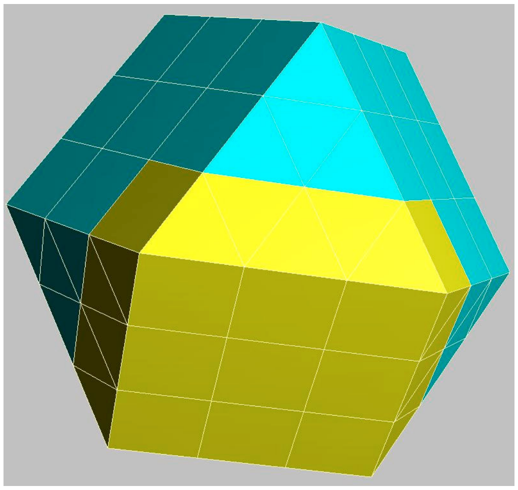
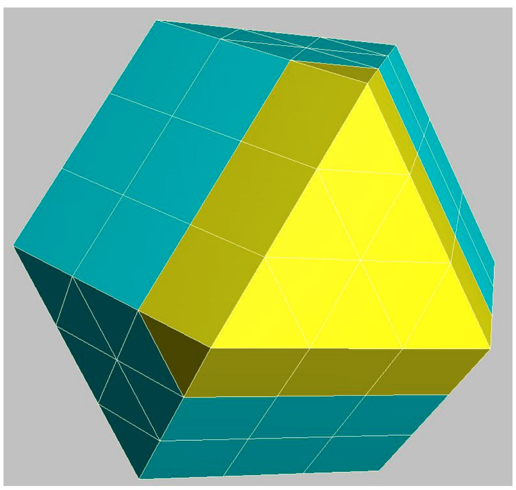
Virtual True Rubik Cuboctahedron

by Eduard Baumann

Rotational puzzles can be played virtually in a computer programm. If real versions of such a rotational puzzle are available on the marketplace then there is no doubt that they are superior to any virtual version.





It is possible that very obvious generalisations of real rotational puzzles are very hard to realize. The necessary mechanics can be very difficult or impossible.

In such cases a virtual programm may be more convincing. Lets look at the Cuboctahedron which still has a small number of faces (14). The illustrations show the rotated parts (slices) in another color.

The triangular slice is okay but the quadratic slice has a non planar border so a mechanical realisation is certainly difficult.

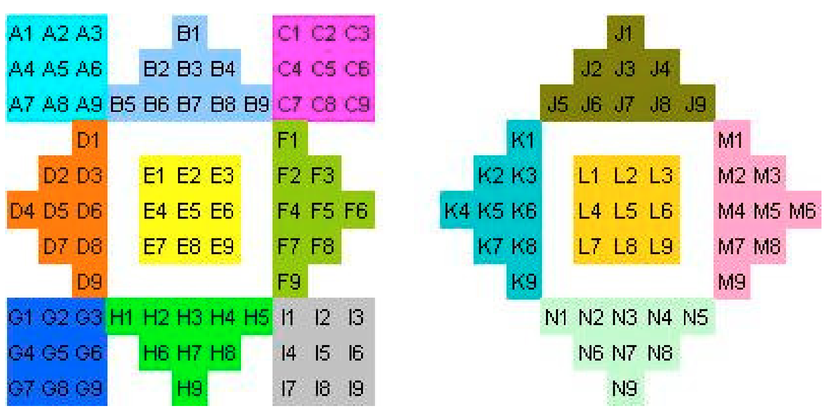
A programmed rotational puzzle can have the very big advantage of easy undo's and the automatic execution of compactly written sequences. It is also possible to retain the whole history of all movements done on the puzzle.

I recommend do this programming in a simple Excel worksheet. Use the following guidelines in doing so.

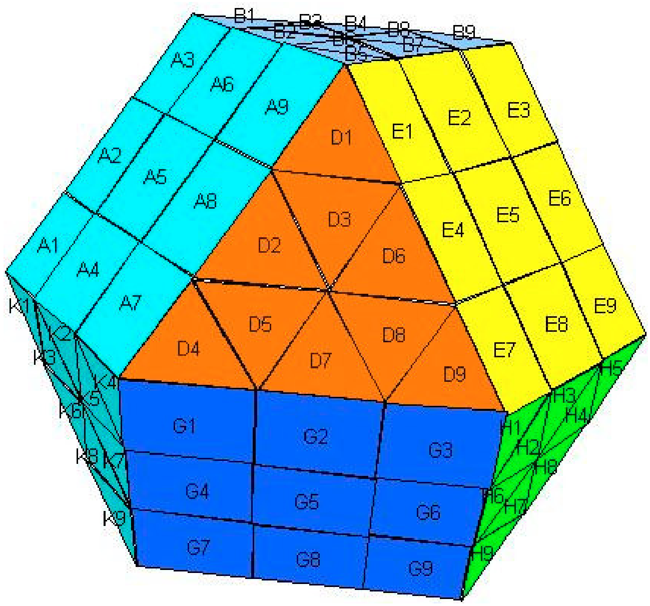
Principles

(a) record simple cell circulations
(b) give such recordings one letter names
(c) have one macro which calls all such recordings following a sequence given in form of a letter string in the selected cell
(d) the saving of positions are simple copy/pastes and 'undo' can also be prepared by copy/paste.

The forced cartesian arrangement of the cells in the Excelsheet shows as follows (shown are at left the front part and at right the back part seen from inside). A better



view is offered by a “showing” macro which transfers color and value of cells to the associated drawing elements. In order to not loose interesting aspects it is clear that all parts must be distinguishable. This can be obtained giving a individual letter for each label.



Here the front sight is shown with drawing elements.

In the following the letters C,D,E, … mean the turning of the corresponding slice 90° or 120° clockwise.

Only the turning of slice E and B and two turnings of the whole cuboctahedron (U for a rotation parallel to E and T for a rotation parallel to A) are predefined. With this all other turns can be constructed. For example C=$TE\underline{T}$.

Notation:

$\underline{T}$ is the inverse of T. The elements in round brackets change place in a cycle. [BE] is the commutator of B and E and means $BE\underline{BE}$.

The far neighbourhood commutators FNC like [EG] or [DB] are the simplest ones and yield 3-cycles for corners without any other effect because there is only one element in the intersection.

The near neighbourhood commutator NNC like [BE] has an intersection with 2 corner- one edge- and 2 face-elements. It can be used to arrange the edges in a first stage of an startegy ignoring all side effects.

Repeting such an NNC three times gives a 5-cycle for face elements without edge effect. Commuting with a turned version gives a smaller 3-cycle whose elements can further be put narrower together. w=[BE]^3, t=$Utw\underline{TU}$, X=[wt], Y=$EX\underline{E}$, Z=[XY] yields (D3 H2 B8) with some effects on the corners.

You can observe that all edge elements return to their home place without any twist.

For the corners we need a sequence which turns corners on place. The sequence EF turns the corner E3 by 180° with a lot of side effects. So the commutator with [JB] intersecting EF only at E3 gives the desired result. Q=[EF [JB]] turns the two corners E3 and C1 by 180°

With all this sequences you can apply the following strategy

(a) place all edge elements with NNC's
(b) place all face elements with Z
(c) place all corners with FNC
(d) orient the corners with Q

An Excelsheet with the True Rubik Cuboctahedron is available at baumann@mcnet.ch.