respective cubelet to be swapped with (note: this process is $(uf,df)(ur,dr) = (F^2R^2)^3$ - face A is the upper face). The algorithm to swap any two edge cubelets in general involves positioning the desired cubelets to be swapped so that we can use our familiar formula to swap cubelets in the positions just previously described.

Let A, B, C, and D refer to each of the four cubelets we wish to be involved in the swap, and let us wish to swap A with C and B with D. Let us start with A and say that A is already in the desired position (we get one free-bee) and now we will say that A is in the fu position, and define the faces of the cube according to that. Now let us put B in the desired position. From here on out we will want to make note of all the moves we do, because we will perform their inverses after the swap is made. This follows from the previous problem where it was shown that the conjugation of the sequence $(F^2R^2)^3$ by any sequence will only swap two sets of edge cubelets and leave the rest of the cubelets unchanged. We want B to move to the ru position, but since only the A cubelet must be stabilized while moving B to the ru position, we can safely assume that this is fairly trivial and can be left as an exercise to the reader.

When we are positioning the C cubelet, we have to be a little more careful because we want to stabilize both A and B cubelet. There are three cases for where the C cubelet can be: if we separate the cube into three layers - an upper, middle and down layer - then the cubelet must be on one of these layers. If the cubelet is on the down layer, then between zero and three repetitions of the D motion should position it accordingly. If the cubelet is on the middle layer, then a conjugation of D by an appropriate motion from one of F, L, R, or B should move it to the bottom layer, and then when it is on the bottom layer we should know what to do with it. For the last case, either C is in position lu or bu (because A and B are in the other two spots). If C is in position lu, then L^2 should move it to the bottom layer, and if C is in position bu, then B^2 should move it to the bottom layer - note that both of these moves do not disturb A or B. So now A, B, and C are in position.

Still trickier is getting D into position without disturbing the other three cubelets. If D is in any of positions bu, rb or lb, then between one and three repetitions of DBD^{-1} will rotate D to the desired position. Similarly, if D is in any of positions fr, bd, or ld, then between one and three repetitions of FDF^{-1} will move D to the correct spot. The only other possibilities for D to be are on the left face, and an L or two should move D into lb where it can be moved to the desired location, rd.

Now we should have a sequence, call it Q, for getting cubelets A, B, C, and D into the correct position, so finally, the sequence

$$Q(F^2R^2)^3Q^{-1}$$

Should yield a cube identical to the starting cube except with cubelet A swapped with cubelet C and cubelet B swapped with cubelet D.

But, this algorithm will not necessarily position all of the cubelets - there could be a triplet of edge cubelets that is out of place, in which case the process for switching two pairs will be of little use. In this case, one of the following moves taken from the Rubiks Cube Handout will work, although they may need to be conjugated by a set of moves in order to permute the correct cubes.

$$(uf, ur, ub) = (U^2R^2)^3(B^{-1}UB)(U^2R^2)^3(B^{-1}U^{-1}B) \tag{1}$$

$$(uf, ur, fr) = (FU^{-1}B^{-1}D)(R^2D^2)^3(D^{-1}BUF^{-1})(L^{-1}UL(F^2D^2)^3L^{-1}U^{-1}L) \tag{2}$$

$$(uf, ru, fr) = (RBL(U^2L^2)^3L^{-1}B^{-1}R^{-1})(DR^{-1}D^{-1}(L^2F^2)^3DRD^{-1}) \tag{3}$$

f. Correctly orient the edge cubelets.

The following process taken from class will swap the orientation of two cubelets. Again, this move may need to be conjugated with a set of moves to get the cubelets we wish to switch into the correct positions for this process:

$$(uf, fu)(df, fd) = (F^2R^2)^3(RD^{-1}RFL^{-1}F)(F^2R^2)^3(F^{-1}LF^{-1}R^{-1}DR^{-1})$$

Repeat this process as many times as necessary, and now the cube should be solved.